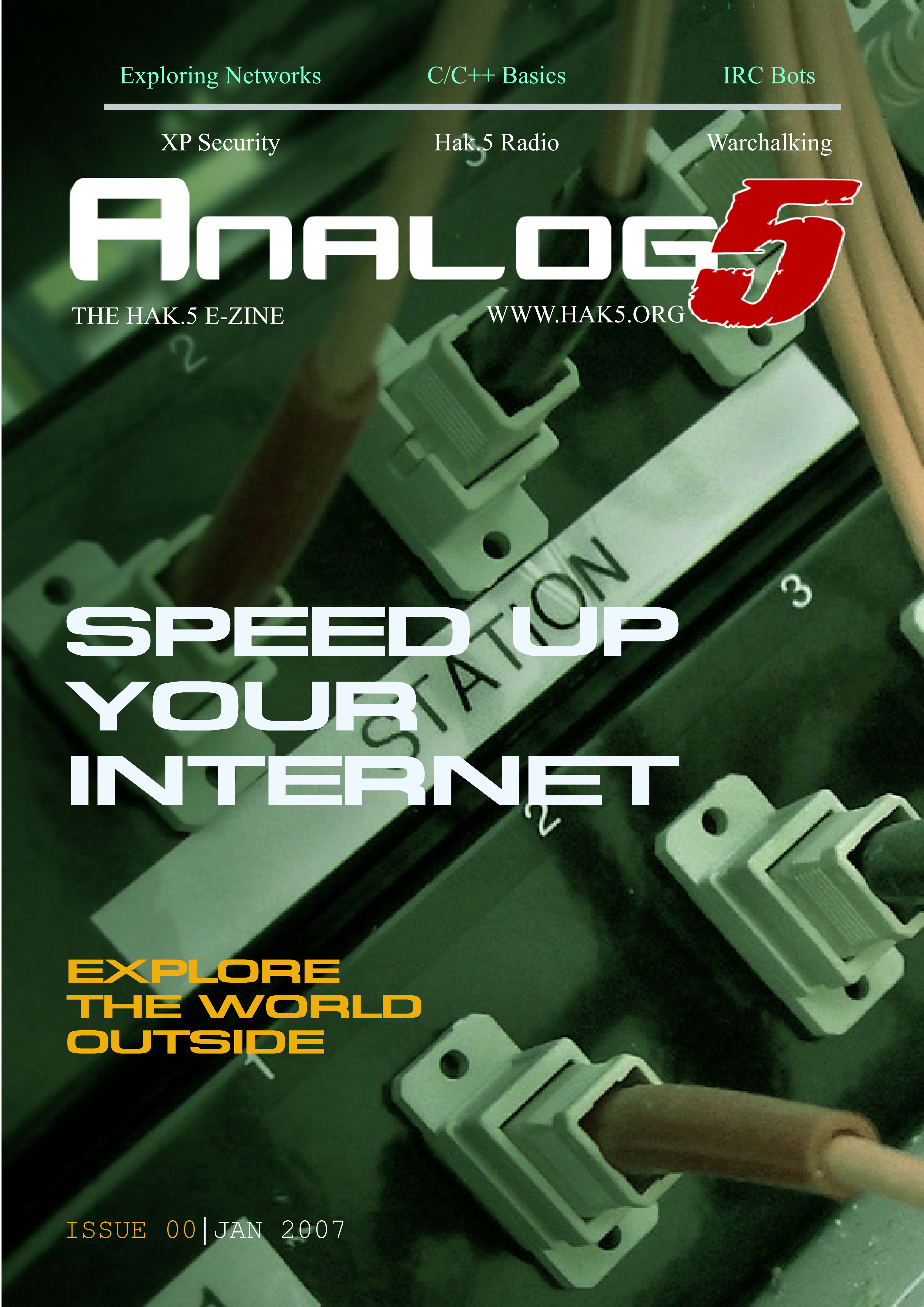
Exploring Networks · C/C++ Basics · IRC Bots

XP Security · Hak.5 Radio · Warchalking

Analog5

THE HAK.5 E-ZINE
WWW.HAK5.ORG



SPEED UP YOUR INTERNET

EXPLORE THE WORLD OUTSIDE

ISSUE 00 | JAN 2007

Analog5

The Hak.5 E-Zine

Analog5 magazine
January 2007, Issue 00
Check out the Hak.5 podcast, forum and more at
www.hak5.org

EDITORIAL

Famicoman
Editor-In-Chief
Famicoman@live.com

Mubix
Executive Editor
Jd.mubix@gmail.com

Moonlit
Executive editor
Moonlit@live.com

Brainedchild
Magazine Layout

CONTRIBUTING WRITERS:
Famicoman, Cooper, ChevronX, TomB, Mubix, TechCentric-Nick, nickisgod1, Moonlit, Shinmaryuu

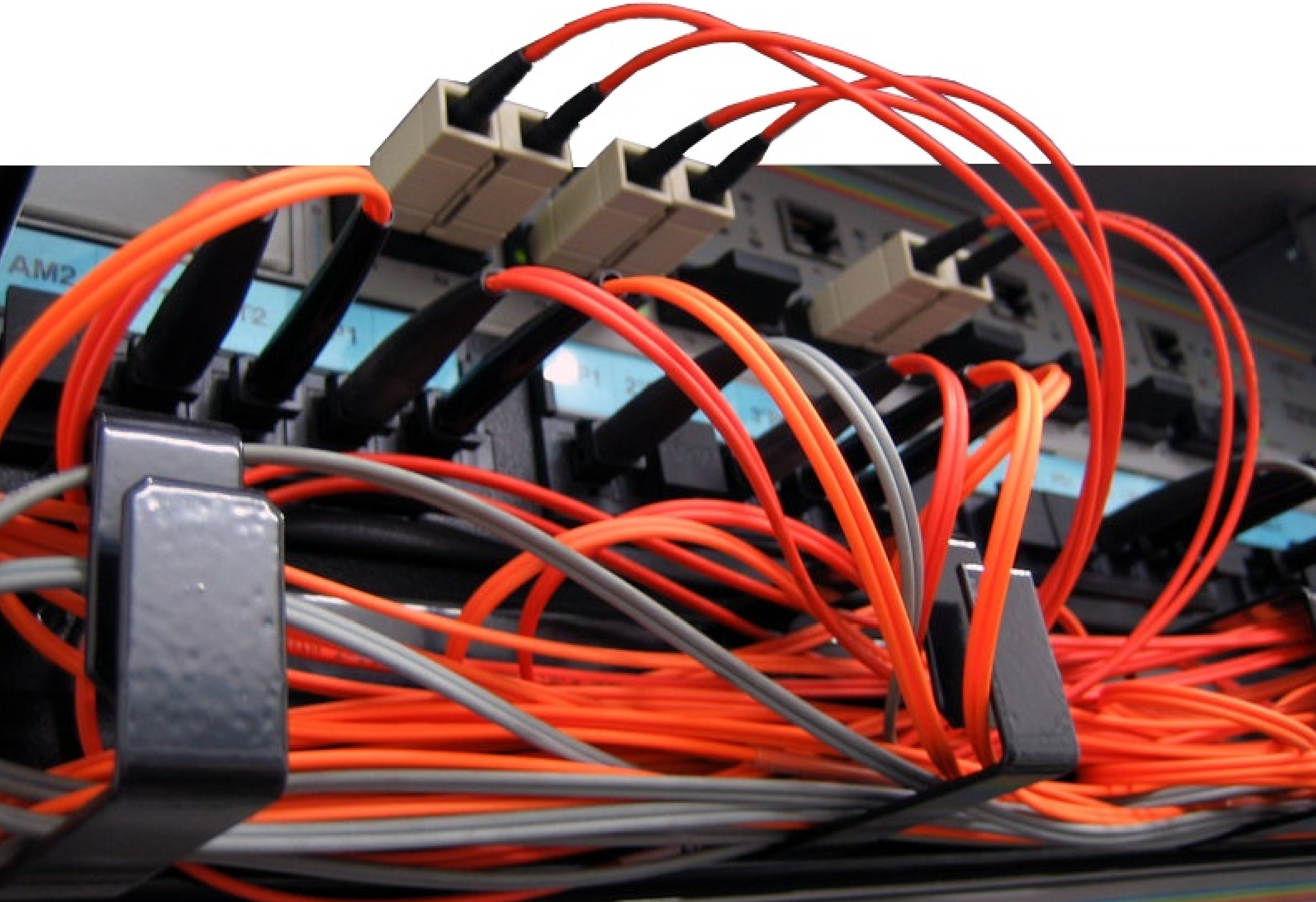
COVER ART:
Grossebox By Brujo (www.deviantart.com)

ADDITIONAL ART:
Images from Deviantart.com.
The Circles of Life By VictoriouSmiles (Pg. 2), Fiber Optics By Telli (Pg. 3), Gracies Stock Floor by FantasyStock (Pg. 5/6), Space Scene by Aziroth (Pg. 10), Robby by Dogmadic (Pg. 12/13), Molten by eccentricgfx (Pg. 14), Chalk Luigi by yooki42 (Pg. 15), Tubes by Othursdayz (Pg. 16), Coil by vuhstock (Pg. 17), Cables01 by pdtnc-stock (Pg. 18), Sith cat by Wessel-1984 (Pg. 19), Hand of Light by John_Bryan71 (Pg. 20), Blue waterfall by Della-stock (Pg. 21), Directory by Brujo (Pg. 22), Empath mixer by b_u_c_k_l_e_s_stock (Pg. 24), Power on by iceleaf_stock (Pg. 25)

Analog.5 is a hacking eZine designed by the Hak5 community, for the Hak5 community. The editors of Analog.5 ask you to create articles to appear in the ezine and help bring technolust to the masses. Articles can be any size, and submited by email. Almost all entries are instantly accepted. This zine is licensed under the Attribution-NonCommercial-ShareAlike 2.5 license. If you don't want your work to be redistributed, do not submit.
Analog5@Live.com

CONTENTS

JANUARY/FEBRUARY 2007



Analog.5

A Rundown

By Famicoman (hackinacan.siteled.com)

Well, a few of you reading this right now will most likely know everything that Analog5 is about. However, I'm guessing the majority of you readers may have no idea what this thing is, and just picked it up sometime this morning amidst the daily grind with cup of coffee in hand. If you haven't picked it up already, Analog5 is an eZine, or electronic magazine.

The history of electronic magazines is extensive at most, dating back to the days of Dial-Up BBS and the 300 Baud. But why release a BBS at this day in age? A day with email, Bittorrent, FIOS connections and instant messengers. A day nothing like a day twenty years ago. Frankly, when I stumble around the internet at my leisure I see all sorts of independent media. Media distributed how electronic ezines of the past would be distributed: free to anybody, and keeping it that way. Even though this is great, there is something missing, something that should be in place but isn't. Ezines seem to be lost in the heyday of independent movies, podcasts and IPTV. Ezines are a great medium for information and shouldn't be overlooked. So in the communal spirit, Analog5, an ezine made by the Hak5 community, for the Hak5 community takes its shape here in this form.

Ezines seem to be lost in the heyday of independent movies, podcasts and IPTV.

