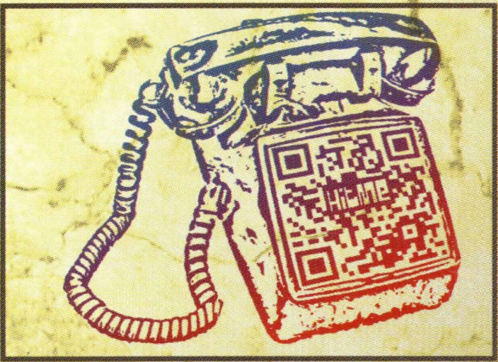
Volume Thirty-One, Number One!

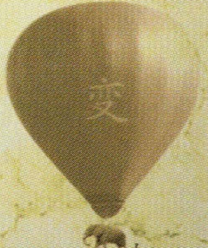
Spring 2014, $6.95 US, $7.50 CAN

2600

The Hacker Quarterly



XXX.XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX.XXX



AHOY!

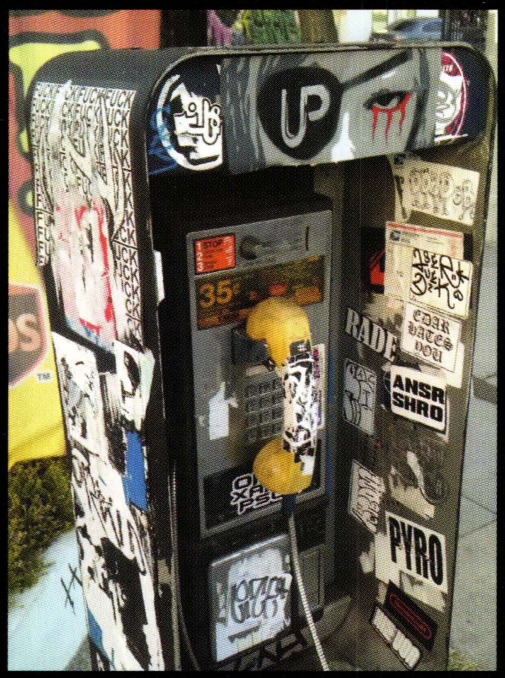
(That's how Alexander Graham Bell used to answer his phone. For some reason, it never caught on...)

0 74470 83158 7 4 1>

Artistic Payphones

Thailand. This isn't the first Thai phone we've printed that appears to be heading back to nature. This one was spotted in Sai Yok, near the Death Railway (don't ask).

Photo by Kimmo



United States. Seen on Melrose Avenue in Los Angeles, they call this a punk rock payphone and it's easy to see why. In fact, we wouldn't be surprised if this one went on tour in the 1980s.

Photo by Glenn Griffin

Canada. Montreal is apparently known, not only for its plethora of payphones that people actually still use, but for occasional artistic payphone expressions.

Photo by Jonathan Mertzig

China. Meanwhile, over in Shenzhen, the artistic look is a bit... minimal. In fact, the inside of this booth would make a pretty convincing prison cell.

Photo by DrSm0ke

Got foreign payphone photos for us? Email them to **payphones@2600.com**. Use the highest quality settings on your digital camera! (Do not send us links as photos must be previously unpublished.) (More photos on inside back cover)

map

be the first

Thirty Years On

With this issue of *2600,* we have begun our fourth decade of publishing. It's rather hard for us to believe, but that's what the start of our 31st year means. If there's any theme that's accompanied every aspect of this whole project and the world we're a part of, it's that life is always a whole lot more unpredictable and strange than anything that was ever predicted.

That's not to say the many people who have been involved in our previous 152 issues didn't anticipate a lot of what today's world has become in their visions. These, after all, are the pages that held many of the ideas and values that helped to define the Internet. The very notion of security was outlined repeatedly right here, with endless examples of what constituted bad practices on every level imaginable. And technology of all sorts, ranging from devices of mischief to tools that made serious strides in improving lives was discussed, theorized about, and demonstrated year after year in nearly every edition of *2600*.

That, perhaps, is the strangest part of this entire evolution: that we have been there to witness it, comment on it, and even help it along in various ways. We never guessed that the things those of us found interesting in the beginning would ever be relevant to anyone else, let alone such a huge part of the world. We enjoyed playing with telephones because they linked all parts of the world together in what seemed like a magical accomplishment. Today, of course, the whole world knows this and connections to other parts of the planet are routine. We used blue boxes and routed calls on our own to expand the possibilities and eliminate the costs, which by definition was stepping outside the boundaries of legality. Now, reaching out like that is not only permitted, but encouraged. We explored computers of all different networks because there was so much to learn but precious little in the way of opportunities to do so. Being told that we weren't allowed to learn about UNIX, for example, wasn't something that we as hackers took kindly to. If the only access to such a system was by breaking in and exploring, then that was what had to be done. And now we are deluged with UNIX-derivative systems for people to run on their own devices, devices that are a tiny fraction of the size of the cumbersome mainframes of the past with infinitely more potential and capacity.

It's easy to predict that technology will improve; what's hard is knowing how society and individuals will change as a result. It wasn't so hard for us to assume back in the early days that things would get faster, smaller, and cheaper. What we didn't know was that it would mean so much to so many others. This was just something we were interested in because it was cool and because we were curious and a bit mischievous. Sure, there was always a bigger picture looming in the background for some of us: the extension of First Amendment rights to digital outlets like BBSes, opposing the abuse of power by authorities in search and seizure activities, battling the irrational fear of what hackers could do, and getting more access to technology for the mainstream. But the true mystique here was in that sense of adventure and the feeling that we were embarking on a quest that few understood and that many more feared. Contrary to much of what we've all been taught, unbridled fun can often lead to far more meaningful accomplishments.

That adventurous spirit is something we need to make sure isn't lost in the shuffle. While it's great to see improvements in technology and access, it would be a misfortune to lose sight of the wonder of it all. For instance, when hackers of the past would make a long distance connection, there was a real sense

of accomplishment and appreciation at what was actually taking place. The same was true of making a computer perform a task that it hadn't necessarily done before. Realizing that all of this interaction was taking place in a tiny electronic environment felt like a bit of magic. One thing to worry about today is that this magic has become so routine that the excitement has been drained out of it, and all of these things we're playing with have become commonplace and even boring. That might be how it turns out for people with little imagination, but in the hacker world, even the most common bits of technology continue to inspire and captivate us. Many decades ago, the thought of human voices being carried over wires seemed a true marvel. Well, it still is. And as with anything else, just because it's now commonplace and exponentially faster doesn't make it any less amazing. When that fact is forgotten, we lose the excitement factor which is what traditionally has propelled us forward.

As we reflect on all of this, it's also fascinating to see how much *hasn't* changed:

- We still see hackers being demonized every time something goes wrong with technology. The difference is that there's a lot more technology today - and a lot more people willing to define themselves as hackers.
- The powers that be continue to want to get more personal info on all of us. Thirty years ago, we were talking about credit reports, isolated surveillance cameras, and pen registers. Today, it's a whole lot more insidious: many of us *want* to help add to the databases for the sake of convenience and better social lives. And, of course, the all-encompassing nature of the data that is collected dwarfs anything we used to be worried about. What hasn't changed is our overall concern and suspicion, as well as the desire of the authorities to push ever further into that which used to be out of bounds.
- The media continues to miss the story. Even with all we've learned from Wikileaks, Manning, and Snowden, the mainstream continues to be told that *they* are what constitute a threat, not the egregious violations they've uncovered. And, like we've seen so many times over the decades, every time a security hole exposes private information, it's hackers (sometimes even theoretical hackers) who get the blame, not the poorly designed systems and lack of accountability.

On nearly every level, today's technology is virtually unrecognizable from what we would have been working with 30 years ago. But the desire for exploration, the threat of oppression, the fight for freedom, and the specter of ignorance are all pretty much in the same positions we found them in 1984. That doesn't mean significant progress hasn't been made. It's precisely because those who care haven't given up that we can still have these conversations and reach more people than ever. In a world with such rapidly and dramatically changing technological abilities, the threats from those who wish to control, subvert, and abuse them are always going to be present, no matter how forcefully they were turned back previously. If, say, we succeeded in completely dismantling the NSA's ability to violate our privacy, they would simply return in some other form down the road. Historically, these things just don't go away. But then, apparently neither do we.

The hacker spirit is one of those traits of humanity that cannot be suppressed. It's in our nature to push back when told we're being restricted, whether that is being applied to what we're allowed to know about, what software we're permitted to use, what rules we're expected to accept without question.

This coming year also happens to be when our Hackers On Planet Earth conference series marks its 20th anniversary. Our tenth conference (HOPE X) will mark the occasion this July in New York. We expect to have an unparalleled selection of talks and activities devoted to new technology, free speech, surveillance, dissent, and all kinds of other issues of interest to hackers. We hope to see you there.

We'll undoubtedly be facing all sorts of challenges in the next year and the next 30 years. Without the spirit and skill that this community is constantly building upon, the roads ahead would indeed be dark ones - for all of society.

Lessons from "Secret" History
From Cable Vetting to Tempora

by Poacher

"No doubt it is comforting to be told that one's privacy is as fully protected in a public telephone booth as it is at home. But it is less reassuring to realize that one's privacy is no better protected at home than in a public telephone booth." - Telford Taylor, *Two Studies in Constitutional Interpretation,* Ohio State University Press, 1969

In the early days of last summer, I was reading a piece on the news about the damage to two undersea cables off of the coast of Egypt. Somewhere along the line, I was pointed to a map of the world's undersea cables for carrying Internet traffic. It's an astounding map showing the hundreds and thousands of miles of cable laid on the sea beds of the world. I was further amazed to find that one of the links between the U.K. and Western Europe came ashore a few miles from my house.

I knew the location - I remembered childhood expeditions to the beach there and the faded yellow sign warning ships of the cable. Of course, it hadn't been a thick collection of multi-mode fiber optic cables back then, but a bundle of copper phone lines. I took a walk down there, curious to see what the landing station for probably one fifth of the U.K's Internet traffic with Europe would look like.

What I found looked little different from when it carried a simple copper wire across the Channel. A small brick building, probably built in the late 1940s, little larger than a garden shed. The yellow diamond shaped sign was still there on a wooden pole about ten yards down the slope to the beach, facing the sea and warning any ships so hopefully no one would drop anchor and drag the cable up. Beneath the tiled roof, vents had been knocked through the bricks and a telephone company sign was screwed to the wall proclaiming the building to be an exchange. The windows had all been more recently bricked up and some fairly high end locks fitted to the green painted front door. A sturdy wooden fence surrounded the rear and, peering over this, I was greeted with what had once been a small but well kept garden, now overgrown, but suggesting a long past era when a small exchange like this would have been manned.

Over the course of that summer, a number of subjects I was looking at all came into an odd kind of coincidence and a strange story emerged linking Edward Snowden's revelations, the First World War, and a political scandal in the 1960s. The starting point was the cable landing station, the finishing point an unpleasant conclusion about widespread state surveillance.

By the nineteenth century, Britain was at the height of its imperial power - the empire that the sun never set on and an economy to match it. Administering such a vast commercial enterprise required armies of civil servants and a communications infrastructure that was state of the art. So when in 1844, the first successful electrical Morse transmissions were made between Washington and Baltimore, it is no surprise that Great Britain would adopt this new technology with zeal and vigor.

Britain had a couple of advantages over the rest of the world at this time in history which gave it the head start in the nascent communications revolution. The size of its economy meant there were plenty of people willing to take the risky step of investing in new and unproven technology. Along with the largest Navy in the world, Britain also had the largest merchant fleet. There were clear commercial advantages to being able to communicate with your ship's captains as soon as they made port rather than wait for them to return home weeks later to receive their next instructions. A last advantage, which we shall touch on again before this is over, was the empire itself. Being a maritime economy, Britain had amassed a large collection of islands and coastal territory all over the world. These were vital for ships to take on fresh water and food, and later as bunkering stations when coal and then oil took over as means of powering ships. Often little more than outcrops of rock in a vast ocean, these stations became very important as relay stations, first for telegraph and then later for wireless. Little surprise

then that by 1896, of the 30 cable laying ships in the world, 24 of them were British owned.

Despite being privately financed and owned, it is clear that such an important tool as the world's first electronic communication network should be subject to government control. Government operators on the system could claim priority in sending messages. The British also realized the importance of communications security very early on. Alert to the dangers of cables passing through territory they didn't control, where the cable could be cut or listened in to, they set about creating what was to be called the "All Red Network," named so because the areas on a world map belonging to the British Empire were colored red.

The importance of this became very apparent in the 1914-1918 war. At the outbreak of the conflict, although the Marconi company had begun to build a wireless network to replace telegraph, Britain had a fleet of 28 cable laying/cutting vessels, more than twice the rest of the world combined. These were put to good use in 1914 when war was declared on Germany and Cable Ship "Alert" was deployed to cut the five cables linking Germany with France, Spain, and the Azores, thus severing Germany's links with North America, save for wireless, which British Naval intelligence could intercept.

We now jump to 1967 - sadly missing the stirring tales of wartime signals intercept and code-breaking, the formation of GCHQ out of the Government Code and Cipher School and the birth of the NSA, among many others. Lyndon Johnson is resident of the White House, Harold Wilson is Prime Minister of the United Kingdom. The Vietnam war is in full swing, The SEACOM telephone cable is inaugurated, the Boeing 737 makes it maiden flight, the Six Day War happens, and, more dramatically, there is a 13-day television strike in the United States.

Harold Wilson came into power after the resignation of Harold Macmillan. In opposition, Wilson had seen the effect that several high profile spy scandals had had on his predecessor and has been said to have been extremely sensitive about matters of security while in office, to the point of paranoia.

In 1966, Wilson established what has become known as "The Wilson Doctrine." This rule states that no member of Parliament should have their phone tapped. This rule has been continued by every prime minister since and now covers electronic communications as well. Harold Wilson's decision to implement this rule becomes interesting a year later.

On the 21st of February, 1967, journalist Chapman Pincher published an article in the *Daily Express* newspaper exposing the practice of "Cable Vetting," a process where all international telegram and telex messages were passed on to government agencies by the cable companies. Purportedly, the story originated with a disgruntled employee of one of the cable companies.

Sadly, it seems the real issue of the story became overshadowed by the misguided attempts by Wilson to cover it up. In the U.K. since 1912, a voluntary system of press censorship had existed known a "D Notices" or "D-A Notices." These "Defence Advisory" notices were requests by Government to the press not to publish stories on a range of subjects that could be detrimental to national security. They were not legally enforceable, however, it was almost unheard of for an editor to ignore a D-Notice.

The resulting scandal which hinged upon whether a D-Notice had been issued in respect of the story rumbled on for quite some time, and it seems the actual story became forgotten in the mass of inquiries that followed. The political scandal is now what's remembered and not the interception of private messages.

The issue that Pincher exposed is resoundingly familiar in 2013, the widespread interception of private citizens' correspondence enabled by a secret relationship between communications companies and Government departments.

Coming almost up to the present day and this time *The Guardian* newspaper is publishing material provided by ex-NSA contractor Edward Snowden. On Friday the 21st of June, 2013, *The Guardian* ran a story describing how GCHQ is tapping fiber optic cables to access the world's Internet traffic.

A look at the submarine cable map will show the British Isles as having a huge number of cables landing on its almost 8000 miles of coastline. Take a look at some of the more remote landing stations and you'll find they are often in what were historically British controlled ports and islands. In fact, take a look at some historical maps of the early telegraph cables and you'll find a lot of them are in the same places as the current Internet links.