Whether or not you've been a supporter of Analog5 since the idea was first brought up, or if you just heard about this ten minutes ago, you might not know the story of how it came up and where its all going, so without further ado, on with the story…

Back in early November, 2006, I was reading through some old ezines from the days of BBS up through present time because of the interest I have in the subject.

Whether or not an ezine was successful or not, whether or not it had more than one issue, whether or not people liked it, it leaves its own unique impression in one way or another. Simply the idea that these publications could gather information from numerous sources and then release it freely to the public without any distribution expenses at all seemed like a really easy way to get what you want out there. A model that would change and evolve over and over again for many mediums.

So its early November, at this point, after reading some zines, I'm drifting in and out of conversations in IRC trying to figure out something to do with my time when it hits me, create a Hak5 ezine. I was excited with it, the few people I bounced it off of (one who I believe to have been Moonlit) were excited with it, so I ran it by the development community who accepted it with open arms. A few days later, I released my idea to the complete public and waited for the submissions to roll in. Here I sit about two months later with eleven articles under my belt and a fantastic group of followers always there to hear out my ideas and make suggestions for improvement. The magazine could not have made it this far without a single one of them.

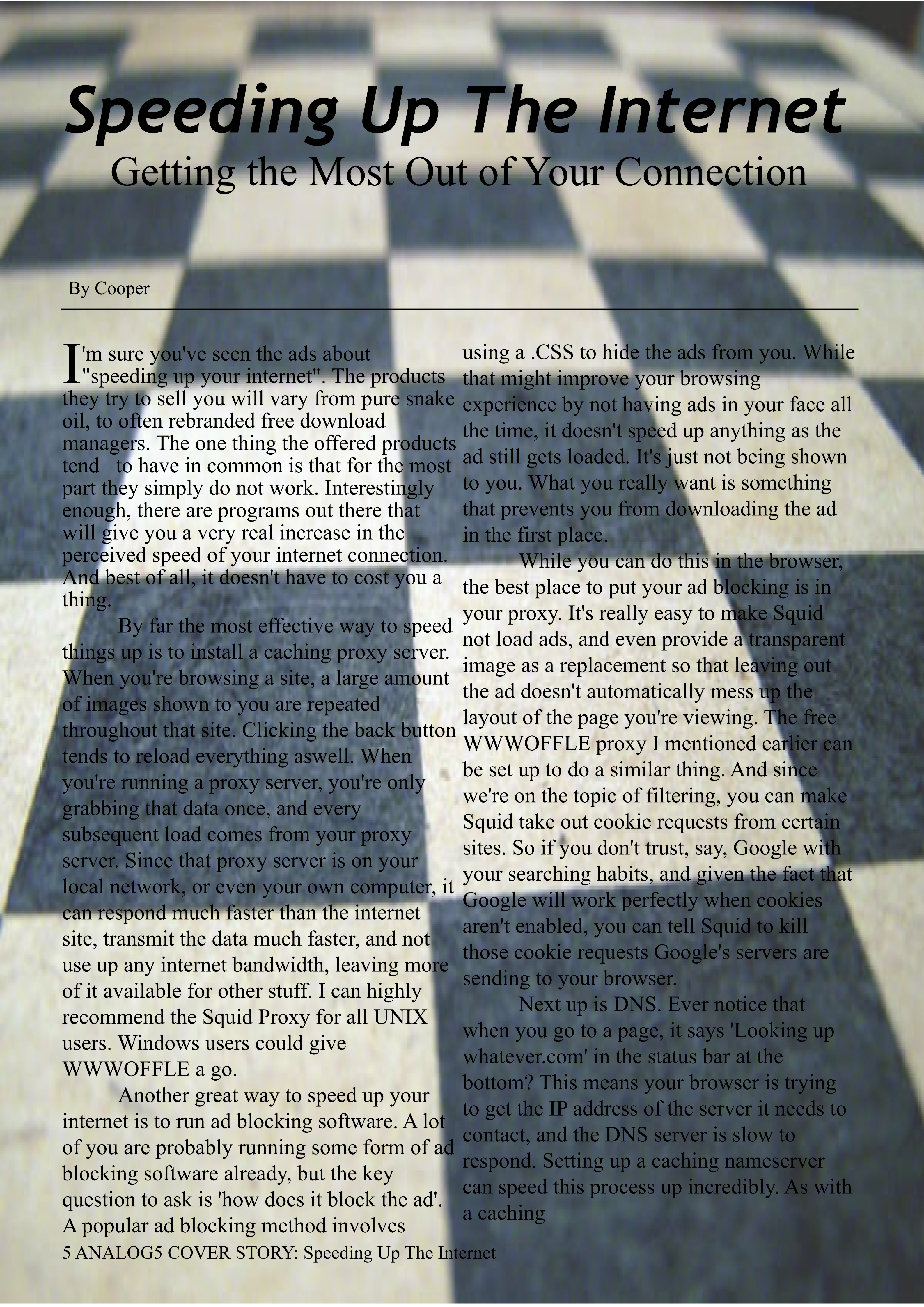
How it looks, what form its in, and if I'm still doing anything for it is uncertain, but as long as someone wants to create content for it, it will keep going

In the matter of where this zine is going, the best way to have that question answered would be by using the famous "Where do you see it in five years?" approach. Answering honestly, I see it surviving at least that far. How it looks, what form its in, and if I'm still doing anything for it is uncertain, but as long as someone wants to create content for it, it will keep going. Offering information by Hackers, Gamers, Geeks, Nerds, Phreakers, SysAdmins and everyone else to create a genre almost irresistible.

I think a lowdown of all of the functions of the magazine deserve to be discussed, so I'll cover everything I have decided on thus far. This ezine will be distributed though a variety of means. Http downloads, its own wiki page, and hopefully a few other methods, perhaps even through a Hak5 BBS. If you would like to mirror this somewhere, be it on your website, your BBS, an FTP server, you may freely do so due to the fact that Analog5 is licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 2.5 license. I would just ask you to notify me in some fashion so I could perhaps recognize you in future publications, and see how far Analog5 is reaching out. Depending on how many articles I get from you the submitters, I would like to release every month on the 20th to be spaced out with the monthly Hak5 releases.

If you would like to send in an article, feel free to drop it at Analog5@live.com and it will most likely instantly be put into the next distribution. For a list of submission rules and guidelines, I'd like to point you at http://hak5.org/wiki/Analog5 for any questions you may have.

In all honesty, those are the only two things completely solid about Analog5 so far. Its always changing and growing to be what the community wants it to be. Because of this, I'm also completely open to all suggestions. Whether you love this issue, or hate it, I'd like to hear from you. I hope you enjoy this issue of Analog5 and I believe that's all for this article, so its about time for you to get on with the rest of the ezine.



Speeding Up The Internet

Getting the Most Out of Your Connection

By Cooper

I'm sure you've seen the ads about "speeding up your internet". The products they try to sell you will vary from pure snake oil, to often rebranded free download managers. The one thing the offered products tend to have in common is that for the most part they simply do not work. Interestingly enough, there are programs out there that will give you a very real increase in the perceived speed of your internet connection. And best of all, it doesn't have to cost you a thing.

By far the most effective way to speed things up is to install a caching proxy server. When you're browsing a site, a large amount of images shown to you are repeated throughout that site. Clicking the back button tends to reload everything aswell. When you're running a proxy server, you're only grabbing that data once, and every subsequent load comes from your proxy server. Since that proxy server is on your local network, or even your own computer, it can respond much faster than the internet site, transmit the data much faster, and not use up any internet bandwidth, leaving more of it available for other stuff. I can highly recommend the Squid Proxy for all UNIX users. Windows users could give WWWOFFLE a go.

Another great way to speed up your internet is to run ad blocking software. A lot of you are probably running some form of ad blocking software already, but the key question to ask is 'how does it block the ad'. A popular ad blocking method involves using a .CSS to hide the ads from you. While that might improve your browsing experience by not having ads in your face all the time, it doesn't speed up anything as the ad still gets loaded. It's just not being shown to you. What you really want is something that prevents you from downloading the ad in the first place.

While you can do this in the browser, the best place to put your ad blocking is in your proxy. It's really easy to make Squid not load ads, and even provide a transparent image as a replacement so that leaving out the ad doesn't automatically mess up the layout of the page you're viewing. The free WWWOFFLE proxy I mentioned earlier can be set up to do a similar thing. And since we're on the topic of filtering, you can make Squid take out cookie requests from certain sites. So if you don't trust, say, Google with your searching habits, and given the fact that Google will work perfectly when cookies aren't enabled, you can tell Squid to kill those cookie requests Google's servers are sending to your browser.

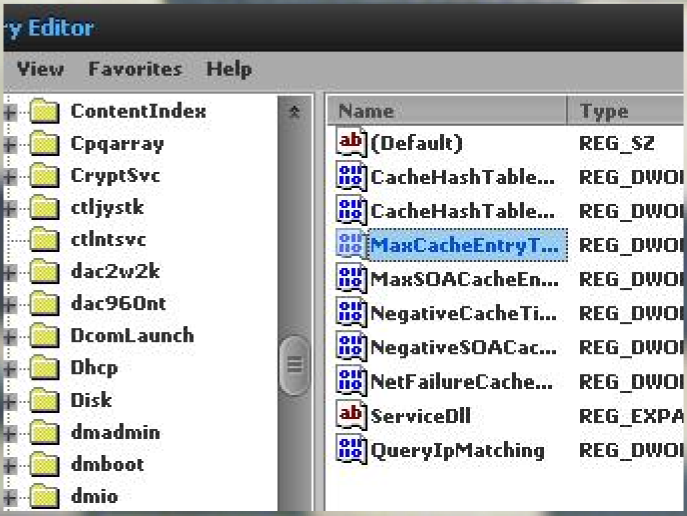
Next up is DNS. Ever notice that when you go to a page, it says 'Looking up whatever.com' in the status bar at the bottom? This means your browser is trying to get the IP address of the server it needs to contact, and the DNS server is slow to respond. Setting up a caching nameserver can speed this process up incredibly. As with a caching proxy server, a caching nameserver will forward your DNS request to your ISPs DNS server for the first lookup, and cache the response. Any subsequent requests for that domain name will be served from the cache. It won't make much of a dent in the bandwidth you use because these DNS requests use very little data, but sites will appear to respond much faster. Unix users will probably want to use either BIND or DJBDNS.

Those who run Windows XP or newer don't have to do anything. Windows caches all DNS responses for 24 hours by default. This however is a bit much. Most DNS servers on the internet update their data every 4 hours these days,so you might want to shorten these caching times. To do this, open regedit and find the entry:

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\DNSCache\Parameters

There should be a MaxCacheEntryTtlLimit key there. If not, add it as a numerical value. The value is the caching time in seconds, so for 4 hours that would be 14400.

Another key you'll want to change is the NegativeCacheTime key. Windows by default not only caches succesful DNS lookups, it also caches failures so any temporary DNS problems automatically will last 4 hours on your system due to this caching. To turn off the caching of the DNS failures, set the NegativeCacheTime to 0. While you're in there, add a key named QueryIpMatching with a value of 1. By default, Windows will accept and cache any DNS response it receives, including those from machines that it never even sent a DNS query to. This is an obvious security issue with Windows. This setting will make Windows accept and cache only DNS responses from servers it queried.



ContentIndex
Cpqarray
CryptSvc
ctljystk
ctlntsvc
dac2w2k
dac960nt
DcomLaunch
Dhcp
Disk
dmadmin
dmboot
dmio

Name	Type
(Default)	REG_SZ
CacheHashTable...	REG_DWOI
CacheHashTable...	REG_DWOI
MaxCacheEntryT...	REG_DWOI
MaxSOACacheEn...	REG_DWOI
NegativeCacheTi...	REG_DWOI
NegativeSOACac...	REG_DWOI
NetFailureCache...	REG_DWOI
ServiceDll	REG_EXPA
QueryIpMatching	REG_DWOI

There's one last thing you can do, but this is pretty advanced. Using a Quality of Service, or QoS package you can do traffic shaping on your network. Many of you will know and hate QoS software because it's typically used to slow down bittorrent downloads, or othr similarly useful services your network provider doesn't approve of. However, it can also be used to give you decent throughput while browsing, a very low ping when gaming, and acceptable speeds on your bittorrent traffic. It does this by prioritizing certain types of traffic, and limiting other types to prevent them from drowning out the rest. On top of this it can be used to prevent the normally rather excessive buffering that's performed by your modem or router. As this really is for the more hardcore networks guy, I'm going to skip that subject for this article. If you're really interested though, read the Linux Advanced Routing and Traffic Control Howto. I've yet to find a free QoS package for Windows, so for the time being your Windows guys are sadly out of luck.

DNS Registry Settings
http://www.microsoft.com/technet/prodtechnol/winxppro/reskit/c24621675.mspx

Howto
http://lartc.org/lartc.html

The Hacker's Manifesto 2.0

By TomB (www.ownthebox.net)

NOTE: Do not take seriously. This is meant as a joke.

Another one got spammed today, it's all over the blogs. "Nigerian dies, and the will money is handed out".

Damn tubes. They're all full.

But did you, in your three-piece suit and 2000s cubicle, ever take a look behind the eyes of the Counter-Strike n00b? Did you ever wonder what made him pwn, what forces aimed his gun, what may have molded him?

I am a pwnerer, enter my map...

Mine is a map that begins with hostages... I'm faster than most of the other CTs, this crap they speak bores me.

Damn trucks. They're all slow.

I'm in my basement. I've listened to podcasts explain for the fifthteen time how to hack an Xbox. I understand it. "No, Mr Dvorak, I didn't upgrade to Windows Vista, I installed Linux."

Damn ISPs capping connections.They're all alike.

I made a discovery today. I found the outside world. Wait a second, this is different. It does what the government tells it to. If it makes a mistake, it's because religion fucked it up. Or the US government finds oil in the middle east again. Or thinks the voting machines are broken.

Damn government. All they do is play wargames. They're all stupid.

And then it happened... the tubes opened... packets rushing through the phone line like water through an internet connection. A VoIP pulse is dialled out, an ACK packet is sent in. "This is it, this is where I belong... MySpace dot com.", "I know everyone here... even if I've never met them, if I ever met them, they would be pedophiles from another city". "I know you all..."

Damn podcasters. Tying up the tubes again. They're all alike.

You bet your micro we're all alike... we've been spoofing TV shows since you were playing Half-Life 1. We've been dominated by Rev3, or ignored by Apple. The few that had something to teach found us idling in #Hak5 on IRC.

This is our internet now... not some nuetral bullshit, the beauty of the packet. We make use of a tracker already existing without paying for what is overpriced. We click links.. and you call us criminals. We seek free music... and you call us criminals. We exist without girlfriends, without lives, without normal jobs. You hunt Afghans, destroy trade centers, and wage war against terror. You cheat into office, and make us believe it's for our own good.

Yes I am a pirate. My crime is that of piracy. My crime is that of stealing MP3 files. My crime is that of filming movies in the movie theatre. My crime is that of outsmarting Einstein.

I am an IPTV presenter, and this is my bullshit. You may stop this IPTV show, but you can't stop Rev3... after all, they run Digg.

Windows XP Security and Protection

By ChevronX (www.chevronx.com)

Windows Update

One of the first things to help secure your computer is to use Windows Update, which is an online service supplied by Microsoft to help secure your operating system against the latest exploits!

Open Internet Explorer and click Tools, then Windows Update.

If you are offline, it will ask you connect to the internet so if your on dialup then enter in your Username and Password and click connect. When you first go there it will ask you to install special Windows Update plug-ins for Internet Explorer to allow you to use the Windows Update website efficiently, so go ahead and click Ok.

Then it will redirect you to a page, asking whether you want to do a Custom or Express install, Click Express it is by far the easiest, Custom allows you to remove and add things to the update, such as other tools Microsoft has provided, and Express already has the most critical patches selected.

A dialog box should pop up, and you just click Install to have Windows Update download and install the latest patches for your system. If an update wishes you to install it alone, and needs a restart, then click Ok and restart the computer once it tells you too after it has installed the patch and do the previous to download the rest of the patches.

Automatic Update

With Service Pack 2 (which can be obtained through Windows Update if you don't already have it) of Windows XP, Microsoft perfected the ability for your computer to download the latest updates automatically, while your computer is connected to the computer. I will show you how to configure Automatic update the easy way.

Right click My Computer (Either on the desktop or the Start Menu) and click Properties.
Click the Automatic Updates tab.

Here it gives several options, you get to Turn it Off/On, Schedule the updates so it downloads the updates when you wont be using the computer, Just notify you that updates are available, or Download the updates and choose when you want to install them. Before the updates take affect, you need to shut your computer down.

Windows Firewall

Another thing to make sure you have activated is the Windows Firewall, which is dramatically improved with Service Pack 2. A firewall is an application whose aim is to block access to your computer from the outside, such as Crackers and others trying to gain access to your system. It is also useful to block applications from access the Internet, especially if you don't know what the program is, it could be an application that has gained access to your computer and is about to send all your email and documents to its creator.

Generally Windows Firewall is sufficient to use, and doesn't require you to download any third party software and any patches that are needed are rolled out in Windows Update.

Viruses and Spyware

Before we do anything else, I want to point out the difference between Viruses and Spyware. Viruses are created with malicious intentions, such as destroying all your data, or better yet opening it out so all the world can read that CV of yours.

Viruses still have a big impact on the world today, as its ability to spread itself through the internet has cause a lot of hassles. Fortunately the updated Virus scanner can do wonders to keep your computer hassle free. Spyware is acutally advertising to hijack your Internet Explorer homepages. It also has a major impact on the performance of the computer, by slowing it down quite a lot. Even 30 infections can sometimes do the damage of 30,000. Luckily for us, there are utilities to protect our computers from both.

Virus Protection

There are many applications for Virus protection out there, but the ones I recommend and are sure are legit and not Spyware or viruses in itself are AVG Free Antivirus, and an excellent NOD32.

I do not recommend Norton Antivirus or Norton Internet Security at all because it causes a lot of problems, with dramatically slowing your computer down, etc and it doesn't offer the excellent protection that even the free AVG Antivirus does.

AVG Antivirus

Once it has been downloaded, Run the file and it should launch the AVG Setup. Most of the steps are just there for if you want to customise it, I recommend keeping it at default, so click Next, right through till you get to Finish and then click that. AVG should now be running, in the System tray (Where the clock is), and AVG Settings wizard will come up.

If for some reason AVG does not appear in the System Tray, then restart your computer. To update AVG, right click on the dull AVG icon in the system tray and choose, Update, Internet it will get updates form the internet and installs them.

If you needed to install updates manually, such as if you have the update file on a floppy disk or CD (u$$$$.bin). Then you choose manual and point AVG to where the update it stored. It should then say it found an update and install it for you and that AVG icon should become nice and colourful to indicate it is up to date.

AVG when fully updated, will automatically scan incoming files and emails for viruses but you cannot be too careful. To run a scan, right click the AVG System Tray Icon and click Test Center. It will open the Test Center, click Complete Scan and it will begin the Virus scan of your computer. Just a reminder to update the antivirus first for maximum protection.

Spyware Protection

I know of two great spyware scanning software, Adaware SE, and Spybot Search & Destroy. Both are excellent and free, and when used together can remove most if not all that your system can be infected by.

Lavasoft Adaware SE Personal Edition

Once you have downloaded the setup file, and then run it and it will launch the setup. It is again, simple to install, by clicking next or I agree to the dialog prompts, till you reach the end with check boxes. Untick all checkboxes to avoid confusion and click Finish.

The Adaware SE icon should now be on the Desktop, double click it. It will tell you that your definitions are old, so you know what you are going to download; definitions are updates with the latest list of Spyware and the ability to remove them. Click Connect to allow it to connect to the internet and download and install the latest definition. Once it has downloaded and installed the definition file, it will take you to the Adaware SE main screen. You have to scan for Spyware, it won't do it automatically for you, so I recommend letting it scan during the night. To scan click Scan Now, make sure Perform Full Scan (You can choose Smart System Scan to scan for Spyware in the most logical places where it usually resides, but if you want full effect, go Perform Full Scan), then click Next and it will begin scanning.

Once it has finished scanning, it will display its results, to remove the Spyware from your computer Right click a result, for example "Tracking Cookie" and select Select All Objects, to select all the Spyware the scan found on your computer. Click Next, a dialog window will come up, asking if your sure you want to delete the Spyware, select Ok, and then it will remove the Spyware, and then take you back to the main Adaware Screen.

There may be Spyware that cannot be removed, because it is currently protected, and it will then show a dialog asking whether you want Adaware to run at next restart. Click Yes, so when your computer restarts it will run Adaware before the Spyware has a chance of protecting itself. Click Scan Again and it will scan and be able to remove the Spyware.

Spybot Search & Destory

Spybot Search and Destroy another great Spyware scanner software and ran with Adaware it is usually picks up a few that Adaware hasn't found.

Double Click the setup file to launch the installation.

Just follow the prompts and installing is a breeze. Once it is installed, run it. If you have Adaware on your computer then

Spybot will detect this and bring up another dialog. Just click your way through that. Updating Spybot is easy. At the main spybot screen, there is a Search for Updates button, click it. Spybot will then reach out to the internet and show you a list of the latest updates it can download, you don't need the help ones, or tea timer but select the rest and click Download Updates. It downloads and installs the updates. To scan, click Check for Problems, it will then scan and then display the results. To remove Spyware, click Fix Selected Problems. All the results will automatically be selected.

Just a reminder to update your Spyware database before your scan so you can make sure you don't miss any! There are other Virus and Spyware software out there, even Microsoft has released an application called: Microsoft Windows AntiSpyware.

Links

Windows Update	http://www.windowsupdate.com
AVG	http://free.grisoft.com
Ad-Aware	http://www.lavasoftusa.com
Spybot	http://www.safernetworking.org

Hak5 Radio

How does he do it?

By Mubix (www.mubix.com, mubix.vox.com)

Well, I am finally putting it out there, the down and dirty of how Hak.5 Radio comes to life. In the following paragraphs I will tell you how I get the Song Request site up as well as the back end server portion. Check it out.

I'm going to start with the server build, because without that, you are just talking into a microphone that goes nowhere. Okay, so we need a server. A Linux one will do, whether it is hosted by a provider, such as Hak.5 Radio's is, or you can host it at home. Either way, a stable, always on, Linux machine is needed.