The geographical cards that Great Britain held are really the underpinning of the special intelligence relationship between Britain and the United States of America. The now widely known U.K.-U.S.A. agreement, just one of a

complex web of agreements dating back to the Second World War, was always an asymmetrical relationship. What could the U.K. offer against the vast resources, cash, and manpower that the U.S. intelligence community had? The answer is some very useful real estate, both in the U.K. and abroad. A prime example of this is the effect that a temporary ban on U.S. reconnaissance flights from U.K. bases had. Imposed by Harold Wilson in 1967, this coincided with the outbreak of the Six Day War in the Middle East. At one point, the U.S. had to resort to flying spy planes from the Eastern Seaboard of the U.S. all the way to the Sinai Desert, involving a large number of hazardous in-flight refueling, both going and returning.

After PRISM, Snowden revealed Tempora, GCHQ's massive cable tapping program where petabytes of information are pulled from the cables and stored for up to 30 days. Tapping over 200 fiber optic cables and processing data from over 46 at a time, GCHQ and, by extension, the NSA are listening in on a huge percentage of the world's web traffic.

Once again in echoes of 1967, we find that this has been happening behind the scenes with the complicity of the private companies entrusted with carrying the data. And yet again, we have seen the attention of the media shifted away from the actual story and focusing upon the surrounding scandal: the sensational hunt for Snowden and then his stranding in Moscow turning the spotlight away from what Edward Snowden was revealing and towards Snowden as a media event. Much the same has happened to Julian Assange and Wikileaks.

So, from a remote cable landing station, we arrive at the latest mass surveillance initiative. During the D-Notice affair, it was revealed that "cable vetting" had been going on since at least the 1920s. Tempora had been going for a couple of years when it was revealed.

But what of the intervening years? There is the period of time between telegrams and the Internet when the majority of communications traffic was carried through the plain old telephone system. It is surely inconceivable that governments used to being able to listen in to its citizens at will since the creation of electronic communication would have sat back and done nothing.

Short of documents being declassified, and I doubt we'll see that in this lifetime, we are left with waiting for another whistle-blower to reveal the truth. There is, however, a little evidence out there that may point to what we all suspect has been happening.

In 2000, the United Kingdom government was taken to the European Court of Human Rights over the wholesale tapping of telephone calls between the U.K. and the Republic of Ireland. A year before, Channel 4 News had reported on a tower in Capenhurst, Cheshire, which was used to intercept microwave links between the U.K. and Ireland. The tower, subsequently sold off for 20 million pounds, sat between telephone relay stations at Gwaenysgor, Clywd, and Pale Heights near Chester. It allegedly had the capacity to intercept 10,000 simultaneous phone channels. The site was in operation for ten years from 1989 until 1999.

This is probably the tip of a very large iceberg. It's likely that there has been widespread monitoring of the phone system since it began. In the U.K., it would have been trivially easy, as for most of its history the telephone network in the U.K. was run by the government. Starting as the Electric Telegraph company in 1846, by 1912 the running of the network was taken over by the General Post Office, a government department. It was not until 1984 that it was privatized to become British Telecom.

Wholesale government monitoring of the communications of its citizens is as old as the communications networks. Despite being exposed, they just keep on doing it and the public seems quick to forget. It's sad to think that the bravery of people like Ed Snowden may ultimately come to nothing, but so far that's the lesson that history is teaching us. It's up to the rest of us now not to forget and not to willfully ignore what's going on and to demand transparency, not just from governments, but also from the companies that carry our traffic. After all, we're paying for a service; we have the right to stipulate how that service is provided.

Bibliography

- *Intelligence in War,* John Keegan, 2010
- *GCHQ,* Richard Aldrich, 2010
- "The NSA Files," *The Guardian,* `http://www.theguardian.com/world/the-nsa-files`
- "GCHQ taps fibre-optic cables for secret access to world's communications," *The Guardian,* June 21st, 2013

Google's Two-Factor Authentication: The Sneaky Trust Feature

by Samuel A. Bancroft
SamuelBancroft@gmx.com

Since September 2010, Google has provided its users with the option to use two-factor authentication to add a layer of security to their accounts.

ATM machines are widely used as an example of two-factor authentication. The user has to provide a PIN number - something they know - and a debit card - something they have - in order to be able to gain access to their account. Having only one of the two factors would not be enough to satisfy the requirement to gain entry.

In the case of Google's two-factor authentication, Google uses their Google Authenticator phone and tablet app to create the second verification factor. But note, the user may use another application to generate the second verification if they so choose to. [3] Google Authenticator is capable of using both the HOTP and TOTP algorithms to generate six integers that the user then uses to authenticate themselves. [1] [2] The TOTP, or Time-based One-Time Password algorithm uses a cryptographic key and the network's time to generate a new and unique six digit passcode every thirty seconds. For a user to be able to access their Google account, the user would in turn have to provide their password and also the unique six digit passcode, generated on their phone or tablet. Knowing the password alone would not suffice. This makes stealing the user's password useless unless the TOTP passcode for that specific moment in time can be derived or stolen.

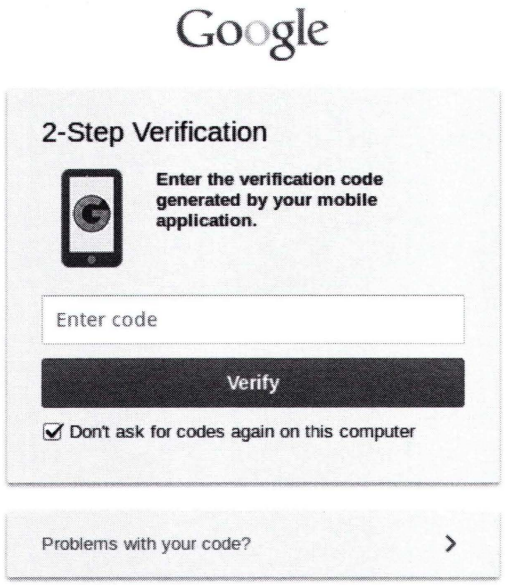
On a side note, Google allows the printing of a special set of static passcodes in case the user loses access to the Google Authenticator app, i.e., if they lose the phone or tablet which they were using to generate the TOTP passcodes. These static codes are usually kept in the user's wallet.

But this article is not about how two-factor authentication works or how to beat it. Instead, this article will touch upon a feature Google is currently using that has the possibility of leaving the user open to an attack without them realizing it.

Google, in their efforts to make two-factor authorization less intrusive to the user, has implemented a feature that I have dubbed as the "sneaky trust feature."

When Google notices that a specific computer is constantly being used by the user to login, the two-factor authentication code page will automatically check the "Don't ask for codes again on this computer" check-box by default. Consequently, if the user enters their six digit code and logs in without noticing the check-box or forgetting it's checked, the two-factor authentication for that specific computer is turned off without any notification to the user. Furthermore, this checked check-box is persistent. It will be checked every time the user attempts to login regardless if they have unchecked the check-box in the past.

The lack of notification creates a situation where the user may be unaware that they inadvertently trusted the computer. This in turn would give an attacker a window to use the newly trusted computer to gain access to the user's account without worrying about two-factor authentication. This attack window will likely be closed the next time the user logs in since they ought to notice that the TOTP passcode are no longer being asked for. Keep in mind also, since the user thinks they are protected by two-factor authentication, they may become more relaxed as to logging into their account from questionable computers, making stealing their passwords easier.



An example of the "Don't ask for codes again on this computer" being checked by default. The only notification is the small check-box being checked.]

I installed Linux on a machine to test how often a user has to login in order for the "Don't ask for codes again on this computer" check-box to be checked by default. I found that after logging in 22 times within a time span of 23 hours, the check-box became checked persistently. In order to clear the check-box from being checked, the user has to clear their cookies.

1. http://tools.ietf.org/html/rfc6238
2. http://jacob.jkrall.net/totp/
3. http://code.google.com/p/google-authenticator/

IDENTITY AND ENCRYPTION VERIFICATION

by xnite

Given the recent leak of the spying program known as PRISM, a lot of people in our community have been worried about how safe their communications online actually are. In light of these events, I decided it would be a good time to start talking about different methods of fingerprint and public key verification as well as key signing parties. When we create a PGP public/private key pair, we are given a fingerprint for our key which is unique. A key signing party consists of giving others this fingerprint in person, so that they can look your key up on a key server and sign the key with their own PGP key. By signing your key, they are telling the world that they 100 percent trust that this key belongs to you. The signed key can be placed up on a key server and others can view and verify the signatures. It may not be a bad idea to start asking around *2600* meetings for others to sign your PGP key.

A common method of encrypted communication which I see and use is via OTR (off the record) on XMPP (Jabber) servers. In this case, when you start an OTR conversation you transfer your public key to the other party and, in turn, you get their public key. The client will usually ask you to verify the key by checking the fingerprint. To check the fingerprint, you would usually want to be on the phone with the other party, or have already obtained a copy in person. In many cases, this is not possible, so my favorite method of giving my OTR fingerprint to others is by creating a text file containing the fingerprint and signing the file with my PGP key. This validates that my key was used to sign the message, and they can check to see who has signed my key and ultimately decide if they will trust my OTR fingerprint.

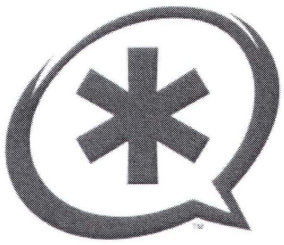
For people who are not willing to expose their true identity, it's hard for others to actually verify that they are who they say they are. Nonetheless, it does not mean with 100 percent certainty that they cannot be trusted. An example of one of these people might be a political activist or hacktivist. These people usually communicate in plain text somewhere such as Twitter. We tend to assume that we can trust that the posts coming from their Twitter account are actually them speaking, but please proceed with caution. The best method that can be used to verify their identity is by them placing their key on a site such as Pastebin and then sending the link over a source where their identity can be vouched for (such as their Twitter account). After people have their public PGP key, the person could share other information such as OTR fingerprints, throwaway email addresses, other usernames, etc. by placing them inside of signed PGP messages.

It is always good practice to give your PGP keys an expiry of at most six months to keep your keys fresh and secure. After this, your signatures cannot be transferred to the new key but there is still a way to let people know you are the same person. What I do when a key is about to expire is sign my new PGP key with my most recent previous PGP key. This way, people will see that I have signed my key, and are able to check both keys to verify my identity.

This method of verification is probably not a good idea if your previous key has been compromised though. Once a key is compromised, the person who compromised it could do anything with it, including creating a new key and signing it with your old key. In this case, all level of trust is dropped for your old key and you should start over fresh.

I hope at least one person out there takes something away from this, and if anyone has other methods of identity and encryption verification, please email me at xnite@xnite.org (*please* include "2600 [volume#]:" in the beginning of the subject line).

For those of you out there with XMPP and OTR, here's my username/fingerprint info: `http://pastebin.com/NPX4ZM50`.



Asterisk: The Busy Box

by MasterChen
infoinject@gmail.com

After turning my Asterisk PBX server into an apartment gate opener, I had an idea to bring back the old school phreaker busy box. I got into the phreaker scene right as the old text files were becoming obsolete, so this trick here is my tribute to the old days. As always, have fun, but not at the expense of others.

Our goal here is to make the line of our target go busy so they cannot make or receive calls. Maybe we know the target is expecting a call from a prospective employer. Maybe it's April 1st and we want to troll a few people. The list is potentially endless. We are going to need to write a bash script, a call file, and a context in the Asterisk dialplan that will handle the target *if* the call goes through and is answered.

We will start by building the call file that I have named testcall.bak (more on file extensions later on when we address the bash script).

call file code

```
#The first line states the
➥ channel we want to use, the
➥ target number, and our SIP
➥ provider's outbound call
➥ function
Channel: SIP/7025811212@vitel
➥-outbound #phone number changed
➥ to protect privacy
MaxRetries: 50
RetryTime: 2
#MaxRetries are high and Retry
➥Time is low to prevent target
➥ from answering while keeping
➥ action on the line
Context: testing
Extension: s
Priority: 1
#The above three lines direct
➥ the call file to a precise
➥ point in the dialplan if the
➥ target actually answers
```

We save the call file as testcall.bak instead of testcall.call because Asterisk deletes the call file upon completion of the call. We want repeated use of the call file, so we save it as a .bak and then handle multiple copies of the file with the following bash script.

bash script code

```
#!/bin/bash
counter=$1
while [ $counter -gt 0 ]
do
  cp testcall.bak testcall.call
  chmod 777 testcall.call
  mv testcall.call /var/spool/
➥asterisk/outgoing
  counter=$(( $counter - 1 ))
done
```

The counter is the number of copies of the call file we want to make. We set this high as well as "MaxRetries" in the call file in an effort to keep the target's phone line busy. With these numbers high, we account for call waiting and, if the call is answered, we can still send more calls to keep the line busy thereafter.

Our last step is to make a context in the dialplan to play a sound file if the target does answer one of the calls.

```
[testing]
  exten => s,1,Answer
  exten => s,2,Playback(/var/lib
➥/asterisk/sounds/tt-weasels)
  exten => s,3,Hangup
```

One great benefit to this setup is that, unlike the original busy box, this will work on both landlines and cell phones. We also do not have to attach any physical equipment anywhere, so not being seen is a plus. So, this is my tribute to the old school. I hope you enjoyed it thoroughly.

Shoutouts to telephreak.org and all of the other ninjas here and abroad.

Using Square from Outside the U.S. to Obtain Dollars at a Reasonable Rate

by R. B.

I live in Argentina, an unstable country in South America with outrageous corruption, a high inflation rate, and permanent currency devaluation. In this scenario, if you are lucky enough to save a little money, the only way to maintain the purchase level is to change saved local Argentine pesos to a strong and more stable currency like U.S. dollars.

But Argentina has a curious currency market with an official dollar exchange rate that nobody is authorized to get and a free exchange rate that is way too high (almost 80 percent more than the official rate). As an example: with 1,000 pesos you can get USD $100 in the free market or USD $183 at the official rate. (Exchange rates, perception, and policies are constantly changing in Argentina.)

When you purchase with local credit cards using a foreign merchant, the exchange rate is, of course, the official one, plus a recent fee of 15 percent named "Percepcion RG 3378/12."

My objective was to generate a schema where I can pay to myself with a credit card in order to obtain dollars at an intermediate currency exchange for savings. While traveling in the U.S., I had signed up for an account with Square and connected the Square card reader to my Samsung Galaxy S2 running Android. I was able to purchase in Argentine pesos and those pesos were exchanged to U.S. dollars at the official rate plus 15 percent, and available the next day in my account.

However, when I returned to Argentina, the Square card reader refused to process any transaction due to geolocation restrictions. Reason: Square merchants should be located in the U.S.

The question was: how could I let Square think that I was still in the U.S.?

I installed a Samsung USB driver in my Notebook, then downloaded ODIN 3.04 and the ROM `CF-Root-SGS2_XX_XEO_LPQ-➥PROPER-v5.4-CWM5.tar`, and rooted my Samsung Galaxy S2.

The next step was to locate an application that overwrote my real location in rooted Android with a location in the U.S. I was happy to find an app named Location Spoofer. There is a free version that allows you to set up any location with latitude and longitude or map a selection for one minute. That time is enough to authorize and charge a card, so I have used the free version.