Cool. We have a server and it's running Linux. Next we get to pick a streaming service daemon. Now, the two widely used are Shoutcast and Icecast. Both are great service daemons and have so many configuration settings that they are flexible from 5 listeners to 5000 with zero growing pains. But, what is going to make me choose one after the other you say? Well, you can google a blow for blow comparison, but this is how I see it: Shoutcast has a listening service that you automatically get posted on which will bring in quite a few listeners of "off the street" if you will. Icecast on the other hand, has smaller delay on your stream. This is important when taking calls on live shows. Icecast has a 3-8 second delay from the time it hits the microphone to the time the listener hears it. Compared to Shoutcast's 30-40 second delay, it makes for a tough decision for any new radio station.

Server, check! Service daemon, check! Wait…. You mean I have to configure it? Yup, the cruel truth of Linux is you have to configure everything. For both Shoutcast and Icecast, the manuals, FAQs and Forums are undeniably better than anything that will ever fit into an article for any eZine/magazine. Thus, I will leave the configuration up to you. I will go over a couple things to look into in the configuration during other sections of this article.

Now that it is configured, we want to make sure that it stays up. There are a couple of ways to do this. The easiest is to make a cron job (scheduled task) to check if the server is running and to run it if it is found not to be running. The following script does just that:

```
#!/bin/bash

if [ ! "$(ps ax | grep icecast | grep -v grep)" ]
then
  screen -d -m icecast -c ~/icecast/icecast.xml
fi
```

Save the file as "checkice.sh" and run chmod +755 to make the file executable. Run crontab -e and put `*/15 * * * * /path/to/file/checkice.sh >/dev/null` in or, mv the file into one of your distro's `/etc/cron.(hourly/daily/weekly)` folders. Either way you will be making sure that your server stays afloat. Now, you might notice that I used the command 'screen'. Screen is a way of running things on a virtual console. Basically it's a "insta-daemon" program. I will have links to everything I go over at the bottom of this article.

Ok, we are all done on the server side. But what does all that hard work do? Absolutely nothing without a client or 'streaming software package' to send the music or video to the server for people to get it. There are a limited number of programs that can do this. SAM Broadcaster is the most widely used, but costs money.NSVTools w/ NSVCapture is used for video, and I'll go into that a bit later, but it is free. WinAMP with Oddcast is also a free alternative. I actually use all 3 of these solutions to run Hak.5 Radio.

I use SAM Broadcaster when ever I want to "go live", where I will be DJing myself, queueing up songs and using the mic. WinAMP with Oddcast can do this just fine, and is a awesome alternative, SAM just has a better interface and more options. I use WinAMP w/ Oddcast and another plug-in for WinAMP called "Song Requester". Guess what it does? Yup! Gold star for you. It auto-magically generates a website of all the songs in your play list and allows Internet users to request songs by searching for a song and clicking on it. You can also limit how many songs a single user (by IP) can request at one time. The default is 3, but on a low traffic station, I have found that 5-10 works better.

The last trick to the radio station is the ability for listeners to upload songs that they want to hear. For that I use a configured HFS server on the "AutoDJ" box that has a virtual folder that allows the upload of mp3s, waves and other such audio/video formatted files. WANQer is yet another plug-in that watches that folder for new songs in the folder and adds them to the play list. The only problem is Song Requester doesn't update it's cache auto-magically. I had to set up a scheduled task to push the update cache button for every 5 minutes to get around this obstacle.

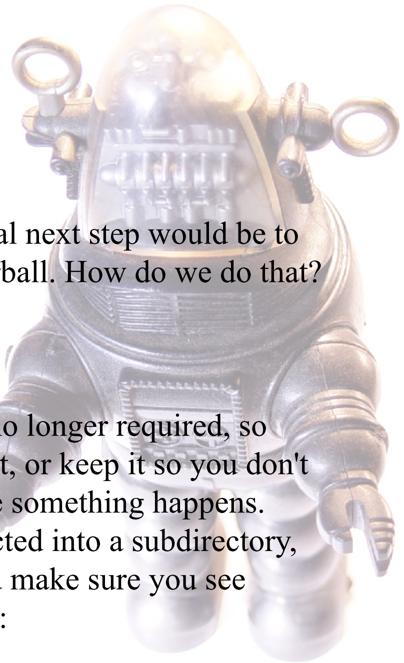
Other than that, that is the basics of how Hak.5 Radio works. There are a lot of little tweaks and specific configurations that will be unique to how you set up your radio station but this article should be a huge bump in the right direction.

LINKS

Shoutcast	http://www.shoutcast.com/
Icecast	http://www.icecast.org/
Intro to Cron	http://www.unixgeeks.org/security/newbie/unix/cron-1.html
Screen@ Wikipedia	http://en.wikipedia.org/wiki/GNU_Screen
SAM Broadcaster	http://www.spacialaudio.com/products/sambroadcaster/
WinAMP	http://www.winamp.com/
Oddcast	http://www.oddsock.org/tools/
NSVTools	http://www.nullsoft.com/nsv/
SongRequester	http://www.oddsock.org/tools/gen_songrequester/
HFS	http://www.rejetto.com/hfs/
WANQer	http://www.winamp.com/plugins/details.php?id=99396

Setting up an IRC Bot

By TechCentric-Nick (www.techcentric.org)



Setting up IRC bots nowadays isn't the hassle it used to be. Early on, you had to be able to code a bot from scratch, including adding in support for anything and everything you wanted to do. Now, there are several excellent bot frameworks to base bots off of, limiting the amount of work to be done while maximizing its power and capabilities, and streamlining it into nearly a 1-2-3 process. The point of this article is to walk a newbie user through setting up an IRC bot with the goal of adding external capabilities to it in the future. We're going to be using Eggdrop in the tutorial, as it is by far the most common framework.

Prerequisites

Now, Eggdrop is first and foremost a Unix-based bot. While some people have success using Cygwin and other tools to run it under other operating systems, it would be recommended that for the sake of this tutorial you have a Unix shell account or a local Unix box to be practicing and/or running the bot from. For the purposes of this demonstration, I will be demonstrating using an i686 machine running Debian, commonly known to most as PhreakBox.

Whatever you're using, you're also going to need the following applications: tar, gzip, and gcc. I can't explicitly say how to get them because there are too many variables involved like distro and installation conditions, but check with any repositories your distribution may have if the tools aren't installed by default, or substitute others where necessary.

Getting the Bot Up

Now, logged into your shell space, download (or upload, if you're moving the package from another machine) the Eggdrop source package from Eggheads.org to the machine.

```
wget
ftp://ftp.eggheads.org/pub/eggdrop/source/1.6/eggdr
op1.6.18.tar.gz
```

[NOTE: This is all on one line wih no spaces]

Once this is done, the natural next step would be to extract the files from the tarball. How do we do that?

```
tar xzf eggdrop1.6.18.tar.gz
```

At this point, the tarball is no longer required, so you can delete it if you want, or keep it so you don't have to re-download in case something happens. Your tarball probably extracted into a subdirectory, so cd into that directory and make sure you see something along these lines:

```
nick@phreakbox:~/analogbot/eggdrop1.6.18$ ls
aclocal.m4    configure.ac
ChangeLog     CONTENTS
config.h.in   COPYING
configure     disabled_modules
```

The next step in prepping your bot for use is configuring your compile. Do this by running ./configure at your shell prompt. You should see a crapload of status lines and stuff fly by, and it can take a while to run through. The best advice at this point is to get up and pour yourself a drink.

Assuming all went well, you should be ready to compile Eggdrop. If something went wrong, the configure script will let you know, and using this information you can seek help out. In this test case, everything worked out fine. So, we can now run make.

If you're anticipating everything to work out alright, you can kill two birds with one stone as I do below. If you're new to this or think you may run into a problem...separate the commands and run them separately. Note that this is not recommended, and definitely not common practice, and should be done with caution.

```
make config && make && make install
```

[NOTE: This is all on one line wih no spaces]

Again, your compile might take upwards of several minutes. If you're already in need of a refill on that drink you just got, you should probably go get it now. You will probably see a few warning messages mixed in; this is normal, and everything will sort itself out. If there's anything that really ends up a problem, you'll be notified and it won't compile any further. Once compiling is completed, give your Eggdrop a test run..

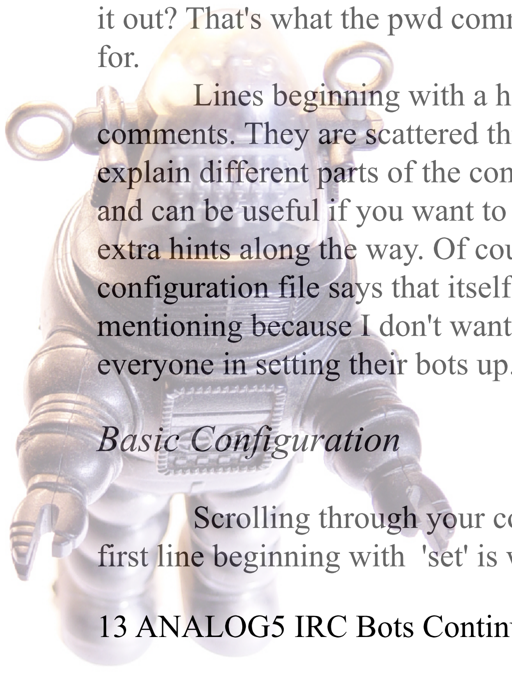
```
./eggdrop

        Eggdrop v1.6.18 (C) 1997 Robey Pointer (C) 2006 Eggheads
        [15:13] --- Loading eggdrop v1.6.18 (Mon Jan  1 2007)
        [15:13] * Please make sure you edit your config file completely.
```

Wait a minute, why did I just have you do that if we're not done configuring? This was simply to test that your compiled copy was working. Assuming you saw that...you're in business. If not, try recompiling, try deleting everything and redoing this tutorial from scratch. Failing that, ask for help, and if nobody helps you (not my problem), you might as well call yourself an emo and play in traffic.

Configuring Your Bot

Now we can go through and edit your configuration file for getting the bot online. Open up your favorite text editor, be it Kate, Pico/Nano, Vi, Emacs, or whatever you have, and open up eggdrop.conf.

The first line, which looks like a CGI shebang line, is a request for you to fill in the complete path to your bot. I'd show it, but some idiot is bound to use my path in their script and then complain that it doesn't work (and it would also be a potential security hazard, something I won't have). I'm not interested in people doing that, so I'll make it fair; everyone has to punch in their own. Can't figure it out? That's what the pwd command at the shell is for.



Lines beginning with a hash mark (#) are comments. They are scattered throughout the file to explain different parts of the configuration to you, and can be useful if you want to leave yourself some extra hints along the way. Of course, the configuration file says that itself, but it's worth mentioning because I don't want to have to baby everyone in setting their bots up.

Basic Configuration

Scrolling through your configuration file, the first line beginning with 'set' is where we need to start. 'set username' sets the username that appears in the IRC bot's hostmask. If you want to leave it at the default (not recommended), you can, but it would be wise to change it. Right below it is the admin line; fill that with your information following the pattern outlined in the default.

I believe it is safe to assume you're not going to be networking your bots anytime soon, so we can safely leave the next set line alone. You can set the timezone if you want to, but it's not required (only useful for logging and some scripts, which we will get to momentarily).