So right now, from Argentina, with a rooted Galaxy S2, a Square card reader, and free Location Spoofer app, I´m able to purchase to myself in order to obtain U.S. dollars at a reasonable rate for savings.

Governments and companies always like to talk about globalization, but regionalization restriction is what you really get from them.

Useful Links

- Square: `https://squareup.com/`
- Location Spoofer: `https://play.➥google.com/store/apps/det➥ails?id=org.ajeje.fakeloc➥ation&hl=es_419`
- Root Galaxy S2: `http://www.wired➥mash.com/how-to-root-sam➥sung-galaxy-s2-i9100-jelly-➥bean-4-1-2/`
- Square Reader APK: `http://www.➥androiddrawer.com/7862/down➥load-square-register-2-5-1-➥app-apk/#.UeWyFo09-HM`

TELECOM INFORMER

by The Prophet

Hello, and greetings from the Central Office! I write to you from the desert Southwest, where I am spending some time in an Arizona central office before starting my new management career. It is sunny outside, but I am busily applying skills learned in my management training toward "maximizing deferred maintenance asset value," otherwise known as "don't actually fix anything, but try to keep customers paying their bills for as long as possible anyway." It's almost exactly the kind of work I was doing years ago, except I was the person implementing the plans instead of making them. It feels good to be back in the saddle again. In the next few months, I will travel around the world clockwise, from the U.S. to Amsterdam to Croatia to China and then back to the U.S. again. I'll be busy "maximizing" a lot of "value" and my new career is going to be great. I expect to double my previous salary!

Having spent the last several months in Central America, I became interested in seeing more of South America. This is a part of the world where I haven't spent much time, and the opportunity to visit arose when I found a "mistake fare" from Phoenix to Quito, Ecuador. It was under $400 for the trip (usually fares to Ecuador are triple that) and, best of all, I could stay overnight in Mexico City on the way back. The opportunity to explore the telecommunications landscape in two countries was too exciting to pass up, and I immediately booked the ticket.

Ecuador isn't a place where many Americans visit, even though they use the U.S. dollar. It is a friendly, clean, politically stable, and rapidly modernizing country. A few years ago, it was difficult to get Internet access, but today, access is available throughout the country via ADSL and 3G. In some larger cities, broadband is also available via cable modem. Speeds are fairly slow; 1.5Mbps seems to be the norm. However, pricing is reasonable, with the typical household paying around $25 for a basic Internet package. Service is available throughout the country, even in very remote areas. The Ecuadorian government considers the availability of Internet access to be a national priority, and this has been one of the heaviest infrastructure investments bringing good quality connectivity to nearly all areas of the country. Free Wi-Fi Internet access is widely available in public areas, such as libraries, city halls, and even museums.

Mobile phone adoption runs below other countries in the region, largely owing to the exceptionally high tariffs on handsets which adds over 100 percent to the cost versus the United States. Ecuadorians have also largely failed to join the smartphone revolution because they are priced out of the market. Even the most basic of Nokia handsets costs around $50, a large sum in a country where a mid-level manager makes only $1200 per month, and a worker makes half that. Given the high price of buying a handset, carriers keep the cost of a SIM card very low to encourage adoption. A new SIM card costs $3 and typically includes $5 in calling, although the rates are expensive (about 28 cents per minute for calls and five cents each for texts). As in many countries, you can subscribe to service packages which include text and data service, and also international calling. Internet service is fairly expensive, costing $10 for 500MB of data.

There are technically three mobile phone providers in Ecuador (all running GSM networks), but effectively only two. One of the licenses is held by the former government telecommunications monopoly, who has failed to invest in their network. Coverage is limited and 2G only, and subscriptions can only be done on a contract, so this company now has less than one percent of the market. The two largest providers are Movistar and Claro, both multinational providers who operate throughout Latin America. In Ecuador, Claro has the largest network with the best coverage and fastest data service. However, the service is considerably more expensive than Movistar, so I chose

Movistar as my mobile provider. The coverage proved adequate in the areas where I traveled, although I definitely noticed gaps, and 3G coverage dropped to 2G outside of cities. The speed of data service was generally poor, and it was not fast enough for Skype - not even in the business district of Quito.

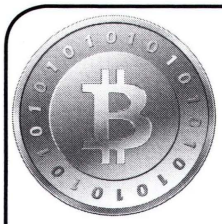
What I found especially interesting - and so incredibly different from most other places in the world - was the prevalence of payphones and other public calling services. Owing to the slower adoption of mobile phones, these are not being removed in Ecuador; in fact, many are newly installed. Some are operated by the former government telecommunications monopoly (this company operates under different names depending on the region of Ecuador - and their payphones use land lines), but Claro and Movistar also have wireless payphones. If you use a Claro payphone, the rates are really cheap to call a Claro mobile phone, and Movistar payphones are cheap to call Movistar mobile phones. If you want to call a land line, you get the best rates on a payphone from the land line provider. You very often find three payphones all in a row, one from each company. All payphones in Ecuador charge by the minute, most take only prepaid smart cards, and the rates are around 10 cents per minute. While it is possible to make international calls from payphones, you really wouldn't want to; the prices are just as high as making international calls from a mobile phone without a service plan (around 60 cents per minute to the U.S.).

International calls are where "cabinas" come in. These are shops throughout Ecuadorian cities where you can make phone calls. They are outfitted with a half dozen or so small rooms equipped with a bench and a telephone. There is a door that closes for privacy, and you can make calls anywhere in the world. In Ecuador, these aren't scary and dirty places like they can be in other parts of the world; Ecuadorian people are very clean and the "cabinas" are generally maintained in a spotless condition. Most of these shops use a VoIP service on the back end (typically a SIP provider). I made test calls to the U.S., Canada, and China from different shops and the quality was - to my surprise - uniformly excellent to all of these countries. You leave your ID card or passport with someone at the front desk, make your call (billed at low international rates - for example, six cents per minute to the U.S.). and then pay for the call after you finish, whereupon your ID is returned. These shops are all over the place and most Ecuadorians use them when they want to make an international call because it's the cheapest way for most people to make a call. To illustrate just how common this is, one of the largest banks in Ecuador (called Banco Pichincha) operates a chain of these shops throughout various Ecuadorian cities and they can directly debit your bank account for the price of the call. Many of these shops also offer Internet service, printing and copying, bill payments, and "recargas" prepayments for mobile phone service.

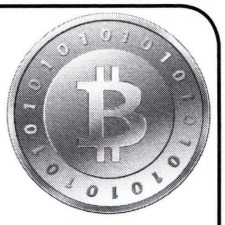
After a thoroughly enjoyable two weeks in Ecuador, it was time to head back to the U.S. On the way back, I had the chance to stop in Mexico City to get a taste of telecom in one of the world's largest cities. Mexico City is a lot like Beijing - smoggy, heavy traffic, and both the political and cultural capital of its country. Fashionably dressed people carry the world's most modern smartphones, with a particular affinity for sleek models from Samsung. Mobile phone service is offered in both prepaid and contract form. As you ride the subway, you'd be hard-pressed to distinguish that you're in Mexico and not New York or Paris.

There is a difference in Mexico, though: payphones are flourishing there as well. None look new, but the ones in place aren't going away and compete vigorously for business. Not only are there TelMex fortress phones everywhere, but COCOTs do a brisk business too. Many COCOTs even offer innovative services like web browsing, email, and SMS messaging! "Fifty percent cheaper than calling from your mobile," beckons one ad pasted to a public phone. "Unlimited duration, flat rate!" beckons another. Payphones cannot compete on convenience, but they apparently can compete on price. I was surprised to see young, cost-conscious consumers making use of them.

And with that, it's time to head out into the Arizona sunshine and hit the golf course with the execs. If you notice an "enhancement" in your billing statement introducing a new requirement that you buy a land line along with your ADSL service, don't forget to thank me. If you don't thank me, I might introduce some "new rates" in your next statement as well. I'll be at HOPE this summer in New York City, and I'll look forward to meeting you. And don't forget... Internet service is an unregulated "Information Service!"



ROBBING THE RICH USING BITCOIN



by 0rbytal
0rbytal@burntmail.com

This is a concept article (i.e., I haven't actually tried this), and is for educational/informational purposes only. You *should not* steal from other people, no matter how much they have, or how little you have. You have no *right* to someone else's property or the fruits of *their* labor. In this article, I am simply illustrating how thieves *could* exploit a couple of the most common security flaws still found in practice today. I *love* the concept behind Bitcoin, and I also enjoy the convenience of online banking... so, *please* don't abuse them by using this methodology to your financial advantage!

This "attack" is founded upon the classic, yet prevalent, blunder of using weak passwords, and the all too common habit of repeatedly using the email address for personal activity. If you have a very strong (i.e., difficult-to-guess or brute-force) password, or you don't do *any* online banking, you should not be vulnerable to this sort of attack. I'm merely speculating on a new method by which a technologically-savvy thief could rob an online banker, and likely avoid getting caught.

Online banking has been around since the early eighties, and millions of people around the world take advantage of its convenience every day. Unfortunately, many of these online bankers do not like having to remember complex passwords to access their account, and therefore resort to using a weak password that could be guessed or brute-forced will relative ease. These security-ignorant online bankers also tend to be the kind of people who use the *same* email address to register/access all of their important accounts. So, once an attacker discovers someone's email address, it is not a stretch to assume the attacker can use that email address to access the associated bank account - *if* they know the password.

By learning more about the target, the thief could use a script that scrapes certain websites to generate a list of potential passwords. *[NOTE: such a script can be found following this article.]* If a thief discovered a bank that allowed unlimited login attempts, or the thief was patient enough to try logging in three times per day until they succeeded, she could execute a custom, brute-force login script that tried commonly-used and "relevant" passwords (from the generated list) until she gained access to the targeted account. Now, once a thief has attained access to her target's online bank account, if she started transferring the target's funds to her bank accounts, she would surely get *busted* from the logs and audit trail. Enter Bitcoin....

Bitcoin allows relatively anonymous transfer of funds with almost no way to determine the sender or recipient. There is a long hash value associated with each Bitcoin user's account, and unless the user has associated their email address with their hash value somewhere on the Internet (e.g. forums, blogs, personal website, online vendor, etc.), the Bitcoin user can be fairly confident that nobody will discover their identity based solely upon the hash value (i.e., their Bitcoin "account number").

So, back to our thief... if the target already has a Bitcoin wallet, there's a possibility the email address and password used to access the target's bank account will also work to access the target's Bitcoin wallet (assuming it's online, and not a local client). Otherwise, the thief could create a Bitcoin wallet *for* the target, just to convert transferred funds from the bank account into Bitcoins. Once the funds are converted to Bitcoins and sitting in the target's new Bitcoin wallet, the thief can send all of those Bitcoins to his own Bitcoin wallet, and then convert those stolen Bitcoins back into another fiat currency stored in his own bank account.

Using this technique, a thief could steal money from one person's bank account, and put it in her own bank account, without anyone being able to track it. The lessons I hope everyone learned from this article:

(1) *Use difficult-to-guess/brute-force passwords*. I suggest using a password safe (e.g. KeePass), generating a 20+ *random* character password for each account you have, and storing the encrypted database in your favorite cloud storage. This way, all of your passwords are different, random, and accessible anywhere you go (via the KeePassDroid app), so you

don't even have to remember them! Plus, if one of your passwords is compromised, *all* of your other accounts aren't automatically compromised as well.

(2) *Use different email accounts*. By using different email accounts to register for services, if someone gets one of your email accounts, they don't automatically know your username for *every other account* you have.

(3) *Beware what you share*. If you have a Bitcoin wallet, and you share your hash on the Internet, you've just associated your identity to that account.

(4) *Use two-factor authentication*. If you use a service that offers two-factor authentication (e.g. DropBox, Gmail, etc.), you really should enable this feature so you are notified if someone is trying to access your account without your permission.

Stay curious, stay safe, and Hack *All* The Things!

```
### wordlistgenerator.py by blerbl
import re, sys, os, urllib
#### custom useragent
class AppURLopener(urllib.FancyURLopener):
    version = "Mozilla/5.0(compatable;MSIE
9.0; Windows NT 6.1; Trident/5.0)"

urllib._urlopener = AppURLopener()
uopen = urllib.urlopen
uencode = urllib.urlencode

#################################################################
###
### Helper Function
###

def ls(file):
    print(open(file,'rb').read())

def google(query,numget=10,verbose=0):
    numget = int(numget)
    start = 0
    results = []
    if verbose == 2:
        print "[+]Getting " + str(numget) + " results"
    while len(results) < numget:
        print "[+]"+str(len(results)) + " so far..."
        data =
uopen("https://www.google.com/search?q="+query+"&start="+str(start))
        if data.code != 200:
            print "Error " + str(data.code)
            break
        results.extend(re.findall("<a href=\"([^\"]*)\"
class=(?:l|s)",data.read()))
        start += 10
    if verbose == 2: print "[+] Got " + str(numget) + " results"
    return results[:numget]

def genWordlist(targetlist,word_reg,outfile,verbose=0,quotes=True):
    quote_reg = re.compile("\"([^\"]{2,35})\"")
    ###
    ### Initialize Engine
    ###
    words = []
    append = False
    total_wb = 0
    dircount = 0
    totalcount = 0
    ###
    ### Read the old list
```

```
    ###
    if outfile.startswith("+"):
        outfile = outfile[1:]
        words = open(outfile).readlines()
        append = True
        total_wb = len(words)
    ###
    ### Hit the sources
    ###

    for target in targetlist:

        data = None
        ###
        ### Get the data
        ###
        if os.path.isfile(target):
            data = open(target).read()
        elif os.path.isdir(target):
            dircount += 1 # for stats in end
            subtargets = os.listdir(target)
            for subtarget in subtargets:
                if os.path.isfile(subtarget):
                    data = "\n\n" + os.read(subtarget)
                else:
                    targetlist.append(subtarget)
                    #We will get it the next time around
        else:
            try:
                res = uopen(target)
                if res.code != 200:
                    print "[!]Error: " + str(res.code)
                else:
                    data = res.read()
            except Exception as e:
                print "[!]"+str(e)

        totalcount += 1
        if not data:
            if verbose: print "[-]No data from source: " + str(target)
            continue
        else:
            if verbose:
                sys.stdout.write(str(totalcount) + " of ~" +
str(len(targetlist)) + " sources complete\r")
                sys.stdout.flush()
            else:
                pass
        ###
        ### Format the data
        ###
        data = re.sub("(<!--|-->)"," ",data) # keep comments as normal
            ➥ text
        data = re.sub("</?[^>]+>"," ",data) # remove the html tags

        data = re.sub("\r|\n"," ",data) # make it a strait file

        ###
        ### Add the new words
        ###
        allwords = word_reg.findall(data)
        allquotes = quote_reg.findall(data)
        for quote in allquotes:
```

```
            allwords.append(quote)
            allwords.append(quote.replace(" ",""))
            #flw = ''
            #for each in quote.split(' '):
            # if len(each) > 0: flw += each[0]
            #if flw: allwords.append(flw)

        for word in allwords:
            ###
            ### Mangle
            ###

            if( word.endswith('.') or
                word.endswith(',') or
                word.endswith('!') or
                word.endswith('?') or
                word.endswith(';') or
                word.endswith('"') or
                word.endswith('\'')):
                allwords.append(word.strip('.,!?;"\''))
            if re.match("\A.*\.(jpg|png|txt|com|html)\Z",word):
                allwords.append(word.rsplit('.',1)[0])

            ###
            ### Add
            ###
            if not word in words:
                words.append(word)

    total_wa = len(words)
    total_s = len(targetlist)
    words.sort()
    of = open(outfile,'w')
    for word in words:
        of.write(word+"\n")
    of.close()
    if verbose:
        print "[+]Complete!"
        print "[+]"+ str(total_wa) + " words in the list."
        if append: print "[+]"+str(total_wa - total_wb)+" are new."
        print "[+]Collected from " + str(total_s - dircount) +
              ➥ " sources."

if __name__ == "__main__":
    ###
    ### User input
    ###

    verbose = 2
    minlen = 6
    maxlen = None
    find_quotes = True

    wordrules = ["A-z","A-z0-9","A-z0-9*-.!$#@%"]

    wordrule = None
    while not wordrule:
        print "Select a word rule:"
        for i,rule in enumerate(wordrules):
            print str(i + 1) + " -- " + wordrules[i]
        print str(i+2) + " Custom (WARNING: ADVANCED!! not validation)"
        que = raw_input("Rule[1-"+str(i+2)+"]:")
        try: que = int(que.strip())
```

```
        except: que = -1
        if que == i+2:
            wordrule = raw_input("Wordrule:").strip()
        elif que < 1 or que > i+2:
            print "Not a valid selection"
        else:
            wordrule = wordrules[i-1]

    if not minlen: minlen = 3
    outfile = raw_input("Filename:")
    if os.path.exists(outfile) and not outfile.startswith("+"):
        que = raw_input("[?]This file exists! Overwrite[y|N]:")
        if not 'y' in que.lower():
            exit(0)
    targetlist = raw_input("Input target list, separate by ';'
            ➥ no space or
quote\n"+
                           "Use %g<query>%<numresults> to use google
                                ➥ query
sites\n"+
                           "Targets:")
    targetlist = targetlist.split(';')
    for target in targetlist:
        if re.match("%g[^%]+%[0-9]+",target):
            if verbose == 2: print "[+]Google sources: " +
target[2:].split('%')[0]
            new_targets =
google(target[2:].split("%")[0],target[2:].split("%")[1],verbose)
            targetlist.remove(target)
            targetlist.extend(new_targets)
    if verbose == 2:
        print "[+]Gathering data from the following targets:"
        for target in targetlist: print "[+]"+target
        print "============================================="
    ###
    ### Prepare and call
    ###
    word_reg =
re.compile("(["+wordrule+"]{"+str(minlen)+","+str(maxlen)+"})")
    genWordlist(targetlist,word_reg,outfile,verbose)
```

The Night the ATM Went Down on Me

by the Piano Guy

I've been underemployed for a while now. One of the companies that I applied at was a bank that actually has their world headquarters within scant miles of my house. The commute to work would be more than a bike ride, but not much.

I originally applied for IT security positions. The person in charge of that couldn't be bothered to give me the time of day. At an ISACA meeting, I finally caught up with some other employees from this bank who got my resume in front of the hiring manager. I know this because I received a phone call being told, in essence, that I didn't have the skills and that I shouldn't bother her again.

Fast forward a few weeks. One of the contract houses sent me to an interview at this bank's corporate offices. Not to do IT security,

but to do break-fix and phone support. I figured it was a foot in the door. The guy who interviewed me was very sharp, and told me that the position that I was sent for was much below my skills, but that I should apply for the security openings. Having told him that I had and that I was being ignored, I further expressed interest in working for him so I could get my foot in the door. He told me that he would love to do that, but he'd not be doing me any favors if he did. As it turns out, they start people as low as they can, and no matter how much their skills jump, the raises are small.

I lay all this out to make it clear how management thinks at this fine bank.

I don't bank with this bank, but I do like using their ATM for deposits. It is one of the more modern NCR models. It lets you deposit checks one at a time and does not require an envelope. It tries to do OCR on the check and offers what it thinks is the amount of the check (and it is usually right even for nicely hand-printed numbers). It prints a picture of the check on the receipt, which helps me keep track of who as well as how much and when.

I took care of a few computer clients that day, and then ran off to a music rehearsal. I realized before I got home that I still had one check in my pocket, and I thought it would be wise to deposit it that night before going home to bed.

My deposit went perfectly fine. I put in my card. I put in my PIN. I put in the check, which it properly read. I got my receipt, wrote the name of the client on it (belt and suspenders), and put my ATM card away.

As I was about to drive off, the ATM screen flashed, and then went black. This was at about 10:30 at night. I thought to myself "hey, at least I have a receipt, and it probably finished my transaction before it died." I decided to stay and see what I could learn by watching it go down.

The screen came back to life, and then a Windows XP splash screen came up. Windows was shutting down. I was astounded. This ATM obviously does things that other ATMs which are less modern don't do, yet they still use an OS that is about to hit end of life - forever. Microsoft recently did a big publicity push to make sure that people realize that using Windows XP makes you eligible for zero days - forever. I'm now less inclined to use this ATM.

I figured that after Windows shut down, I might get to see more. Yes, the camera was watching me, but it isn't illegal to watch an ATM shut down. (I know this ATM has a camera, because I have a picture of myself from it. I had a bank deposit that I had to make pursuant to an estate I was settling, and the receipt didn't print. When I got the replacement from the bank, it had my picture on it, taken from when I was sitting in front of the ATM.)

Once it started to reboot, I got to see just how old the CPU was. The system was running the NCR extended BIOS from 2004. When I got to see all of the BIOS spit up on the screen, I saw that the unit had many USB ports.. I also figured out that Windows XP is loaded on a CD. The OS didn't look particularly customized. It looked like a standard Windows startup.

Then, it started to make a lot of clicking and whirring noises. I could tell that it was printing a journal of some kind, as it went on and on. I could also hear clicking and whirring that made me think that it was taking money and offloading it out of the ATM to someplace else. This would not be a bad thing to do if it were possible, as people have been known to wrap chains around the ATMs and drive off with them.

I noticed a series of front panel light flashes. It was going through its own little POST. Then a script window popped up a couple of times. I figured I'd have my ATM back soon.

Alas, it wasn't meant to be. The system came up with a blue colored screen (which is not a BSOD) with the message stating that the terminal was currently unavailable, and that they were sorry for the inconvenience. Then the journal started printing again.

I know more about what I don't know. I do not know if this cycle happens at 10:30 each night to reconcile. I do not know if I was the lucky last depositor before the ATM filled up and had to offload deposits. I do not know if the ATM came back without intervention, as I had to get up early the next morning, and couldn't stay to stake it out. That, and I knew I was on camera and being recorded, and had been there several minutes already. I do not know if this is simply breaking down and having enough sense to shut down and stay that way until someone could resolve the issue.

What I do know is that these ATMs use vulnerable software (Windows XP), and that the bank's desire to keep up with the technology times is similar to their management philosophy.

I also know that I plan to start using a different bank's ATM.

ANDROID REVERSING BOOTCAMP

by Andy G (@vxhex)

So, you've built your first Android application. Now what?

This is a brief introduction to Android application reversing. It assumes a basic knowledge of Java (packages, classes, etc.) and the Android SDK (activities, intents, and the manifest). If you're new to Android development, it'd be helpful to read through some of Vogella's excellent tutorials. [1]

Most of the tools we'll be using are available in the "Reverse Engineering" section on the latest BackTrack. [2]

Reversing engineering can violate some EULAs. It can be used for malicious or legitimate purposes. Be careful what you hack (or who you talk to about it).

First Things First

Android apps are packaged into an APK (application package) file for distribution. APKs are based on Java's JAR format: they're zipped archives containing the app's manifest, resources, and code. Like JARs, you can unpackage them with any zip archive manager.

To get our hands on some APKs, we'll be using ASTRO File Manager, available in the Google Play store. Astro allows you to "back up" your apps by saving them to your device's memory as an APK. In Astro, navigate to the Application Manager, select an installed app, and click "backup." The APK will be saved to backups/apps/. From there, you can upload it to your dropbox, email it to yourself, or USB it from your device.

Other methods exist for acquiring APKs (like scripts for the Play store and ADB pulls). If you're interested in trying these out, flex your Google-fu and let me know what worked best for you.

XML Xcitement

Now that we have some APKs, let's unpack them using apktool. Apktool is a program for unpacking and repacking APKs. You can unpack an APK with:

```
apktool d application.apk
```

This will create a folder containing the unpacked APK's components.

AndroidManifest.xml is a good place to start. [3] Here we can check permissions, services, and the app's main activity.

An app's starting activity will have an intent-filter listing an action of `android.intent.action.MAIN`. An app is permitted to have multiple entry points, but it is common to see just one. Make a note of the app's starting activity, as that will be the starting point for our code analysis.

The res folder contains the app's resources, like icons, menus, and strings. Android encourages storing strings and values in XML files instead of hardcoding them into your application, and these can be found in `res/values/`. Menus, also defined in XML, are found in `res/layout/`.

An assets folder may also be present, containing miscellaneous files used by the app.

Reading Some Code

It's fairly easy to reconstruct decent Java from an APK. The Java typically won't be perfect, but it's readable and lets you examine the app's logic.

First we'll convert our APK to a JAR using dex2jar.

```
d2j-dex2jar.sh application.apk
```

This will produce a JAR file, named application-dex2jar.jar, that can be reversed like any other Java application.

We'll use JD-GUI to look at what we've got. [4] Although it doesn't come standard on BackTrack, JD-GUI will run out-of-the-box. Just extract the tarball and click the "jd-gui" icon to run. From here, head to `File->Open`, and select the newly-created jar. This will load the app into the decompiler and you should see the packages laid out in a nice tree to the left. You can start from the main activity's onCreate() method and work your way through the application's flow.

If you don't want to install any new software, you can use a Java decompiler called jad. We can unzip the jar file, explore the package structure, and run jad on the .class files we're interested in. This will produce .jad files that contain the class's Java code. From here, you're free to grep away.

```
unzip application-dex2jar.jar
jad com/package/application/
➥*.class
grep onCreate *.jad
```

That Was Too Easy

Let's head back to apktool's unpacked stuff and check out the "smali" folder. This folder contains the decompiled bytecode of the application. Its folder structure represents the various packages that make up the app, and the .smali files can be opened with any text editor.

Smali is an assembly-like translation of the Dalvik bytecode. This normally sits inside of the APK in a file called classes.dex. Because smali is a direct translation of the app's code, once you understand how it works, you can edit these files to modify the app. This is commonly how APKs are cracked or repackaged with malware. Conversely, it can also be used to remove advertisements or malicious payloads. This ability to edit and repackage an APK makes Smali worth diving into a bit deeper.

Smali Syntax

This article won't make you fluent in Smali, but this should give you enough information to start hacking on things. Keep a reference guide open as you work. [5]

Smali uses single characters to represent Java's primitive types.

```
Z - boolean
I - int
C - char
V - void
B - byte
F - float
D - double
J - long
S - short
```

Arrays are represented as a "[" before a variable type. For example, "[[I" would be a two-dimensional array of ints.

Methods follow a format of methodName (parameters) returnValue. For example, here's a method that takes a char array and int as parameters and returns a boolean:

```
Smali: method([CI)Z
Java: boolean method(char[], int);
```

Objects are represented with a capital L followed by the object's package and name. For example, an object of Java's String class looks like:

```
Ljava/lang/String;
```

L designates the object, java/lang/ is the package name, and String is the class itself. Object attributes appear as Name:Type. An object's methods and attributes are accessed using the -> operator.

Comments can be added by starting a line with a # character.

Smali Instructions

Smali instructions are human-readable representations of Dalvik opcodes. A reference will usually be necessary to look up exact syntax and functionality of an instruction, but you can generally infer what's happening. [6]

Like assembly, Smali instructions operate on registers. These are represented by a letter, indicating the type of register, and a number. Registers starting with a v, like v2, are local registers, while a p indicates a parameter register.

Smali Examples

Now let's look at some examples and break down each one.

```
if-nez v0, :label_name
```

The if-xxx statements are conditionals. if-nez stands for "if not equal zero." This will evaluate to true if our target, v0, is not equal to zero. :label_name is the label for the block of code we'll jump to if our condition is met.

```
:label_name
const-string v0, "v0 has a non
➥zero value."
```

This is a labeled block of code that moves a string constant into the v0 register. This block of code can be jumped to by referencing label_name. After this operation, we can use this string by referencing v0.

```
invoke-virtual {v9}, Ljava/lang/
➥String;->trim()Ljava/lang/
➥String;
move-result-object v9
```

invoke-xxx statements are used to call methods. In this code, Java's trim() method is called on the String object located in v9. The resulting String object is then moved into v9, overwriting our original. The v9 register is our reference to Java's "this," or the calling object. The method prototype follows the syntax previously described: the calling object type (String), the method (trim()), then the return object (also a String). move-result-object then moves the previous instruction's return value into the designated register: v9.

Smali can be a bit overwhelming in large doses, so again grep is your friend when hunting for specific functionality. Otherwise, start in the main activity and look for the onCreate method:

```
.method public onCreate(Landroid
➥/os/Bundle;)V
```

After you make changes to an app, you can rebuild it using:

```
apktool b UnpackedAPK
```

The resulting APK can then be signed [7], via Keytool and Jarsigner, and distributed for installation.

What Now?

Practice makes perfect. You'll learn quite a bit by building basic "hello world" type apps and hacking on them.

Other topics to explore include ProGuard, SQLite, OWASP's GoatDroid Project, binary reversing (for proprietary binary assets, like those used in Rovio's apps), and apktool's debugging features.

Continued Reading

Blog dedicated to android cracking: androidcracking.blogspot.com

Forum for mobile developers: forum.xda-developers.com

Android reversing examples: www.exploit-db.com/papers/21325/

References

1. www.vogella.com/articles/Android/article.html
2. www.backtrack-linux.org
3. developer.android.com/guide/topics/manifest/manifest-intro.html
4. java.decompiler.free.fr/?q=jdgui
5. code.google.com/p/smali/wiki/TypesMethodsAndFields
6. pallergabor.uw.hu/androidblog/dalvik_opcodes.html
7. developer.android.com/tools/publishing/app-signing.html

Hacking Commercial Maytag Washers and Dryers

by KingFlathead

Some background on this: If you, like myself, live in an urban highrise, you may not have your own washer and dryer. That's cool with me. I mean, all that ductwork must be a nightmare to maintain, but I'm already paying $$$ to live here, and ever since my building started advertising to students, the prices on the washers and dryers have been slowly climbing up from $.50 to $1.50.

Adding insult to injury, we used to have a machine to recharge the laundry cards accessible 24/7, but they moved the damn thing into the office. I don't want to have to show up at work in my cleanest dirty pants just because it's after 5 and I'm short a buck on my silly smartcard.

Fortunately for me, the fine folks at MacGray don't really care about anything other than emptying the cash machine - it's not like they have time to actually check the programming on 180 machines in three buildings. They won't even come out to service a broken machine that was flooding out the hallways without at least a week's notice. That means that I, a random hacker, have ample private time with the machines on my floor late at night.

Now, of course, Atmel Cryptocash is usually implemented in an insecure and exploitable manner, and exploits have been demonstrated for this, but that requires a microcontroller between the card and machine, which means a bit of embedded prototyping and a wedge card. But, since I am lazy, there is a better way: reprogram the machine in service mode to run for free.