Most home machines will have one IP to bind to, so you can scroll down past that too. (In my case, I have to configure it.) However, you probably also noticed that some of the configuration options at this point are commented too. Keep that in mind as you scroll through the file, and notice that if you want to change some of the settings as you scroll through the file.

The next two major points of interest are 'set userfile' and 'set pidfile' (and further down, 'set chanfile'). These can be left alone if you're on a single-user machine, but if you have the remote chance of being on a shared machine (most likely for shell accounts or anything that isn't a personal box), change them to something unique, perhaps 'pid.yourname' for clarity.

Now, Eggdrop features some special options as well, such as administration through DCC and telnet. I just want to take the time to point out that if you screw up configuring these, they can be a potential security hazard. Read all the comments before you make choices, and NEVER KEEP DEFAULT
SETTINGS!

Continuing on, 'set owner' is remarked but required. Unremark it (remove the #), and set it to your IRC nickname. In the Channel section, read the comments about how to set up your bot to join channels, and make sure when you set modes for each channel you keep in mind what they do.

'set nick' and 'set altnick' define the bot's IRC nickname, set these to whatever you want. Change 'set realname' while you're at it too; this is a tipoff of a poorly configured bot.

Setting servers up is self-explanatory; just follow the patterns provided. Keep in mind that if you leave these set to defaults, you might be wondering why your bot doesn't show up in your channel.

For the sake of keeping this article short, I won't continue with configuration settings; the comments above them do a well-enough job of helping you decide what to set them to, follow those and use common sense. Just keep in mind that there are two "kill lines" embedded in the configuration file; these begin with 'die' and are placed to make you read through it. Remark them or delete the line. (Remember how the bot died when we tried starting it after we compiled it, and how it said we needed to edit the configuration file? It was one of the "kill lines" that did that...not the bot application itself.) Oh, and 'set modpath'should probably be set to null because you compiled this fresh - but it depends on how you set things up, again.

Getting Online

Once the bot is configured (try a test run again, only this time the bot should be saying something along the lines of "User File Not Found". If so, good, run './eggdrop -mn'. If not, go fix your configuration file; Eggdrop is nice enough to point out the problem for you.

Back on IRC, go to the channel where the bot is set up to go to, and wait for it to appear. When it does, send a private message to it with 'hello' (or if you changed it, the new keyword) so it recognizes you.

In the console window, you should see a line like the following appear as you get some PMs and notices from the bot:

```
[16:03] Bot installation complete, first master is
<yournamehere>
```

Respond to the PMs using the commands the bot gives, such as setting a user password.

Congratulations, the bot is now set up. Send a SigKill to the bot, and start it with just ./eggdrop. It should join back to the channel.

In future articles, we can discuss adding new capabilities to your bot to make it useful. But for now, bask in your success, go get (yet another) refill of your drink of choice, and stare at the bot in your channel's userlist.

Final Disclaimer

I'm installing Eggdrop 1.6.18, the most recent release at the time of writing. These steps may differ as new versions are released, and may not match your individual setup. I'm not responsible.

WARCHALKiNG
The Lost Art

Famicoman wuz here

hackinacansiteled.com

Warchalking is an activity made to go along with the act of Wardriving. For those of you who don't know what wardriving is, its simply just going around with something like a laptop or PDA and looking for WiFi networks in your area. There are numerous articles and publications on wardriving, so I won't delve into them here.

Warchalking originated in the Summer of 2002 by Matt Jones and according to him, it was modeled after hobo signs. Hobo signs are marks that the homeless make to show that certain houses will give you a glass of milk, or another house will throw dogs for you or etc. Basically, you wouldn't understand the signs unless you were a hobo, and thus, you wouldn't understand warchalking signs unless you were a fellow warchalker.

Sadly, warchalking never caught on, mostly due to the fact that it was instantly made commercial by companies. Nowadays, you never even hear people talking about warchalking, or see a mark when you glance down at the sidewalk. Even though this may be, warchalking is still pretty unique in the fact you mark wifi hotspots in public with chalk. There are three warchalking symbols...

The first is the open node symbol. This looks like two parenthesis back to back with the ssid displayed overhead, and the bandwidth displayed below. Example:

linksys
)(
1.5

This sign above signals that there is an open node with the ssid, "linksys", and has a 1.5MB connection speed. The second symbol is that of the closed node. The closed note is simply displayed as a circle with the ssid put above it. Example:

Cisco
()

The warchalking sign above indicates there is a closed node broadcasting the ssid, "Cisco".

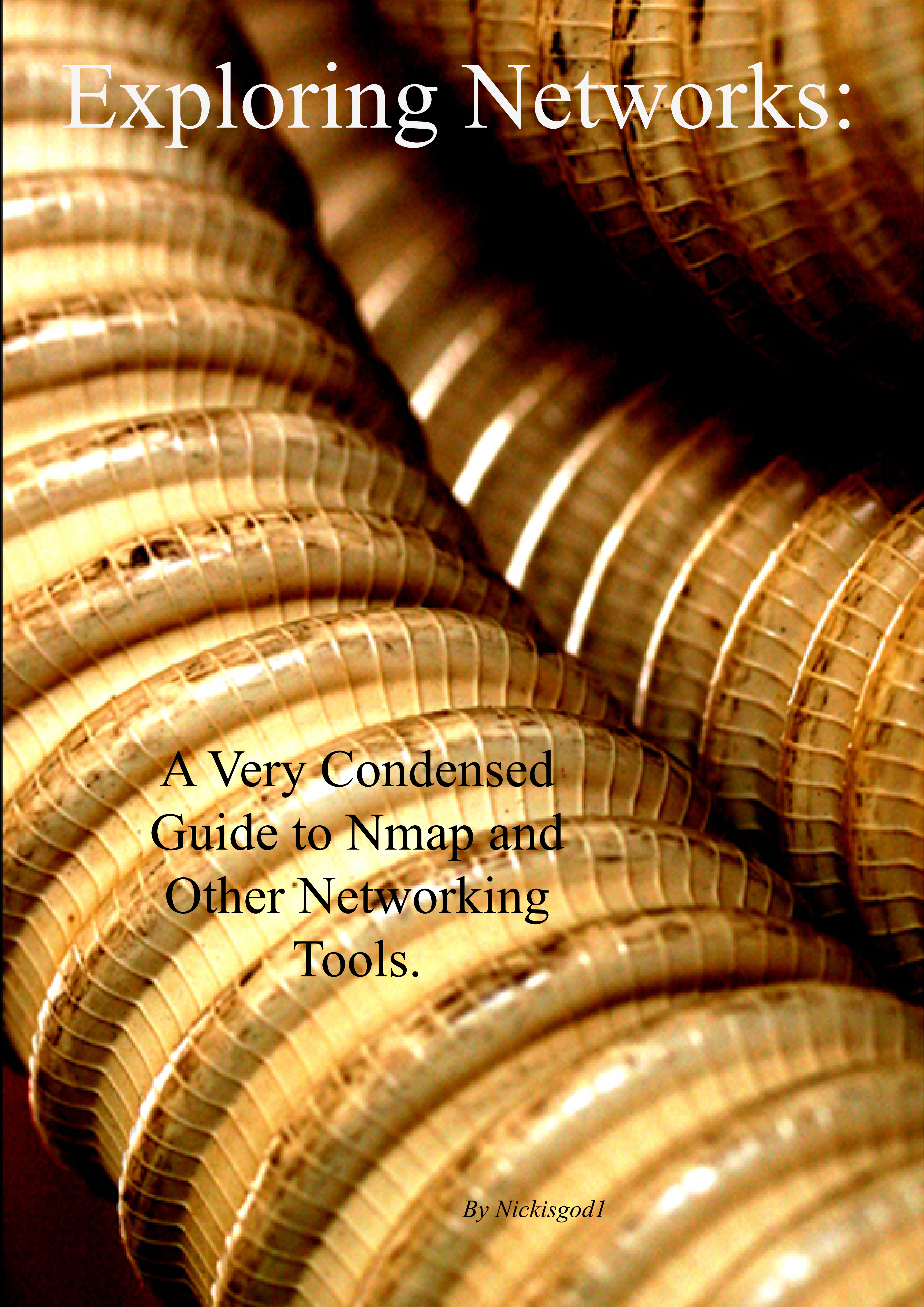
The third and final warchalking signal is for a WEP encrypted node. This is represented with a circle that has a "W" encased in it. Under the circle is the bandwidth, on the top left corner is the ssid, while the contact is in the top right corner. The contact, however is usually unknown. Example:

donttakemywireless
(w)
2.0

The above network is broadcasting the ssid, "donttakemywireless", while the node is WEP encrypted, and the connection is at 2.0MB/s.

If you have a laptop, and a stick of chalk, you can go around and start warchalking. Its that easy. Get a group of friends together, and start a warchalking club. Here is the url of a printable flier that contains a cutable wallet sized chalking mini-book.

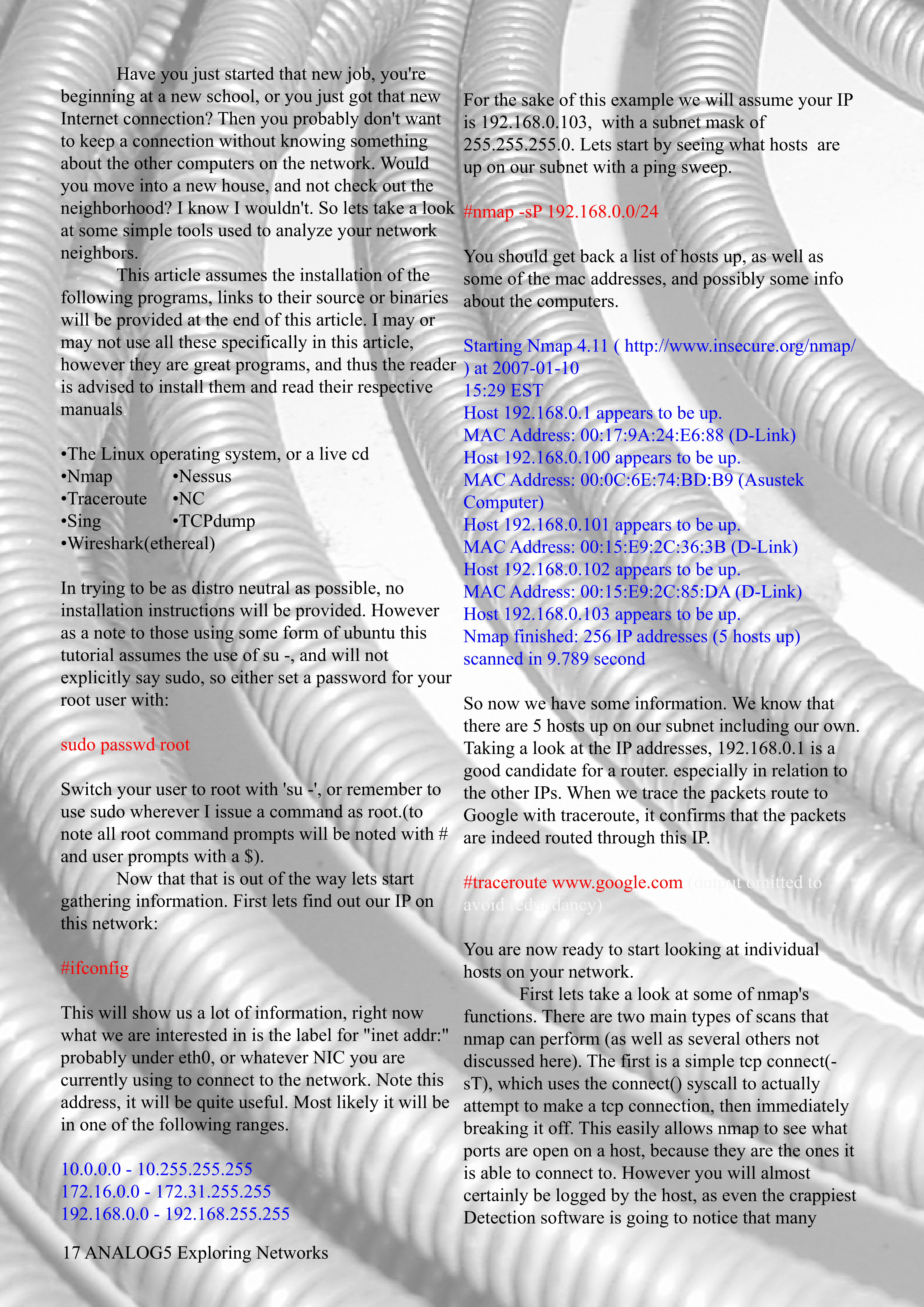
http://hackinacan.sitesled.com/Warchalking/warchalking-symbols.gif



Exploring Networks:

A Very Condensed Guide to Nmap and Other Networking Tools.