Most Maytag commercial washers and dryers out there use a common controller platform. It dates back to the 80s and is still produced. So far as I can tell, every Maytag with a digital control panel is exploitable in this way. The identifying features are a green vacuum fluorescent display with a four-digit green numerical display and six rectangular black buttons. Washers and dryers are essentially the same, card operated and coin-op are identical in their hardware and programming.

Washers first: You will need, usually, a T-25 security bit, easily obtainable in a set for a few bucks at most hardware stores. I have also seen spanner heads and weird three-groove conical head screws, which can usually be removed with a #6 spanner. At the top of the machine, you'll see four screws holding the control panel on. Unscrew them, and remove the trim and display cover. Once those are off, usually pushing up will loosen the entire control panel assembly. Unplug the machine. On the back of the control panel, there's a connector labeled "AA1". It's a three-pin locking connector with two very short loopbacks. Remove it. This places the machine into service mode. Plug the machine back in.

You should see a new display on the machine now. The way that this works is that the code on the left is what you are programming, and the number on the right is the value for that parameter. "Woolens" advances to the next parameter, "Delicates & Knits" toggles things, and "Permanent Press" increments things. Here's a list of codes:

- *6 - Regular cycle price, in quarters - this is probably what you want to fool with*
- *7 - Wash length*
- *8 - Additional rinse*
- *9 - Cycle counter - once toggled, stays on forever*
- *1. - Money counter - also cannot be turned off once enabled*
- *2. - Special pricing - enables options 3 through 9.*
- *3. - Special cycle price, in quarters - this is*

also interesting

- *5. - Real time clock, minutes*
- *6. - Real time clock, hours*
- *7. - Special price start hour*
- *8. - Special price stop hour*
- *9. - Special price days - 1 = Sunday, 7 = Saturday*
- *A. - Vault view - used for auditing*
- *B. - Value of coin 1, in nickels - only if it's a coin-op or combo*
- *C. - Value of coin 2, in nickels - usually used for dual American/Canadian coin-op machines*
- *D. - Coin slide value, in nickels - you know, those slidey things that take four coins at once*
- *E. - Add coins option - toggles display between number of coins and dollar value*
- *F. - Enhanced pricing option - CP allows pricing per cycle, SP allows a "super cycle" for additional money*
- *H. - Super cycle upgrade price, in quarters*
- *h. - Super cycle type - 01: extra wash, 02: extra rinse, 03: both*
- *J. - Coin/debit option - leave this alone, you may not be able to change it anyways*
- *L. - Price suppression option - turns off the amount to add, just shows "ADD". Why?*
- *n. - Clear escrow - If on, clears credits after 30 minutes of no activity. Cheap bastards.*
- *r. - Spin cycle RPM - default is 800, it's probably wise to leave this alone.*
- *U. - Penny offset - used to bump the price up by pennies on a smartcard machine*
- *A1. - Prewash length - 2-7 minutes, 0 disables*
- *A2 - Final spin length - 3-8 minutes*

It's fairly evident what to do here. Set 6 to 00 for free washes, and maybe set F. to enable, H. to 00, and h. to 03 to make every wash a super wash.

If you're paranoid, or if you're in a higher traffic area, maybe you don't want the machine to be on free all the time - maybe an hour or two a week when you usually do laundry is sufficient. This is where special pricing comes in. Set 2. to enable, and make sure that you set the real-time clock in 5. and 6. correctly. Only wash between 8 pm and 10 pm on Saturday? 3.00, 7.20, 8.22, 9.7S. You get the idea.

Every change you make is committed instantly, so try not to ruin the programming and cause a maintenance call. To put the machine back into service, just unplug it, plug the AA1 loopback in, screw everything back together, and plug it back in. You may need to open and close the door and insert a smartcard a few times for it to come fully alive, and usually if there's a smartcard reader attached you'll still need a card, but it won't debit you when the cycle is selected.

Dryers are a little different, some of the codes are the same, but entry into maintenance mode is different. The ones I have seen have a circular key that actuates a microswitch. You can either use the Bic pen trick, or, usually easier is to unscrew one corner of the front panel, reach behind there, and hold it down. Unlike the washers, you don't need for the machine to be off for this. Here's the dryer code list:

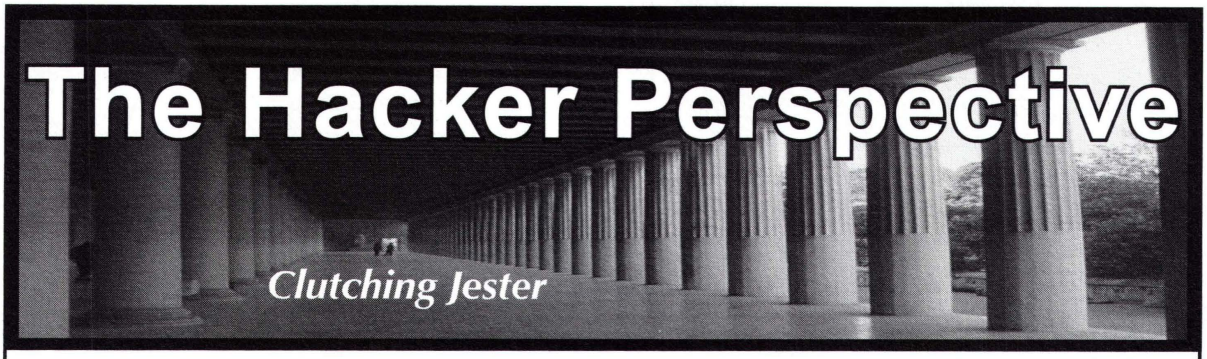
- *6 - Regular cycle price, in quarters*
- *7 - Minutes of drying per coin - free cycles count as one coin*
- *8 - Type of dry time - 00 means that you can add time to a running dryer*
- *9 - Cycle counter - cannot be turned off once on*
- *1. - Money counter - same deal*
- *2. - Special pricing option - same as the washer*
- *3. - Special cycle price, in quarters*
- *4. - Special drying minutes, per quarter*
- *5. - RTC minutes*
- *6. - RTC hours*
- *7. - Special price start hour*
- *8. - Special price start minute*
- *9. - Special price days*
- *A. - Vault view*
- *B. - Coin 1 value, in nickels*
- *C. - Coin 2 value, in nickels*
- *D. - Coin slide option - enables coin slide, one actuation is always one cycle credit.*

So, pretty similar, but less stuff. In my case, I had to go from 6 06 7 10 to 6 00 7 60 to keep the cycle time 60 minutes long. If you set the thing to cheap/free and the cycle time is really short, check option 7.

A word of caution, some models may have additional options to modify the cycle temperatures. *Don't be an asshole,* leave them alone. Setting the laundry room on fire is generally frowned upon.

Happy hacking, spread the free laundry love, and try not to get caught.

The Hacker Perspective



Clutching Jester

My history with computers started in the early 1980s with a custom-built IBM PCjr (I was about seven years old). It had cartridge BASIC, and I could play King's Quest, Ghostbusters, Shamus, Lode Runner, and a variety of other games that captivated my senses and imagination. I also very much enjoyed writing code (and a few years later watching the code of others via the demoscene, downloaded from various BBSes over our single phone line), and loved programming my own Zork-style adventure games in BASIC.

Fast forward to high school, with many more coded and played games under my belt, along with two or three custom computer builds, lots of BBS and ANSI-art experience, a huge collection of downloaded demoscene demos (we're talking a box *full* of floppies), and a shiny new computer lab at our school. This would have been the mid-90s, and we all loved finishing our classwork or homework in time to stick around the lab and play some Doom or Descent. Things (life, computer system regulations, people's attitudes) were much different and more laid-back back then overall and, in fact, the school's own computer professor could often be found playing a medieval-themed strategy game during class if he had finished his things to do. My friend Bill and I had gotten to know this professor pretty well (and he us) over our several years of computer classes, and we always enjoyed how he taught and what we learned. Senior year we got into doing some more advanced coding, and did so via Turbo Pascal. And, without further adieu, our story becomes interesting.

The computers in our computer classroom were all DOS-based, and booted using some kind of custom DOS-based loader and then eventually Windows 3.11. (I guess they didn't upgrade to Win95 by our senior year, but I honestly can't remember - ye olde memory does tend to fade over time.) The custom DOS-based loader was interesting, as it loaded some required network drivers (IPX/SPX) and then required one to log in with a username and password that authenticated to a central server somewhere (we didn't actually ever investigate exactly *how* it authenticated, but you'll see why), and then afterwards it loaded some other protocols and dumped the user into their custom Windows interface with all of their files and folders ready to go.

After my friend Bill and I had learned quite a bit of Turbo Pascal and had logged in to the system above enough times, we started to think it would be fun, since the login system was DOS-based, to see if we could use our Turbo Pascal knowledge to write a "clone" of the login program that actually captured passwords during the login process. We really didn't care to log in as anyone else and do anything with their accounts; we just wanted to see if we could pull it off.

Enter Alvin, stage left. Of particular note in this scenario is another gentleman in our computer class named Alvin. We had absolutely nothing against Alvin and, in fact, he was a nice enough acquaintance but not someone we knew well since he was a grade below us. At any rate, what we *did* know about Alvin was that he was *super protective* of his password, inasmuch as he would look around the room before typing, making sure nobody was watching, and then hunch over the keyboard so that nobody could see his keystrokes as he typed. *Every day.* In this day and age of shoulder-surfing being potentially even more costly, the value of this strategy certainly seems more reasonable, but back then it just made Alvin look like the ultimate challenge/target for our password-capturing login trojan.

And so, we got to work. There were several programmatic challenges during the process because the login program had certain characteristics that we needed to replicate very accurately. For one of those features in particular, reaching the standard of accuracy we required would be - how can you say - obvious. That is, one of the things the actual login program did was "beep" on an unsuccessful login. Since we had decided that the behavior of our program would be to "fail" at logging in no matter what was typed by the user, the length, tone, and style of the beep needed to be "pitch perfect," if you will, so as to not raise any suspicions. But, you can only test beeps in a quiet computer classroom so many

times before everyone, head computer professor included, starts to wonder what you're up to. So, that part of the project, as well as a few others, took some extra care, as well as some sacrificed Doom and Descent time after school in order to make sure we got everything *just* right.

Some more detail regarding the program itself: as alluded to in the paragraphs above, our main strategy was to write a program that would capture the passwords, and would do so by simulating the login prompt and then "pretending to fail" when the user logged in. We would capture whatever the user typed in the username and password fields, record them to two different text files in the Windows folder (with very convincing system-sounding names like NETWORK.SYS or IPXSTACK.DLL, or things along those lines), *and* we would encrypt that information with a straight-substitute cypher, just in case somebody happened upon one or both of the files before we were able to remove them to our external media (i.e., a 3.5" floppy disk). If I remember correctly, we did something like "two characters to the right" on the keyboard for our substitution, which certainly at least made the files not very readable!

So, after some care, time, and testing, our program was looking good! We were ready to deploy. But, at the same time, we were also paranoid. What if we put our program on one of the computers, and some random thing happened and we lost the ability to physically control the situation, and our planted code was discovered?? That would certainly lead to some site-wide restrictions for *everyone,* and the perpetrators would be asked to come forward and definitely be sought after. We certainly didn't want the story to end like that. So, we decided to make our software "self-deleting." That is, it would run itself, and once finished would remove all traces of itself (besides the incredibly clever encrypted "system" files) and it would be like nothing ever happened. Yes, that seemed good. But... we were still paranoid. What if, before it could delete itself, the program was discovered? What if, knowing that a program like ours was floating around, our head computer professor went around and pulled power plugs on the computers and thus rendered our program unable to delete itself before the machine was examined?? You see, the systems started with a series of batch files... the main one started the IPX/SPX stuff as mentioned before, and then called another batch file named LOGIN(.BAT) that was on a network drive and completed the actual network login. We needed *our* program to be called instead of the LOGIN script and, being on the network, LOGIN.BAT was sitting somewhere it couldn't/shouldn't be modified without great risk of project exposure. But the main system batch file, *that* one was generic and was running as a distinct (although duplicated) instance on every system... and it ran before the network was up. But still, even if that file was examined, we wanted our software to somehow remain "hidden" even through a thorough inspection.

It was here we took advantage of two characteristics of DOS: 1) we used the fact that DOS displayed everything in "ALL CAPITAL LETTERS" (and was case-agnostic) to hide our program in plain sight, and 2) (less interesting, I know) we used DOS' system path functionality to cause the system to execute our *fake* program instead of the real one. Regarding 1), we decided to call *our* batch file "logln"(.bat). L-O-G-L-N dot BAT. Because we discovered that, after some careful examination of a capital "i" and a lowercase "L" (see what I did there?), there was literally *one pixel* of difference. The upper right pixel was the only difference between the big "I" and the little "l". So, we could effectively "hide in plain sight" and change the main login batch file of the particular machine to run *our* program, LOGlN, and not LOGIN like it was supposed to; and even upon close examination, the presence of our program - ready to execute in the main startup file of the "infected" machine - would very likely go unnoticed. Regarding 2), each system's startup batch file already set the PATH to include the root of C: (along with a few other local directories) in the system path. Since our filename was named slightly differently than the *real* LOGIN script, "LOGlN" (with a little L) would not be found in the current directory and the system would then search the path for it. It would, of course, find it, and the magic would begin!

So, the simulated login program was looking good. It was highly accurate to the original - it beeped the same way, paused the same way, refreshed the same way, looked the same way; to anyone using it, it was absolutely impossible to tell any sort of difference between the real login prompt and ours. And now, everything else was good as well - we could install our program (using a custom-built boot floppy) on any machine in the computer lab in just a few seconds, it only made one tiny, virtually undetectable modification to the login scripts - it copied only two new files to the target system in a place that was unrelated to the other startup scripts, and it completely deleted all traces of itself (besides the encrypted payload) once it was finished. Now it was time for a real-world, real-person, non-test run of our program.

I don't think either of us had ever been so nervous. We had tested our program thoroughly

and done many test runs on different machines in the lab, but putting it out there for a "real person" to try seemed so daunting. But, nerves or not, it was crunch time. So, during lunch break one day, we rapidly finished our food and headed back to the lab. We quickly and quietly booted up one of the machines with our disk, and a few seconds later the magic was done, and we walked into an adjacent room for study hall and breathed a huge sigh of relief. Phase 1 was complete. But back came the nerves as we anxiously peered back into the lab to see how Phase 2 would go.

A little bit later, after the bell rang, another student walked into the lab after lunch for her computer class. As luck would have it (she always sat towards the back), she happened to sit right down at our infected machine. We could see her eyes over the top of the monitor, and we watched as she set her bag down, got out some papers, and then proceeded to type in her username and password. She didn't type super-fast (and was thus pretty accurate), but when she pressed Enter she was met with an unfamiliar "beep" informing her that her password was typed incorrectly. She looked confused, and typed in everything again (in what was now the *real* login prompt), and was logged right in. She shrugged, didn't give it another thought, and began doing her work. With a huge sigh of relief, we looked at each other with big smiles on our faces - it had *worked!*

Later that day, we came back to check our payload. Sure enough, there were our two encrypted files, and there was no other trace of our program ever existing. We moved them to a floppy, "decrypted" the username and password, and then attempted to login with those credentials, and lo and behold - *mission success*. Over the next few days, we tested it a few more times on a few other computers with a few other accounts, and it worked like a champ every time.

Now... with real world success under our belts, we knew it was time for our main target: Alvin. Now Alvin, while highly protective and careful, had one fatal flaw: he always logged in to the same computer. Every day. So we knew exactly where he'd be. We prepped his station before class one day, and moments later sat back and watched Alvin's own puzzled expression as the computer informed him he had typed his password incorrectly. After another scan of the room and another full-body keyboard covering, he tried again, logged in successfully, and carried on without a second thought.

Recovering Alvin's password later that day was like finding a pot of gold or discovering a long-lost ancient artifact or something, and was super-satisfying because of the overall process and challenge. Once we knew we had Alvin's correct password, we walked up to him one day after class as he was packing up his things. We each stood on either side of him, and when he looked back at me and asked what was going on, I just leaned in and quietly said, "Hey Alvin... kingdoma." His eyes met mine, and they were *huge*. He knew how protective he was of his credentials; the fact that they were known, and by only an acquaintance, seemed inconceivable. Then we just walked away (I guess we were trying to be cool, or really didn't know what else to say! ha). Mission accomplished. Alvin learned that sometimes things are not safe no matter how careful you are or how hard you try to keep them that way.

To end this story, I'll point out that we also learned about controlling the group with which you share this type of information. We told some of our LAN party friends from two grades below us about the software and they, of course, wanted a copy to examine. Bill and I discussed it, and hesitantly gave a copy to a couple of close trusted friends. But they, of course, had trusted friends who had trusted friends who had trusted friends... you get the picture. The next thing you know, somebody had used our program to capture the head computer teacher's password from his main machine. He was, of course, very unhappy, and Bill and I went up after class one day and confessed that we were the original source of the code. We explained our intent (which was simply to see if it could be done, not actually log in and use anybody else's accounts or damage or change their files) and how we had lost control of the code. Our teacher looked intently at each of our eyes for a few moments, nodded, said "OK," and went back to his desk and sat down. We never heard anything else about it until he joked with me, while shaking my hand at an awards presentation at the end of the school year, about almost giving me a blank sheet of paper for the computer science award he was handing me, because of the code incident.

What did I learn from the whole experience? The value of great friends is incalculable (Bill and I are still great friends to this day). Be careful who you trust. Self-examine and look for your own "fatal flaws." Be honest. Get to know people well enough to know their hearts, and not just their actions, because you might treat them differently if you do.

Kd]]i kdb;p,j/ tntyi[,t#

Clutching Jester is currently enjoying and living life with his wife and kids, and continuing pursuit of the notion that science, while awesome and important, just may not be able to explain everything... [383133]

Accessing Data Structures Located in a Randomized Address Space (ASLR) (how to eliminate entropy and bring the universe back to the singularity)

by Matt Davis (enferex)
mattdavis9@gmail.com

So, what is one to do when bored and needing something to stimulate the old neurons? Why, inspect memory! With that said, it was getting late one evening and I needed something to keep the brain stimulated, thus I decided to go poking around the memory space of a process. You know, hunt around for golden nuggets within a Linux process to see what shiny new things I could uncover. Now, this isn't the first time I have done this, but I noticed that evening that the glibc library had portions loaded into memory with write permissions enabled. It was then that I wondered what I could do.

Moreover, this led me to the writable portion of the random table in my process. This table is used for generating random values. Since random values are critical for security (e.g. asymmetric encryption, TCP sequence numbers, etc.), trying to manipulate that table might permit me to make such values nonrandom and insecure for applications that rely on them. An attacker can use a known value to aid their attack. Thus, manipulating the random table to produce deterministic values can compromise the security of a protocol or application. However, any program serious about security should not be using glibc for their entropy. Instead, something like /dev/urandom (Linux's driver for producing random values) should be favored. But, if your program (e.g. game) relies on randomness for a non-security dependent purpose, a simple generator like that provided by glibc should be just fine. As a note, I was not intending to manipulate such a table when exploring my process' memory, it just kinda happened.

The following is just an example of the memory space in Linux for an instance of the program "cat":

```
> cat /proc/self/maps
00400000-0040b000 r-xp 00000000 08:01 7084978    /usr/bin/cat
0060a000-0060b000 r--p 0000a000 08:01 7084978    /usr/bin/cat
0060b000-0060c000 rw-p 0000b000 08:01 7084978    /usr/bin/cat
01c18000-01c39000 rw-p 00000000 00:00 0    [heap]
7f533a263000-7f533a406000 r-xp 00000000 08:01 7081205    /usr/lib
➥/libc-2.17.so
7f533a406000-7f533a606000 ---p 001a3000 08:01 7081205    /usr/lib
➥/libc-2.17.so
7f533a606000-7f533a60a000 r--p 001a3000 08:01 7081205    /usr/lib
➥/libc-2.17.so
7f533a60a000-7f533a60c000 rw-p 001a7000 08:01 7081205    /usr/lib
➥/libc-2.17.so
7f533a60c000-7f533a610000 rw-p 00000000 00:00 0
7f533a610000-7f533a631000 r-xp 00000000 08:01 7081212    /usr/lib
➥/ld-2.17.so
7f533a67b000-7f533a804000 r--p 00000000 08:01 7113566    /usr/lib
➥/locale/locale-archive
7f533a804000-7f533a807000 rw-p 00000000 00:00 0
7f533a831000-7f533a832000 r--p 00021000 08:01 7081212    /usr/lib
➥/ld-2.17.so
7f533a832000-7f533a833000 rw-p 00022000 08:01 7081212    /usr/lib
➥/ld-2.17.so
7f533a833000-7f533a834000 rw-p 00000000 00:00 0
7fff8632f000-7fff86350000 rw-p 00000000 00:00 0    [stack]
7fff8639e000-7fff863a0000 r-xp 00000000 00:00 0    [vdso]
ffffffffff600000-ffffffffff601000 r-xp 00000000 00:00 0    [vsys
➥call]
```

Anyway, that writable portion of glibc intrigued me. What could possibly be in that writable segment of the glibc copy that resided in my process' memory space, and why? Well, the "why" can be answered pretty easily. Quite simply, a library has global variables and data that the running process is permitted to manipulate. For glibc, this data can be manipulated via calling glibc functions. For example, calling srand or srandom will manipulate a table used in generating the random values when rand or random are called. To get a better idea of what was going on, I wrote a simple C program, compiled it, and then loaded it up in my debugger (GDB). By using the features of GDB, one can quickly snoop around the memory space and see what lies within the deep depths of their processes. Upon embarking on this sort of late night exploration, I was quickly greeted by the symbol name for one of the items located in this writable memory space, "randtbl." Now, this value is both writable and loaded at an address that is non-deterministic, thanks to my kernel randomizing the address space (more on this in a jiffy). Since I was running in GDB, the address of the randtbl was static and always at the same location. Anyway, performing the following commands in GDB can give more insight about the randtbl location:

```
(gdb) x &randtbl
0x7ffff7dd50a0 <randtbl>: 0x0000
➥0003
(gdb) info symbol &randtbl
randtbl in section .data of /usr
➥/lib/libc.so.6
(gdb) info address randtbl
Symbol "randtbl" is at 0x7ffff7d
➥d50a0 in a file compiled without
➥ debugging.
```

As we can see from GDB, randtbl is a valid symbol, with the first portion of data having a value of 3 and located in the (writable) .data section of the shared library libc. We also know that my libc has no debugging goodies, but that really does not concern us too much. As a GDB fan, I should also mention one additional command useful for inspecting the process' memory space: "info proc maps", which is essentially the same information you would get if you read the /proc/maps entry for the process.

Recall that when the Linux kernel loads an executable into memory, a copy of the writable libraries that the program needs (in this case glibc) is loaded into the process' memory space. That way the process can manipulate the data and no other process will see the changes. This is memory that is only for the process, and lasts only the lifetime of the process. For shared libraries that have non-writable portions (like .code for functions) multiple processes can share the same library code, eliminating the need to duplicate library instruction and reducing the amount of memory necessary for programs to run.

As a security measure, the Linux kernel can be configured to randomize the address space of a process so that loaded libraries are located at a non-deterministic location in the process' memory space. This nifty feature prevents attackers from attacking a process at runtime by using information about known addresses in a library. With address space layout randomization (ASLR), the addresses of loaded libraries are not known until runtime and change every execution. Therefore, it would be pretty tricky to craft an exploit to target a specific address.

Now, back to the randtbl hackery. So, how can I get access to the random table and manipulate it (for research purposes of course) if I do not know its address until runtime? Possibly a linker script could allow me to alias the address, with a variable in my program. But, nah, I don't want to do that. I want to build my program and access the table without having to write a linker script. Let's keep things as simple as possible.

Instead of a linker script, I browsed the glibc-2.17 source code and found that srand makes use of this randtbl. So, I added a call to srand in my program and then hopped into GDB to look at the assembly. It seems that srand is actually wrapped by a function that passes a structure called "unsafe_state" to srand. The first two members in unsafe_state are pointers into the randtbl, as the glibc source code clearly shows.

The flow of execution is simple. My program first calls srand (actually its a glibc wrapper). Next, this glibc wrapper calls the actual srand function with the address of unsafe_state as an argument. Recall that unsafe_state contains pointers to the randtbl. srand then manipulates randtbl and returns control back to the wrapper and then the wrapper returns control back to my program.

Now, this is the key piece. The wrapper calling srand calls a function that uses the unsafe_state as the first argument. After this call is complete, srand returns immediately. srand never clobbers the register last used

to pass unsafe_state, therefore when srand completes, the user program (the portion you write) has access to this register. This means that your program can access unsafe_state and all of its contents (randtbl) by just reading the rdi register. This occurs because a 64 bit Intel x86 uses a calling convention when compiled by gcc-4.8.1 where the rdi register will contain the first argument passed by the wrapper to srand. And that register (containing the address of unsafe_state structure), is never overwritten (clobbered) by srand or its wrapper. This means that someone can obtain access to randtbl by simply calling srand, and then immediately looking at the rdi register, which should be the address of the unsafe_state variable that contains pointers to randtbl. And there you have it, the ability to access a writable randtbl located within a randomized address space! Well, the following does just that:

```
#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <string.h>
#include <time.h>

static void print_rand_
values(int n_values)
{
    int i;

    printf(">> Printing %d
➥ values from rand()...\n",
➥ n_values);
    for (i=0; i<n_values; i++)
      printf("%d\n", rand());
}

int main(void)
{
    int32_t type, n_elts;
    uintptr_t unsafe_state_addr,
➥ rand_tbl_ptr;

    /* Call srandom, which sets
➥ the rdi register to
     * the address of unsafe_
➥state glibc struct
     */
    srand(time(NULL));

    /* Read the address of
➥ 'unsafe_state' state,
➥ defeating ASLR */
    __asm__ __volatile__ ("mov
➥ %%rdi, %0\n" : "=r"(unsafe_
➥state_addr));

    /* Dereference the member
➥(second address) in the unsafe
➥_state struct */
    rand_tbl_ptr = *(uintptr_t
➥ *)(unsafe_state_addr + sizeof(
➥void *));

    /* The second member in 'un
➥safe_state' is a pointer to the
➥ second element of
     * randtbl: randtbl[1]. So
➥ we backup int32_t to get to
➥ the head of randtbl.
     */
    rand_tbl_ptr = rand_tbl_ptr
➥ - sizeof(int32_t);
    printf(">> randtbl located
➥ at %p\n", (void *)rand_tbl_
➥ptr);
    printf(">> Before clearing
➥ random table\n");
    print_rand_values(10);

    /* How large 'randtbl'
➥ can vary.
     * See glibc-2.17 source.
     *
     * Note that the first byte
➥ of 'randtbl' is a flag:
     * If the first byte of
➥ randtbl is:
     * -- TYPE_0 (a value of 0)
➥ then the table contains 0 32
➥bit integers
     * -- TYPE_1 (a value of 1)
➥ then the table contains 8 32
➥bit integers
     * -- TYPE_2 (a value of 2)
➥ then the table contains 16 32
➥bit integers
     * -- TYPE_3 (a value of 3)
➥ then the table contains 32 32
➥ bit integers
     * -- TYPE_4 (a value of 4)
➥ then the table contains 64 32
➥bit integers
     */
    type = *(int32_t *)rand_tbl
➥_ptr;
    n_elts = 0;
    switch (type)
    {
      case 0: n_elts = 0; break;
      case 1: n_elts = 8; break;
      case 2: n_elts = 16; break;
      case 3: n_elts = 32; break;
      case 4: n_elts = 64; break;
    }

    printf(">> Clearing contents
➥ of randtbl "
           "which is an array of
➥ %d int32 values...\n", n_elts
➥);
```

```
    memset((void *)rand_tbl_ptr,
➥ 0, n_elts * sizeof(int32_t));
    print_rand_values(10);
    return 0;
}
```

Now that my program has access to the random table, let's see what happens if I zero the table using memset. To see what I had done, I immediately called rand to see what value it produced. Muahah, it produced a non-random value of 0. Woohoo! I made random deterministic. Of course, this only affects the process and any child process that the compromised process creates (via fork()). If another executable is called (via exec()), then its address space is fresh, and it has a copy of the unmodified randtbl, thus it acts on an unmodified rantbl. Also note that any future calls to srand will reset randtbl and result in rand/random producing values as if nothing ever happened.

So what is the practicality of this being used as an exploit? This would require some pretty clever shellcode, as the exploit would have to inject a call to srand, perform a read to get the address of randtbl, and then zero-out the table. Why is this important? Well, most programs relying on secure uses of random numbers (e.g. TCP sequence numbers, asymmetric crypto, etc.) would/should be using a different source of randomness anyways (e.g. /dev/urandom). Further, we just accessed and manipulated a single variable, that being randtbl. Other variables in other libraries might also be accessed via this same method.

Anyway, I hope that this spiel was insightful. Now go take what you learned and see what other data in some other library you can manipulate!

Shoutouts: The ruxcon crew, count, __ben (the villain)

Resources

- glibc-2.17 source: `https://www.gnu➥.org/software/libc/`

A Little Excitement Never Hurt Anybody!

by lg0p89

Disclaimer: This is for educational purposes only. The information herein is not to be used for unlawful or illegal actions. The reader is responsible for his/her own actions.

Background

I received an email on 5/29/13 (5:29 pm). Curiously, the time stamp was 29 minutes after the sender would have closed. This was from the IRS. Whenever you see the IRS name plate, the reader generally misses a heartbeat and breath, at the same time. There is something guttural that occurs when you see the name "Internal Revenue Service" on a letter or email. There is not necessarily a mistrust issue, given the present issues with charitable organizations applying for their status, however an awareness of the immense power and ability of the entity to circumvent the U.S. Constitution at their will. Enough of this; that is an article for a different journal and time.

Nonetheless, the email was from the Internal Revenue Service. They used the picture of the upper third of the eagle adjacent to the scales of justice. Next to this were the words Internal Revenue Service. When the pointer was rolled over this, it provided the link to `www.irs.gov`. This made it appear yet more legit.

The body of the email showed there was a complaint by Demian Chavoya against myself and nine others, all with the same first name in the email address. In the email it noted the instructions on how to resolve this issue were in an attached zip file. The next three paragraphs were noting how all the involved parties had to agree to arbitration for this to be an option, the IRS had the sole discretion if the complaint could be arbitrated, and the IRS offered a binding arbitration service.

Red Flags/Analysis

First, this was in my spam folder. Generally, if the IRS is going to send you an email, it will hit your inbox. Usually they just mail the infor-

mation or request to you anyway. This was the first issue.