By Nickisgod1



Have you just started that new job, you're beginning at a new school, or you just got that new Internet connection? Then you probably don't want to keep a connection without knowing something about the other computers on the network. Would you move into a new house, and not check out the neighborhood? I know I wouldn't. So lets take a look at some simple tools used to analyze your network neighbors.

This article assumes the installation of the following programs, links to their source or binaries will be provided at the end of this article. I may or may not use all these specifically in this article, however they are great programs, and thus the reader is advised to install them and read their respective manuals

- The Linux operating system, or a live cd
- Nmap
- Traceroute
- Sing
- Wireshark(ethereal)
- Nessus
- NC
- TCPdump

In trying to be as distro neutral as possible, no installation instructions will be provided. However as a note to those using some form of ubuntu this tutorial assumes the use of su -, and will not explicitly say sudo, so either set a password for your root user with:

```
sudo passwd root
```

Switch your user to root with 'su -', or remember to use sudo wherever I issue a command as root.(to note all root command prompts will be noted with # and user prompts with a $).

Now that that is out of the way lets start gathering information. First lets find out our IP on this network:

```
#ifconfig
```

This will show us a lot of information, right now what we are interested in is the label for "inet addr:" probably under eth0, or whatever NIC you are currently using to connect to the network. Note this address, it will be quite useful. Most likely it will be in one of the following ranges.

```
10.0.0.0 - 10.255.255.255
172.16.0.0 - 172.31.255.255
192.168.0.0 - 192.168.255.255
```

For the sake of this example we will assume your IP is 192.168.0.103, with a subnet mask of 255.255.255.0. Lets start by seeing what hosts are up on our subnet with a ping sweep.

```
#nmap -sP 192.168.0.0/24
```

You should get back a list of hosts up, as well as some of the mac addresses, and possibly some info about the computers.

```
Starting Nmap 4.11 ( http://www.insecure.org/nmap/
) at 2007-01-10
15:29 EST
Host 192.168.0.1 appears to be up.
MAC Address: 00:17:9A:24:E6:88 (D-Link)
Host 192.168.0.100 appears to be up.
MAC Address: 00:0C:6E:74:BD:B9 (Asustek
Computer)
Host 192.168.0.101 appears to be up.
MAC Address: 00:15:E9:2C:36:3B (D-Link)
Host 192.168.0.102 appears to be up.
MAC Address: 00:15:E9:2C:85:DA (D-Link)
Host 192.168.0.103 appears to be up.
Nmap finished: 256 IP addresses (5 hosts up)
scanned in 9.789 second
```

So now we have some information. We know that there are 5 hosts up on our subnet including our own. Taking a look at the IP addresses, 192.168.0.1 is a good candidate for a router. especially in relation to the other IPs. When we trace the packets route to Google with traceroute, it confirms that the packets are indeed routed through this IP.

```
#traceroute www.google.com
```

(output omitted to avoid redundancy)

You are now ready to start looking at individual hosts on your network.

First lets take a look at some of nmap's functions. There are two main types of scans that nmap can perform (as well as several others not discussed here). The first is a simple tcp connect(-sT), which uses the connect() syscall to actually attempt to make a tcp connection, then immediately breaking it off. This easily allows nmap to see what ports are open on a host, because they are the ones it is able to connect to. However you will almost certainly be logged by the host, as even the crappiest Detection software is going to notice that many

connections in a row. You do not need root privileges to run this scan.

The second most common scan nmap performs is known as the tcp SYN scan(-sS). The way this works is nmap sends a tcp SYN(Synchronize) flagged packet to a port. If a port is closed, the host will respond with a reset flag. if the port is filtered nmap will get no response, and finally if the port is open the host will return an acknowledged flagged packet, at which point Nmap will immediately respond with a reset flagged packet before a full TCP connection occurs. This will often prevent the scan from showing in the application logs, however most modern firewalls will pick up on the scan attempt, although this can normally be avoided by changing nmap's timing and utilizing other options we will discuss later. The SYN scan does require root access to the machine.

Finally before we start exploring hosts lets take a look at a few of the options nmap offers us. First lets look at the timing options. These are useful because a host is less likely to notice a scan if the ackets are sent at longer intervals. This is set with the -T flag, the options are 0 to 5, with 0 being around 5 minutes between ports, and 5 being around a .3 second delay, at the 0 time level you are very unlikely to be noticed, however a full scan would take forever, but with the fastest (level 5) data could be lost. I recommend a level between 1 and 3. Next lets look at IP Decoys. Decoying is a fairly effective way of hiding your IP. Basically it does exactly what it sounds like, it sends not only packets with your IP, but also other packets with spoofed IPs.

Then you probably don't want to keep a connection without knowing something about the other computers on the network

Therefore, to the host, it looks as if several IP's are scanning it. This is done with the -D flag followed by the decoy IPs separated by commas. Nmap will randomly place your IP unless you do so explicitly. I recommend using the IPs of other active hosts, or it will be fairly obvious who is the actual scanner. Nmap can also be set up to use only spoofed IPs with the -S flag, however you would never get an answer since the host does not know your IP address, so for our purposes it is not very useful. For those interested, your mac can also be spoofed using the -spoof-mac flag. Now for some of the more useful scan options, the first is OS detection.   A

Athough not always entirely accurate, nmap usually tells you if it is not positive. OS detection is activated with the -O flag. Another useful option is the -sV, which tells nmap to probe for what service and version are listening on a port. This is quite useful when looking for possible exploits. Readers may wish to note that both the -O and -sV flags can be set, by setting the -A flag. On a final note, if the user knows that a host is up, one may want to pass the -P0 flag, which tells nmap not to ping the host, keeping your scan much quieter. One may also want to use the -v option for more verbose information. I encourage everyone to run wireshark and tcpdump while learning to use nmap, seeing exactly what you are doing, is often important to learning, it can be quite interesting as well. Now armed with our new knowledge, lets find some information about a host.#

```
# nmap -sS -sV -O 192.168.0.100

Starting Nmap 4.11 ...
Interesting ports on 192.168.0.100:
Not shown: 1676 closed ports
PORT     STATE SERVICE      VERSION
135/tcp  open  msrpc        Microsoft Windows RPC
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds Microsoft Windows
XP...
1026/tcp open  tcpwrapped
MAC Address: 00:0C:6E:74:BD:B9 (Asustek...
Device type: general purpose
Running: Microsoft Windows 2003/.NET|NT/2K/XP
OS details: Microsoft Windows 2003 Server or XP...
Service Info: OS: Windows
Nmap finished: 1 IP address (1 host up) scanned...
```

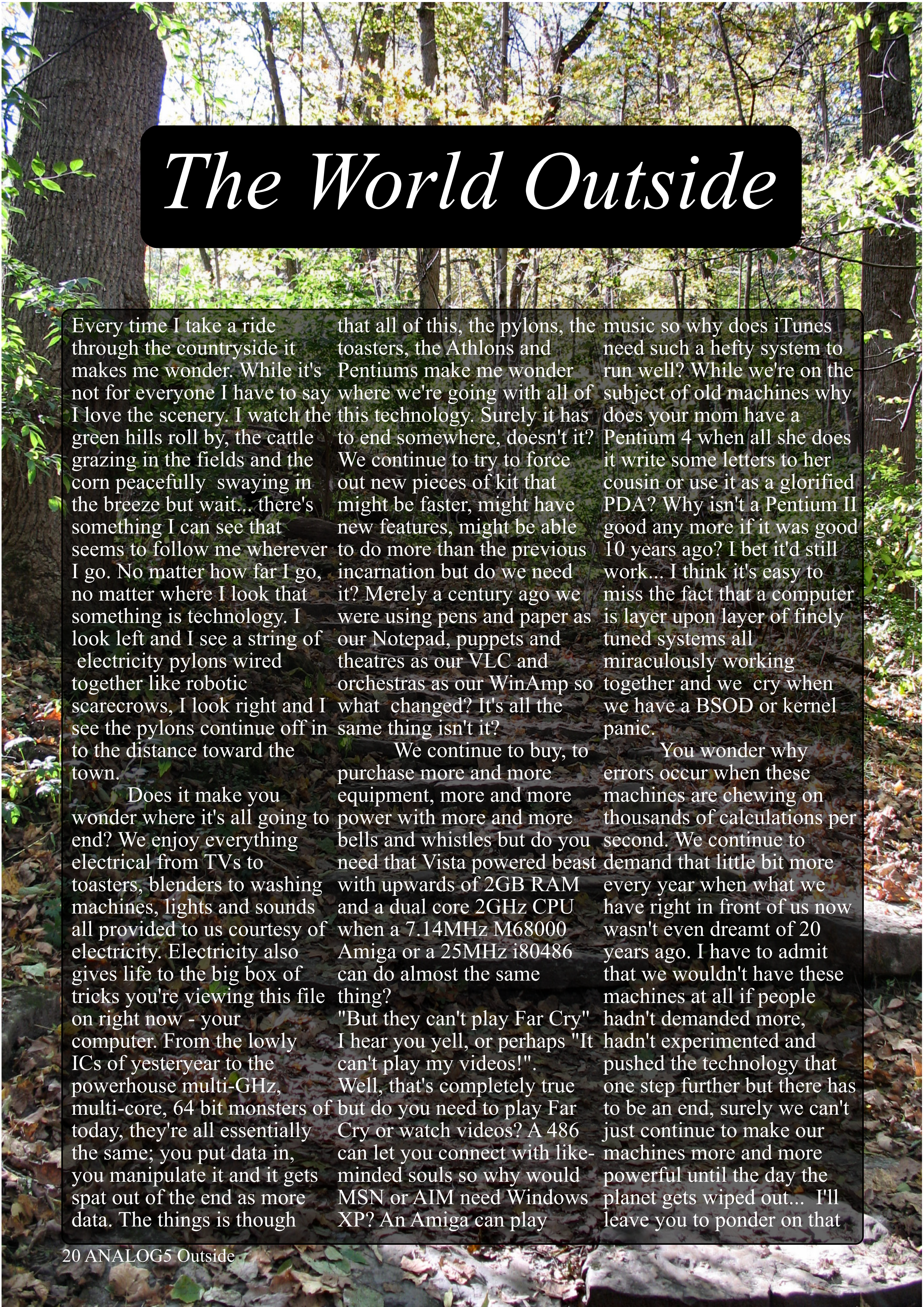
So what does this tell us, lets have a look. First off it is running either win2k3 or XPsp2. It is running 4 services, with only opened and closed ports, none filtered. This process can be repeated with multiple hosts individually, or nmap can be told to scan a number of hosts.