The email showed it was from the IRS, with the email of `fraud.dep@irs.gov`. This was sent to ten different parties, all with the same first name in the email address. It is not likely that all ten parties would have the same complaint and complaint number placed against them.

The email address was spoofed. When I looked at it, it read the email was from `fraud.dep@irs.gov`. The average person at first glance would see the IRS name and .gov extension and freak out, much like I initially did. However, I knew I had done nothing wrong (recently). The header for the email was reviewed. The IP address, 50.xxx.78.xxx, was not an IRS IP address. This email was sent from a `comcastbusiness.net` IP. The location was in Opa Locka, Florida (thank you, traceroute).

If there had been an actual complaint, there would have been the usual attachment. This would have probably been a .pdf, but could have been a .doc or .docx attachment. This, however, had a zipped file folder. I did not open the attachment since I was at a work computer without a sandbox to open this into. I did not need to add further work for the network admin. I have seen what happens to people on the poop list, and I so did not want to be there. Opening the zipped items probably would have infected at least my system and probably more, which would have made my life exciting in the short term.

The context also did not fit the situation. The email stated that the IRS had a complaint against me for my business services. I don't do business with the IRS. This did not make sense. There is also the complaint filed by a Demian Chavoya. I don't know any Chavoya. Also, there has been no work done with or for a Chavoya.

The date was also odd. Apparently, Demian Chavoya filed the complaint on 5/29/13. The email from the IRS was sent also on 5/29/13 - the same day. This is highly unlikely. For my math and statistical friends, this is not a statistically significant possibility.

When you send an email, it is relatively important that it makes sense. In the third paragraph, the email states that all parties have to agree to arbitration for this to be an option, meaning the party filing the complaint and the party that caused the complaint. The next paragraph, however, stated this was solely the decision of the IRS. This clearly did not make sense.

If you are trying to make another party believe the email is from a government entity, the sending party probably should use the updated format for their emblem. This email used their prior format that had not been used for months. This is merely me being nit-picky, but really, if you want a polished and professionally looking spoofed email, then do a minor amount of homework and have it look like it actually is from who you want to portray it is from. This creates fewer questions from the recipient, which is what you want.

What Should Have Been Done

This is for educational purposes only, as noted above.

For the person filing the complaint, it would have been better to have used a common name for a person or business. For an individual, perhaps Sam Flynn or Mary Hamilton would have been a better choice. For a business, perhaps Granger or Verizon could have been used. A person could have an interaction with one of these two entities or another large business. Demian Chavoya is such an unusual name that it automatically piqued my interest and I knew this was not correct. A name that slides in under the radar and doesn't stick out would have been much better.

There was an issue with the lack of a time lag, as noted above. There really should be a time lag between the date of the complaint and the date of the email. Everyone knows how slow the IRS is. This is well documented. This is a large machine that moves at its own pace. The IRS has its own timeline. In this case, the "complaint" was filed on 5/29/13 and the email was sent on the same day. There is no way this could have happened. I doubt even a congressional member could get this done. It would have been better to have a difference of a week or two between the complaint date and the date the email was sent out. This would have been so much more realistic.

Lastly, the content flow did not make sense. This should not contradict itself.

This was not intended as a "how to" but more as a thought exercise on how it should have been done. Let's learn from this on what to look for and use this as a teaching tool so the network admins don't have even more work to do.

Notations

Dear *2600:*

In 30:3, pixter warns that you cannot connect a power supply capable of delivering more than one amp to a Raspberry Pi, or it will destroy the processor.

Having read the cited *Nuts & Volts* magazine article, it simply states that if you use a power supply that can deliver more than one amp and you do something stupid and cause a short circuit, you might draw so much current that you burn out the polyfuse permanently instead of just tripping it temporarily. The processor will be fine and, should this happen, the polyfuse can be bypassed (as some people do already if the voltage on the downstream USB ports is too far below 5V).

If you're careful, then there is no problem using a high current power supply. I just wanted to set the record straight as the warning sounded so dire!

Malvineous

Dear *2600:*

In issue 30:2, Les Hogan fantasized in the letters section about coming back to life and scaring the crap out of his four time great grandson, Little Jimmy.

He had asked if anyone knew who has the record for having the same phone number the longest. I did a little digging around and found that the Guinness World Records folks have a... well, you guessed it, a world record for "most durable mobile phone number."

The record goes out to David Contorno, of Lemont, Illinois, USA. Mr. Contorno has had the 312-550-0512 number since August 2, 1985! The first mobile phone David owned was an Ameritech AC140 put out by Ameritech Mobile Communications. The article goes on to say that David has used Ameritech Mobile Communications ever since 1985. That must have been a heck of a contract!

I can't find who has had a landline the longest, but it seems it's common to have grandparents that have had the same phone number for the past 40 or 50 years.

In any case, I think Les should call the Guinness World Records organization and get the wheels rolling for a "most durable landline number" record: 718-513-7270.

Samuel

Just to be clear, that phone number at the end isn't durable at all, but is the actual phone number for the Guinness people. (You can tell it's not that old since the middle digit of the exchange is a one, which wasn't possible before the 1980s.) We can't imagine why David allowed his phone number to be printed by Guinness like that (we wouldn't have printed it ourselves if it wasn't already public knowledge). Regarding landline longevity, we again feel compelled to point out that the home of the HOPE conferences, New York's Hotel Pennsylvania, has had the PEnnsylvania 6-5000 phone number in the 212 area code since around 1930. Who can beat that?

Dear *2600:*

In 30:2, Les Hogan commented about phone number legacy. My father lives on the family farm, which has the same number my great-grandfather had - and he died in the late 60s. In fact, it's still in his name.

T

That opens the door to another interesting question: how many phone bills (or other bills, for that matter) have remained in the name of someone who's long since passed? It's not like utility companies come out and take pulses occasionally. We wonder what the longest period is that someone has kept a deceased relative listed as an active bill payer.

Questions

Dear *2600:*

I apologize for contacting you, however, I am writing to you as scientist in relation to my master thesis research project at Queen's University Belfast (School of Psychology). I am doing my master thesis in the field of political psychology and analyze stereotypes within the hacktivist community. As in previous interviews, participants were often referring to *2600*. I was hoping that someone within the board of *The Hacker Quarterly* might help me with my qualitative research and would be interested in participating in an approximately one-hour interview through Skype or any other preferred service.

To make this email more reliable, I would like to outline the research in detail:

L

OK, let us stop you right there. Had we printed your "outline," it would have gone all the way to the end of the entire letters section. While we support what you're doing, nobody here has the time to do this sort of thing. (We didn't even have time to skim the entire outline.) What we suggest is that you reach out to the hacker community, perhaps through one of our free Marketplace ads, and you might get some decent responses that way. But we're just too busy with magazine-related stuff and we get so many requests of this nature that this is the best we can offer.

Dear *2600:*

If this photo is not good enough quality and you'd like a better one then please let me know and I'll take another.

Rob

You should probably take another or at least remember next time to attach the one you're referring to here. It is simply unbelievable how many such emails we get each month.

Dear *2600:*

From the Fiat I rented recently in Toronto. The PIN number to connect my Bluetooth phone looked strangely familiar!

Saskman

We can only imagine.

Dear *2600:*

Found dozens of these booths all over old city Quebec. They looked rather unloved.

Drax26

Not as unloved as not being seen at all.

Dear *2600:*

Can you see it?

kmk

The fact that someone would ask this indicates that they knew there was a decent chance of their sending no image at all. Or perhaps it's more of a metaphysical question. Regardless, we don't see an image, we don't see a point, and we don't see any reason to subject our readers to more of these.

Dear *2600:*

Hey! I want to submit two articles to *2600*. The reason why I am emailing is because I want to know if the subject matter is OK before sending in the articles.

The first article I want to write will be about being polite within the hacker and tech community. I feel that many people offend each other by accident. My article would focus on how to avoid offending people or avoiding arguments.

The second article would be on how to apologize and reconcile within the hacker and tech community. I feel that many people may have tech skills but lack skills in communicating with others. Both my articles will be based on interviews from one of the keynote speakers of Devcon 5 Los Angeles. Please let me know if either topic is acceptable for printing at *2600*.

Glenn

We find ourselves offended by the suggestion that people in our community are offending other people. We hope that your second article contains a suitable apology for the suggestions contained in the first one. And we hope you also focus on the sense of humor that infests the hacker world. We look forward to seeing your articles. Seriously.

Dear *2600:*

A quick search of your archives yielded no articles on securing industrial control systems. Can you point me to any relevant articles? I purchased the last three years of annual digests but have not been provided a link to download yet.

John

Hang on. You're saying you bought our online digests and didn't immediately get a download link? How are you not filled with utter rage and threats? This definitely isn't acceptable. Please email orders@2600.com immediately so we can resolve that. Whenever you order any of our online content, you should see the blue download link in the upper right hand side of your screen as soon as your order goes through.

Regarding the archive question, the best way to search our content currently is to use the search mechanism at store.2600.com. In addition to articles, this will also give you results from all of the HOPE conference presentations.

Dear *2600:*

I am interested in locating an article you published a few years ago on hacking the Target department stores' wireless networks. I looked in the archive and was not able to find it. Could you please give me a reference location for that article. With the current news about Target, I would like to check the article out again. The news media is now stating that the hack has taken place through the credit card clearinghouses. The article would still be of interest. Thanks.

Chuck

We do seem to have had a number of articles on Target over the years, again all findable through the search mechanism at store.2600.com.

Dear *2600:*

I found really dangerous malware and servers in China (I think) that almost all anti-virus companies could not detect!

Would you please help me to complete my report and publish this news in your magazine?

M Y

We can't help you write your report, but we'd be interested in seeing what you discover. If we find it something our readers would benefit from, there's a good chance we'll print it.

Dear *2600:*

I recently went through a divorce. How would I go about changing my address?

Norman

Moving is the easiest way to change your address, regardless of whether or not you're divorced.

Assuming you didn't want a wiseass response to your inquiry, we have to try and figure out the context with which you posed the question. It's most likely you were asking us how to change the address we have on file for you for the magazine. For that, you need to contact us either by mail, email, or phone. We'll need the info that's on your address label for verification. Of course, that's something your divorce partner might also gain access to if you're not careful.

Dear *2600:*

We see you are the owner of the domain 2600.com. We are developing a project and need the domain. Please let me know if it's for sale.

Eric Lee

Well, we had a good run, but we always said that if somebody else needed the domain, we wouldn't stand in the way. And a "project" certainly sounds like something worthwhile. By the time this is printed, we will have made the transfer. So... now we need a new name.

Dear *2600:*

After several years reading *2600* off and on via over the counter purchases, I have articles to share that will interest your readers.

Article guidelines? Submission guidelines? Terms and conditions I should know up front?

From past readings of *2600*, content guidelines seem kinda loose and free flowing, and I know that asking for guidelines up front can make a big difference.

What moved me? The tone and flavor of *2600* editor responses in the letters section (Fall 2013 edition). I was amazed at the supportive and positive editorial replies, and the general positive tone and demeanor presented. I have done technical writing in several creative hostile and emotionally hostile environments. The tone and demeanor of the Fall 2013 editorial feedback was inspiring - an impressive effort to uplift your readers from their "funk" - even in the cases where your readers exhibit some emotional and technical "brain damage" to their writing approach.

We all have writer's "brain damage" - it is just a matter of degree.

Feedback welcomed.

Juan

We're glad you appreciate our "style" and hope to see your articles soon. The guidelines are simple - make it interesting to a hacker audience, write from a hacker perspective, don't be too brief, but don't be too long-winded either. The best way to see what we mean is to simply read a dozen or so articles that we've printed. As for what happens once you submit something to articles@2600.com, you should get an immediate auto-response (no more than one every few days in case you send multiple submissions). You will generally hear if it's been selected before the next issue goes to print and, if that happens, you'll get more details as to when your article is likely to be published. Sometimes we fall behind and sometimes it's lightning fast. We'll contact you after it's printed to give you a choice of various items we offer to authors. We do insist that any articles we publish not have been made available elsewhere (in print, on blogs, websites, sides of buildings, etc.) until after it comes out in our pages. After that, you're free to do whatever you wish with it as it's still your article. It may appear in future volumes or collections that we publish as well. We hope that answers your questions.

Dear *2600:*

wht is this

hello

i just got your web site on search tell me what your goals?

Tina

We would absolutely love to see just how people arrive at this stage of befuddlement. Some kind of a web search gone wrong leads them to us and their lives are, at least temporarily, thrown into confusion and turmoil. That is the true beauty of the net.

If this writer actually manages to get a copy of the magazine and sees our response, all we can say is to read what goes on inside these pages and that ought to give you at least a partial view of what some of our goals are.

Dear *2600:*

Hello? I want to get data that come from encrypted database of chat records of Tencent Weixin.

zhangganghong18

We thought you'd never ask. Seriously, what exactly do people think we do with our time? (For those who may not know, Tencent is a massive Chinese Internet company and Weixin is a chat app.)

Dear *2600:*

In the vein of Joe's letter in 30:3 about securing payphones, I am doing some research on parking meters. I mean the old school meters that take coins which are still plentiful here in New York City. I am trying to find out how the companies go about securing these parking meters, where the locks get made, and how one can go about unlocking one.

Brainwaste

We imagine this would indeed be similar to unlocking the cashbox on a payphone, particularly back in the old days when one key would work for a large number of phones. Getting a copy or a mold of a parking meter key would likely give you access to quite a lot of them. We'll print the info if we get it but strongly advise against actually opening up one of these things. There are few activities which could look more suspicious than opening up a parking meter. And, of course, the people who unlock such devices with theft in mind often tend not to think of just how heavy coins can get relatively quickly.

Dear *2600:*

I am interested in a subscription. However, it is near impossible for me to arrange a money order. Would it be possible for me to pay via U.S. postage

"forever" stamps? Obviously, I understand there may be an extra surcharge.

Michael
Federal Prison

In special circumstances such as yours, we try to accommodate when possible. As those particular kinds of stamps don't lose value, we're willing to accept them as the equivalent of cash without any additional charges. So this doesn't get out of control, these are the only kinds of stamps we'll consider taking and only as an experiment. We hope it works out.

Accusations

Dear *2600:*

We are a small business starting up a website which has just been hacked and destroyed by one of your readers. Of course, it could be anybody, but signing off "Hack2600/MFAD" does point perhaps unfairly in your direction.

On the publisher's website, the subscribers to *Hacker 2600* and the magazine itself are described thus: "Published by hackers since 1984, *2600* is a true window into the minds of some of today's most creative and intelligent people." I can hardly agree with "creative and intelligent," since all they have done is guess our moderate and temporary password while we get up and running. Lesson learnt.

It will take a fair amount of work to rebuild - time and resources which should be spent on other aspects of the business. I doubt that any of "today's most creative and intelligent people" even consider this when they do what they do in cowardly anonymity.

This is not a first for me; I've been in IT for many years and seen this many times. Hackers and spammers have over the years gradually worked their way into third place, just below National Socialist Party and pedophiles on a list of people I'd have sent to one way out of the solar system.

But who cares? I'm just some little guy trying to run a business. Oh well, back to the rebuild.