If one is doing this scanning for the purposes of security audits, and not just curiosity, I recommend several other things. Firstly familiarize yourself with the nmap manual ($man nmap), the are several scans not covered here which can give more information, especially in regards to filtered ports, udp scans, and other interesting stuff.

Also a great tool to have in your security arsenal is Nessus. When you come across an interesting host in your scans that you wish to audit for security purposes, I strongly suggest starting the Nessus service and scanning the host, it is an amazing tool. Just want to remind everyone again to not be afraid to look at Wireshark or TCPdump to see what is happening, also take a look at the programs (listed earlier) that I did not get too in this article, sing is also quite useful and a lot more powerful than ping. Happy scanning.

Links

Nmap	www.insecure.org/nmap
Nessus	www.nessus.org
Traceroute	www.traceroute.org
Sing	sourceforge.net/projects/sing
Netcat	netcat.sourceforge.net



The World Outside

Every time I take a ride through the countryside it makes me wonder. While it's not for everyone I have to say I love the scenery. I watch the green hills roll by, the cattle grazing in the fields and the corn peacefully swaying in the breeze but wait... there's something I can see that seems to follow me wherever I go. No matter how far I go, no matter where I look that something is technology. I look left and I see a string of electricity pylons wired together like robotic scarecrows, I look right and I see the pylons continue off in to the distance toward the town.

Does it make you wonder where it's all going to end? We enjoy everything electrical from TVs to toasters, blenders to washing machines, lights and sounds all provided to us courtesy of electricity. Electricity also gives life to the big box of tricks you're viewing this file on right now - your computer. From the lowly ICs of yesteryear to the powerhouse multi-GHz, multi-core, 64 bit monsters of today, they're all essentially the same; you put data in, you manipulate it and it gets spat out of the end as more data. The things is though that all of this, the pylons, the toasters, the Athlons and Pentiums make me wonder where we're going with all of this technology. Surely it has to end somewhere, doesn't it? We continue to try to force out new pieces of kit that might be faster, might have new features, might be able to do more than the previous incarnation but do we need it? Merely a century ago we were using pens and paper as our Notepad, puppets and theatres as our VLC and orchestras as our WinAmp so what changed? It's all the same thing isn't it?

We continue to buy, to purchase more and more equipment, more and more power with more and more bells and whistles but do you need that Vista powered beast with upwards of 2GB RAM and a dual core 2GHz CPU when a 7.14MHz M68000 Amiga or a 25MHz i80486 can do almost the same thing?

"But they can't play Far Cry" I hear you yell, or perhaps "It can't play my videos!". Well, that's completely true but do you need to play Far Cry or watch videos? A 486 can let you connect with like-minded souls so why would MSN or AIM need Windows XP? An Amiga can play music so why does iTunes need such a hefty system to run well? While we're on the subject of old machines why does your mom have a Pentium 4 when all she does it write some letters to her cousin or use it as a glorified PDA? Why isn't a Pentium II good any more if it was good 10 years ago? I bet it'd still work... I think it's easy to miss the fact that a computer is layer upon layer of finely tuned systems all miraculously working together and we cry when we have a BSOD or kernel panic.

You wonder why errors occur when these machines are chewing on thousands of calculations per second. We continue to demand that little bit more every year when what we have right in front of us now wasn't even dreamt of 20 years ago. I have to admit that we wouldn't have these machines at all if people hadn't demanded more, hadn't experimented and pushed the technology that one step further but there has to be an end, surely we can't just continue to make our machines more and more powerful until the day the planet gets wiped out... I'll leave you to ponder on that for a moment as I skip over to anotheraspect of ever improving technology.

We complain about 'big brother' watching our every move and telling us what we can't do but do we need to accept it? While they're all worrying about DVD piracy and who's leeching the new Britney Spears album we could go and experience something that doesn't have a monitor. Go for a picnic, go to a theatre and watch a play, go see a concert. Granted these things aren't police-free but you can enjoy them without fear that you're accidentally doing something illegal (unless you whoop out your handycam so you can be the first scener to XviD Les Miserable).

We seem to have forgotten there's life outside. Perhaps though it's because we're not so fond of people. I know I'm not. Really though, if you find a quiet spot somewhere in a field with a friend or two, perhaps even a lady friend or a strapping young lad with a basket of wine, fruit and cakes you're not likely to encounter obnoxious irritating idiots and much less script kiddies or rude gamers who are just there to scream HAX!! (and much much worse, I'm sure) over TeamSpeak. You get to choose who you go with, you get to choose what you do. I think sometimes we forget that we don't have to accept what the internet feeds us just like we say about the TV. I appreciate that it's not always easy to get away from it all if you're smack bang in the middle of a huge city or whatever but just think for a moment, what could you do? There's always something...

While you're watching the fat lady sing or the birds twitter though you can forget all those Japanese police robots too. Do these make you think it's all gone a bit too far? They have their uses I'm sure, perhaps in situations where it's a little dangerous for a real fleshbag to stride on in and start trash talking the dude with the gun but whatever do you imagine the Victorians would think if you put an autonimous robot on the streets... they'd probably be horrified... now why aren't we? We're striving to make our technology more user friendly to the point of trying to make computers act, talk, think and move like us, rendering lifelike images of humans that are almost indistinguishable from a real member of our species, creating robots with skin. Isn't this a sign that we're missing the point? Computers are tools like a flint, a hammer or a knife so why not treat them as such? You never heard of someone trying to make a chainsaw look like a cow, right?

This has been something of an incoherent rant for the most part but I guess the point of it was that you don't need to be in front of a computer to have a good time. Sure,we can go out and drink ourselves stupid and come home and compile the latest kernel or leech the latest movie but while computers may be an integral part of our lives they are only tools.

There are many other tools you can use to have a good time (yes, alcohol's one of them but there's still even more to choose from). Go out, take a book to the park and sit 'neath the old weeping willow and while away the hours in a world of humour, pain, horror, romance or just plain poetry. Turn off the monitor/TV and go for a walk, maybe take a friend or a partner and talk about all the stuff you could be talking about. Anything, there are so many possible things you could do... We did without computers before then we did fine with low spec computers so what's wrong with a PII being your main rig? You only need $6000 worth of kit if you rely on it to make a living or rely on it to live for you.

Go forth! Don't take the car unless you need it to get somewhere nice, go to the beach, a park, anything! In fact, it doesn't matter where you go if you've got good company. 'Do lunch' with some people, maybe an evening meal. Oh, and one more thing; you don't need to get drunk to enjoy alcohol, it's a tool, don't overuse it. I'm stuggling to find an end to this thing, so I'll nid thee farewell and hope you enjoy exploring the better side of being sociable. Later.

-Moonlit

(DIR)
bytes
Settings
and Settings
(DIR)
ts and Settings\Adm
670
628
(DIR)
(DIR)
(DIR)
(DIR)

C/C++ Basics

Many of you will be reading this because you wish to learn how to code or you know how to code and want to learn something new. I will generally be using C and C++. I will assume anyone reading this has little to no knowledge of C or C++.

To start off with I'll show you simple console input and output in C and C++. To follow these examples you will need a C/C++ compiler with a standard C library, and standard C++ library.

I will provide a list of good compilers at the end of the section. It is a good idea to read your compilers help files to understand how to use it.

Console output

Console output is basically display any type of output on a console (terminal). Generally this is usually text.

For the first example we are going to output 'Trust Your Technolust' onto the console. After I will example what the lines of code do.

```
#include <stdio.h>
  int main()
  {
   printf("Trust Your Technolust!\n");
   return 0;
  }
```

Compile this with your compiler and run the executable. If everything went well it should display 'Trust Your Technolust!' on the console. Now I will explain what each of these lines is doing.

1. #include <stdio.h>
#include is a preprocessor directive which tells the preprocessor to include the file 'stdio.h'. The file stdio.h contains all the function prototypes for standard input/output functions. Which means we can use them in our project.

2. int main()
This is the main function of the program, and should always exist in your code. The 'int' part is the type that the function returns. Sometimes you will see 'void main()' or maybe even see 'int main(int argc, char** argv)', I will explain these another time.

3. {
Functions require you to use curly braces. You will get used to using these in your code.

4. printf("Trust Your Technolust!\n");
This is a function that prints text to the console screen. We give it the argument "Trust Your Technolust!\n". This is a string. The '\n' you see at the end is an escape character. The one we are using means insert a newline. Also remember to put the semicolon after function calls.

5. return 0
This tells the program to return 0 when it finished. Returning 0 means that the program didn't have any errors when executing.

6. }
We must remember to close our curly braces. This is all the code we need. it should look like the following code:

```
#include <stdio.h>
 int main(int argc, char** argv)
  {
    printf("Trust Your Technolust!\n");
    return 0;
  }
```

When you compile this code, and run the executable. It should display the output 'Trust Your Technolust!', and exit.
Now we will look at the exact same application, but written in C++ instead.

```
#include <iostream>
using namespace std;

 int main(int argc, char** argv)
  {
   cout << "Trust Your Technolust!" << endl;
   return 0;
  }
```

As you can see this code is pretty different from it's C counter part. Now we will look at each line of code, and see what it does.

1. #include <iostream>
This does the same thing as in the C example, except it is including the header file "iostream". You will see that many C++ headers don't have the .h extension.

2. using namespae std;
his allows use to use the functions in the std namespace without having to prefix them with 'std::'. This isn't that important at the minute, but remember to include it.

3. int main(int argc, char** argv)
This is the same as explained in the C example.

4. {
The usual open bracket for the main function.

5. cout << "Trust Your Technolust!" << endl;
This is line that outputs the string to the console. As you can see it is very different. 'cout' is the standard output defined as a steam object. The stream object is used in connection with the insertion operator, which is two less-than symbols '<<'. 'endl' is a manipulator that adds a newline. 'endl' also flushes buffered streams. Generally 'cout' will be an unbuffered stream.

6. return 0
This is the same as explained in the C example.

7. }
This is the same as the C example. Always remember to close your opened brackets.

When you compile and run this application, it should do exactly the same as the above C example. Now we have looked at standard output, I am going to show you standard input. Which is basically getting input from the keyboard. I will start off by showing you the example, and then explaining each line.

```
#include <stdio.h>

 int main(int argc, char** argv)
 {
  char name[20];
  printf("Please enter your name: ");
  scanf("%s", name);
  printf("Hello, %s!\n", name);
  return 0;
 }
```

Please understand I am not including any error checking, because it is only the basics. When you become more experience you will be able to incorporate the needed error checking.

1. #include <stdio.h>
This is needed for the standard input/output functions in the standard C library.

2. int main(int argc, char** argv)
This is the main function as usual.

3. {
The open bracket, get used to it you will be seeing lots of them ;-).

4. char name[20];
This is our first use of variables. To store a name we need to store a number of characters in an array. A string is basically an array of characters in C and C++. So we declared an array called 'name' that can store 20 characters, enough for the average name.

5. printf("Please enter your name: ");
This is just displaying the question we are asking the user, using the same function we used in the C example for output.

6. scanf("%s", name);
The function 'scanf' scans the standard input for the intput required and stores them in the specified variables. So '%s' means we are taking a string from the standard input, and we're storing it in our variable called 'name'.

7. printf("Hello, %s!\n", name);
This shows us how we can use the 'printf' function is display output that includes variables as well. The string we want to display is 'Hello, %s!\n', so we are displaying a string variable, and it's passed as the second argument to the 'printf' function.

8. return 0;
This is the same as explained above.

9. }
Remember to close those open brackets ;-).

When you compile and execute this application, it'll prompt you to enter your name, when you do and press enter, it'll display 'Hello, name!', where name is your name, then exit. Now we will look at the same example in C++.

```
#include <iostream>
using namespace std;

int main(int argc, char** argv)
 {
  char name[20];
  cout << "Please enter your name: ";
  cin >> name;
  cout << "Hello, " << name << "!" << endl;
  return 0;
 }
```

1. #include <iostream>
Include the <iostream> header as usual for C++ programs.

2. using namespace std;
Allows us to use the functions without prefixing 'std::'.

3. int main(int argc, char** argv)
The main function.

4. {
The infamous open bracket.

5. char name[20];
This is the same as the variable declared in the C example.

6. cout << "Please enter your name: ";
This is the output for our question to the user.

7. cin >> name;
Now we are introduced to the standard input stream object 'cin'. We use the extraction operator, which is two greater-than symbols to get the string from standard input, and store it in 'name'.

8. cout << "Hello, " << name << "!" << endl;
This is outputting the message 'Hello, name!' to the user.

9. return 0;
This is the same as all the above examples.

10. }
Remember to close the open brackets.

When you run this program, it should do exactly the same as the C program. There is one thing you have to remember when using 'cin' with strings. It will only get the string up to the first blank character. This means we can only get single words.

Next issue I will expand on the different data types, and some selection and iteration.

-TomB (www.ownthebox.net)

Voice Row

Voice Activated Mac Media Center

There are millions it seems media centers out there of many different types and complexity but nothing matches the coolness of the old Star Trek episodes where the computer did whatever you said, literally. This project is not the holodeck but trust me it's just as cool. Even better is having a solid media center or media center front end can be had for as little as 200 dollars. Plus you can talk to it.

Old Mac Vs New Mac

The first thing you have to look at when getting started with this project is do i have an old mac or a new mac. The reasoning behind this is we are going to be using the Front Row software as the base and it only comes with newer macs. If you have a newer mac with Front Row you can skip step 1 and go straight to step 2.

Step 1: Getting Front Row on an Older Mac:
If your older mac is running Tiger 10.4.4 your all set. You can download Front Row from apple and install with no problems. If your not running this version of Tiger your going to have to get your hands dirty with some quirky hacks. Instead of trying to explain that giant mess here I am going to point you to where i learned at this link (www.andrewescobar.com/frontrow) from Andrew Escober's website. He has a great tutorial on getting Enabler to work and get Front Row installed on those older macs. Another quirk with older macs is they do not offer the Apple remote used for Front Row.

There are a lot of ways around this using keyboard commands etc but i prefer to use a wireless remote from a third party like Keyspan (www.keyspan.com). There RF Remote for Front Row Part Number, ER-RF1, is a perfect replacement for the Apple Remote.

Step 2: Let there be Voice
Once you have Front Row set up you should be able to hit Command+Esc and Front Row will open. You can hit it again to close the software. We will now take advantage of this keyboard shortcut via voice. A quick tip is to go into Keyboard & Mouse under System Preferences and disable the Front Row short cut because the next Step does not seem to work unless it's shut off. When you try and type the command it just opens Front Row instead of recording the strokes.

Once you have that shut off go back into System Preferences and click on Speech.Make sure "Listen continuously with keyword" is set to on and the keyword is Computer. Now turn Speakable Items on and say the following command "Define A Keyboard Command" a dialog box will pop open asking for a set of keyboard strokes. Hold down Command and press Esc it will show up as Command+Esc if so click ok. Where it says "By Saying this Phrase" type in the phrase you wish to use. In my case Open Front Row.

Click on the Any Application radial and click Save. Once you have done this go back into Keyboard & Mouse and turn the shortcut back on. Once this is ready say "Open Front Row" and front row should open. Say it again and it should close. You can set up another command using "Close Front Row" if you would like.

Step 3: My table of Commands

Here is a quick list of commands i have set up for voice activation. Of course once you have the first one setup the sky is the limit on this hack. You can control most of your Apple machine using voice commands. It might be a cool idea to learn all of these and impress you friends.

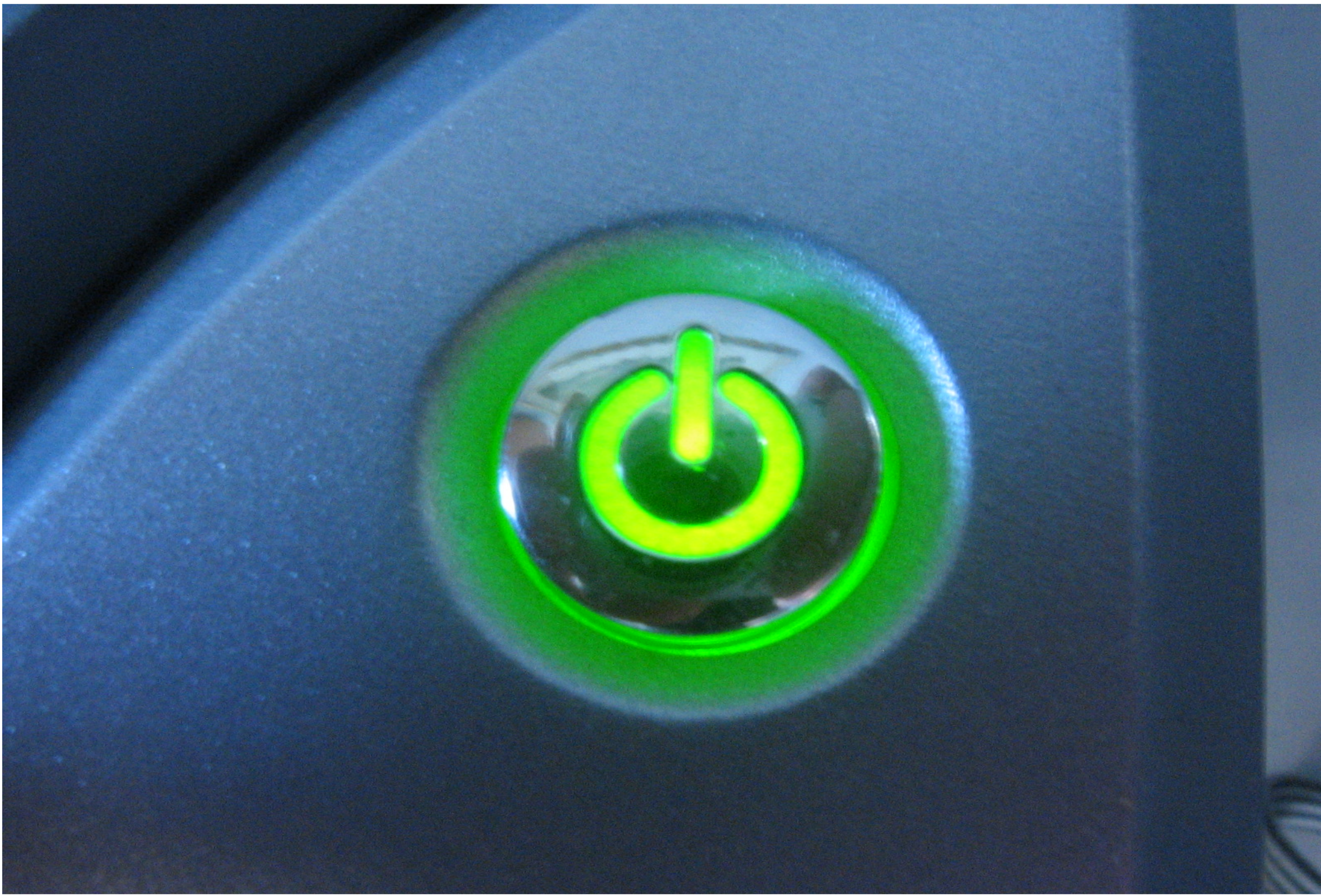
Open Front Row - Command+Esc
Left - Left Arrow Key
Right - Right Arrow Key
Up - Up Arrow Key
Down - Down Arrow Key
Enter - Return Key (Selects an item with in front row.)
Back - Shift+Esc (Esc seems to send you back to the desktop where as Shift+Esc will just take you back a page in Front Row. If your at the turn table. it will also slowly take you back to the desktop which looks very cool.)

Once you have all of these set to voice commands your ready to control your front Row media center with nothing but the sound of your own voice.

Step 4: Notes/Contact Info

- Episode 2 of my show Random Acts Of Anarchy (RAOATV.Blogspot.com) should have a video demo of this in action.
- Any questions can be sent to UndergroundInformation@Gmail.Com with "Voice Row" in the subject line.
- If you would like to follow my progress with this project using a G4 Mac Mini you can hit up my blog (www.Shinmaryuu.vox.com). I am doing a lot with media centers so it's becoming a good resource for information.

-Shinmaryuu (Shinmaryuu.vox.com)



Hindsight

I know what a lot of you are thinking: "What was that?" And I don't blame you. This was a rather rough edition. However, this is the first layout I've ever done, and I was trying to do a cross between PC Magazine and Relevant Magazine.

I don't know if you've ever seen Relevant, but it is quite possibly the most pleasing, eye-candy reliant magazine ever. Every page is filled with color/texture, varying fonts, different layouts and is a joy to read.

I think, though, that this might not work so well for Analog5. The PDF file ends up being huge and the pictures became more of a distraction at some point. I think I may seriously reduce the amount of images I use in the next issue.

Some of you might also wonder why there are random images at different parts of the magazine, such as the Sith Cat picture on page 19. Well, it's simple: magazines have ads. Ads take up any unused whitespace the layout designers have to deal with.

Analog5 doesn't have any ads, so I had to fill some whitespace and I thought that a community based magazine should feature art from a community-driven site.

You may also notice different sized article fonts. I'm not sure what to do there either. Print magazines can get away with 12 pt font. But this is an Internet magazine and I don't think people should have to zoom in to read an article. However, bulking up the font lengthens the articles and the design part gets ugly.

In all honesty, I don't know what to keep and what to get rid of. I need feedback. I need you, the readers, to tell us what you like and don't like. But please don't just write and say "this [enter noun] sucks". Tell us specifics. Tell us why you don't like something. Tell me what size font you like. And please, drop some ideas, we'll probably use them, and then give you credit.

That's all for now. I wish I had a preview of the next issue to include but I don't.

Till next time

-Brainedchild