Simon
England

Wow. So perhaps every time somebody named Simon says or does something stupid, we should look your way? Just because a name is used does not mean there's any affiliation or connection to anyone else using that name. Even if you assume that somebody is a reader of ours simply because they have "2600" as part of their name, how does that equate to representing all of our readers? You don't even know the context of their signature - "Hack2600" could mean that we're the next stop on their hacking rampage. And you certainly aren't looking into the MFAD connection, whatever the hell that is. Clearly, you're not that familiar with what we're all about (for instance, we're not called "Hacker 2600"). We believe you would benefit greatly from listening to what our writers (and readers) say. You would learn a ton about security and how to avoid the kinds of things that you say you keep encountering. More importantly, you would learn not to lump a whole group of people into a category based on perceptions obtained from dubious sources. And, as you continue to work on your rebuild, take some time to acknowledge the people designing some of the software you're using because it's very likely they're a part of the very community you're condemning.

Dear *2600:*

Your whiny editorial bleatings about loving freedom of information and knowledge *might* have been believable *if* you were agitating for disclosure of the full fact-set regarding the Benghazi massacres, or of the Vince Foster papers.

But you aren't, because you ain't.

Lifetime Subscriber

One might almost suspect there's an agenda here.

Litigation

Dear *2600:*

I was wondering if you could help shed any light on legal rights around readers republishing content from *2600 Magazine*.

More specifically, there is a website that has republished a couple of articles I wrote that were published in *2600* several years back. The site hosts anti-Semitic and bigoted opinions and views that I don't agree with and don't wish to be associated with and, as such, I requested the webmaster remove those articles. He has refused to comply.

I know that *2600* says that authors retain the right to publish their articles anywhere that they'd like after they have been published in the magazine itself. Likewise, do authors retain their right to control where the content is or is not published? Does *2600* retain rights on who cannot republish?

Any help you can provide would be greatly appreciated.

Nick

This is certainly an interesting situation, one which has never come up before. We try to keep things as simple as possible without injecting a lot of legalese that tends to stifle the creative process and keep material from being shared. In general, if someone asks us if they can reprint something from one of our issues, we permit it provided they give attribution. If, however, one of our writers specifically requests that an entity outside of 2600 *not be permitted to do this, we will honor that request and not grant such permission. Assuming we didn't already give permission for reprinting this material, your wishes should be followed. It gets a bit sticky, however, when someone refuses to honor such a request. You could go after him legally, but odds are that would cost you money and get him publicity, even if you won. It being the Internet, it can be impossible to remove content and, often, attempts*

to do so wind up backfiring. We think there must be a more creative hacker-inspired solution to this. We ask our readers to help us come up with ideas.

Dear *2600:*

Just a heads up: your YouTube videos aren't available in all countries. Can you resolve this?

Very Anonymous

The way Google/YouTube operates on such matters is quite disconcerting. Both audio and video content are analyzed and compared to ensure there are no copyright violations and aggressively restricted if there are. Of course, we're forced to live by extremely strict interpretations of what a copyright violation is. A song playing in the background could trigger this, as could an image from a movie or television program. There are different levels of what can happen when violations are found, ranging from account suspension to being forced to run an ad for the company claiming ownership of the audio or video content. And in many cases, those rules differ from country to country. We've found that a number of our HOPE videos are restricted from being seen in Germany because of some legal issue involving rights to a bit of music heard before, during, or after a talk that weren't cleared in that country. It's an insane system that hinders so much creativity and dissemination of information. Does it really matter if you hear a snippet of a song in a video that clearly isn't focusing on the music? In the world of litigation, it apparently does, but we shouldn't be forced to accept those draconian rules in the course of our daily lives.

One of the best examples of the absurdity of this system came when we tried to share a video of a talk given at The Last HOPE. One of our speakers had been featured on The Colbert Report *and showed an excerpt during his talk. When we made this available via Channel2600 on YouTube, our account was suspended for violating Viacom's copyright. So, even though this brief clip was completely about the person giving the talk, and that person clearly wanted it to be seen, in this crazy copyright-crazed society we're building, they had absolutely no right to share this material. It gets better. Not only could we not share it in any way (audio or video), but the clip wasn't available on The Colbert Report's website or anywhere within Viacom. So it's not like they wanted to be the sole providers of the content; they didn't want* anyone *to provide the content, period. And legally, they can get away with that. But that doesn't mean it's right or makes any sense. Incidentally, all of this is automated; it's next to impossible to actually speak to a human about any of these actions. We've tried. What's frightening is that the technology is only going to get more advanced and "intelligent." There is great potential for far-reaching restrictions that we can't even imagine.*

Conversation

Dear *2600:*

In response to Barrett D. Brown (30:3), respectfully, you seem to have missed the point. If you're writing for compensation, go send your articles to some commercial magazine. There are plenty around. If you're writing to contribute to a community, send your articles to *2600*. At any rate, please stop complaining. Its rude, and kind of annoying.

2600, thanks for continuing to put out the best rag on the planet. Having read back issues from 1984 to 1990 and every issue since 1999, I can pretty confidently say that you have done a great job of filling the magazine with relevant, interesting articles since day one. (Nothing from 1991 to 1998 is going to make me a liar, is it?) Hopefully, one of these days, I'll write an article worth printing and when I do, you can keep the t-shirt. Seeing my words in your magazine will be payment enough.

Tyler

Thanks for the kind words, but let's be clear about who makes this magazine truly magical. Our readers who become writers and share their experiences, thoughts, and ideas with the rest of the community are the ones who make the framework. We provide the vessel and a bit of guidance. But what we do is merely a reflection of what's already out there. It's an honor to be able to wrap it into cohesive bits four times a year. Regarding the meager compensation we do offer, please accept and wear the shirt. The more people walking around with these things on, the more new people we can reach. You'll probably have some really interesting conversations, too, as a result. As for the content from 1991 to 1998, we believe it stands up, even the stuff from the gas-leak year.

Dear *2600:*

Feelings on "Black & White - The Growing Schism Between Hackers and the Law" (30:4) by Scott Arciszewski: I found this article particularly important, not in that I've been negatively impacted by the law (thankfully) for any type of hacking, but I really felt it was important to touch on the note of anonymity he stresses. In regards to the wonderful article by lg0p89 about my hacker maturation cycle, I'd like to say I'm in the "sapling" stage. Maybe his message might just be the obfuscation of mine.... We just need to plant more seeds.

Mr. Arciszewski states that being anonymous is the first priority. I realize, ironically, I'm planning on sending this from my personal email address... and I don't care. Now, I don't mean to say that and imply he's wrong - in fact, I completely agree. It amazes me to this day how difficult it is becoming to remain truly "anonymous" as well; I find myself in a new career position in which a lot of my co-employees would benefit from certain articles or perhaps some of my own "white hatting." My desire to share with them is immense, but my worry of where their mind goes the moment they

see the term "The Hacker Quarterly" across the magazine certainly comes to mind. Or, mainly, if I present these ideas, am I going to be thanked, or be without a job? How can I prevent the latter from happening?

Before reading the article, I had just purchased my *2600* shirt and calendar - the calendar I intended for work. Sadly, it may have to stay at home. But I want to point out, I think there are other methods we can use to get people at least thinking like "we" (hackers) do; I ultimately feel that a lot of our movement fails in the classification/labels that we hold so dearly. I simply wish there was some way of changing the public's perception of what a true "hacker" is, which I think is embodied in the *2600* community I've read and come to know since I was 12.

It's almost like *2600* 30:4 came at a critical moment for me, and it's great to be able to say that back in even 1987, you guys were already blowing the whistle on these agencies like the NSA, etc. Everything is so relevant to the fact that the government seems to really like making whatever is a threat to their power (knowledge being the largest) the perceived enemy or bad guy.

Mr. Arciszewski states (in regards to getting the feds/police to stop arresting us): "that won't happen." I agree, although I'll point out I still feel, through the venues like *2600* and by maybe more obfuscated methods, we can get our message out. For multiple years, I've had the tendency of leaving copies of *2600s* in restroom stalls, or I just happen to leave a copy at a few friends' houses.... I always make it a point to have the conversation about my interpretation of what "hacking" politically and sociologically means to me. The only reason I have any grasp on that is what I can thank *2600* for. But *2600* is more than a number, more than a magazine, more than the definition of "hackers;" it is a movement, and a positive one, which can collectively grow if we just work on eliminating that fear of the "H" word.

Thanks for reading, and thanks for everything.

Phedre

And thanks to you for this thoughtful letter. One way of working to correct the inaccuracies concerning hackers in the media is to call them out when they clearly get it wrong. How many times have we seen stories that report a massive security hole, yet the only threat is what might happen if "hackers" gain access to it? As if these were the only people who could ever do something malicious with an insecure system. We've seen an increasing number of media outlets use a more accurate term like "attackers" to describe those who, well, attack a system or security hole. To be clear, these could very well include hackers. But they can include all sorts of other people because there's not a whole lot of technical ability that's needed to exploit a lot of vulnerable systems. Just like you don't have to be a computer programmer to run a program, you don't have to be a hacker to mess with technology. What hackers will *do is figure out entirely new methods of both exploiting and protecting systems - and they will usually tell anyone interested in learning. The people labeled as such in the media almost always have no such interest or skill.*

Dear *2600:*

What exactly is the status of Tor?

It seems to be a back-and-forth yes/no between the media and Tor itself. I was surprised to see an article in the Winter 2013-14 issue of *2600* recommending the Tor Browser Bundle, considering all of the videos and news articles as far back as August of last year when the *Guardian* was releasing detailed articles on how the NSA "cracked" Tor. I would really like to hear *2600's* opinion on this in the next issue.

If the NSA manages to circumvent every attempt at anonymity, maybe it's time the people of the free (and/or not so free) world went head-to-head with them. Yes... they have ridiculous computing power, storage facilities, etc., but combined computing, like the SETI and Genome projects could rally the resources of pissed off people who are sick of their privacy rights being violated on an unprecedented scale. If the NSA was overwhelmed with anonymous/randomly generated key words, trigger phrases, etc., maybe it would render them ineffective, at least until (hopefully) their current methods are curtailed through legal avenues.

I think the number of participants would far exceed any of the well known combined computing projects in existence. When I asked my friend's grandmother if she would install such a thing on her computer, she replied "in a heartbeat!" because she is so pissed at the government.

Of course, for the average non-techno-savvy Joe, the deciding factor could be their anonymity in being involved in such a project, which would require many Tor-like services. This brings me back to my original question. (It's just something to ponder.)

~justanothersubscriber

We believe Tor is still one of the best means of anonymously using the net. But that doesn't mean it's secure for people who don't take certain precautions. Some would argue that using Tor Browser Bundle in Windows is a security risk in itself. We also see advice to not use Tor from your home or to use it for too long from the same place. If you're involved in something truly risky, these precautions are common sense. But for those who simply want to hold onto a bit of their privacy and aren't expecting to have their doors kicked in if it's violated, we find Tor is enough to at least slow the surveillance process down significantly. If it's only used by people who are on a government list of subversives, then it's a whole lot easier for them to be tracked down. However, if it's used by a significant percent-

age of the population, especially those who have "nothing to hide" but choose to protect their private info anyway, then the job of the trackers becomes incrementally more difficult and frustrating. So, in short, Tor is still one of the most useful tools out there but, as with most of these things, its true strength comes in numbers and in user awareness.

Clarification

Dear *2600:*

What I like about *2600* is I get to read about topics I know next to nothing about, such as what "Telecom Informer" brings us each quarter. Other times, I learn more about a topic I thought I already knew a lot about, such as Tor. But then I read articles such as part two of the Minuteman III Weapons Systems and feel compelled to respond.

The idea that VHF radios can only be operated on water is false. The VHF Maritime band is a tiny sliver of the VHF spectrum (30 to 300 MHz) and, while it's true those frequencies are only to be used on or near water, the FCC does allow them to be used for other purposes in areas without major bodies of water. Police departments, highway patrols, aircraft, ham radio operators, businesses, FM radio broadcasters are among the countless users of the VHF spectrum. What prohibits transmitting is the lack of a license. It has nothing to do with water. Additionally, it's illegal to intentionally interfere with the primary license holder.

A quick glance at radioreference.com shows many missile bases in the U.S. are using UHF trunked systems and they're all encrypted. Trust me, anything remotely related to our nuclear weapons has long been encrypted.

Given that, it's not a complete waste of time monitoring an encrypted channel. Sure, you won't understand anything being said, but you will know something is being communicated. If, under normal circumstances there's chatter, say, once an hour, and all of a sudden the chatter is nonstop, something is happening. Probably a drill, but it could be the start of World War III.

byeman

Dear *2600:*

In 30:4, Bad Bobby's Basement Bandits had an article about the Minuteman III weapons systems and the crews that operate them.

In this article, he (they?) mentioned VHF radios, and that civilian use of VHF radios was strictly for boating.

The VHF band is a very large space, ranging from 30 to 300 MHz. It includes, amongst other things, 12 channels used for TV broadcasting, the entire FM broadcast band (201 channels there), three ham radio bands, old (very old) cordless phones, aircraft (private, commercial, and military), railroads, various fire, police, and ambulance services, and random businesses including, as so nicely demonstrated in *Freedom Downtime,* the local McDonalds' drive-through window. There are even five channels set aside making up the Multi Use Radio Service (MURS) which can be used by any U.S. citizen for most any purpose - much like a CB, but with smaller antennas and fewer users.

VHF is hardly the sole domain of maritime and military users. The band is crowded, but it holds many different classes of user.

Glenn

Dear *2600:*

I just wanted to make a correction to the article on the Minuteman III system article in 30:4. The author states that the VHF radio bands are only to be used near large bodies of water. This is actually not entirely true. The marine band portion of the VHF radio system is this way, but the marine band is in no way the only VHF radio service out there. Two meter amateur radio is VHF, as is the Multi Use Radio Service (MURS), which is license-free if you meet the power requirements. Many local police, fire, and EMS agencies still use VHF systems, as well as businesses, individuals, and yes, the military as well. The key to remember is the *frequency* used. Most of the VHF military frequencies are in the 160-174 MHz range, as well as some in the 138-150 MHz range, and still more in the low VHF range of 29-50 MHz. Marine radio frequencies are in the 156-160 MHz range. The only military traffic you will hear on marine radio is likely the Coast Guard. And yes, it is illegal to use a *marine* radio in an area not near a large body of water. It is not illegal to use VHF in general inland, as long as it's a frequency you are authorized to use.

William

This certainly generated a good amount of responses correcting the initial statement. Thanks to all of you for the clarification.

Dear *2600:*

In 30:4 of *2600*, the column "Transmissions" by Dragorn mentions building a device to indicate when an E-Z Pass is triggered by a reader. In the January issue of *Popular Science,* on page 72, there is a mention of how to do this and a link to the circuit at popsci.com/ezhack. These instructions were written by Puking Monkey, the same person mentioned in Dragorn's article.

Bandersnatch

Donations

Dear *2600:*

I'm not sure if I'm the only one vibing on the community here, but what if you asked for volunteers willing to OCR and correct old issues?

I'd be happy to do a few. I don't think I can commit to a whole volume. But if you need a few issues digitized, let me know. I already have copies of most of them (inherited from old friends and sought out) and wouldn't expect anything in return, except maybe gratitude. Those old issues are a treasure trove of interesting information, the annals of

hacking, if you will. And I'd be happy just knowing I did my part to allow the young ones of today to experience the magic.

T

We do appreciate such offers. Our project is to present these issues in a number of formats, both text and graphically based. That means it's not simply a matter of scanning, and the limitations of OCR software coupled with our frequent use of microscopic text makes this a very time consuming project. But it's one we care about and one we want to really get right. As of now, the amount of people buying the older digests doesn't justify the amount of work we're putting into them. We understand it may seem strange to pay up to ten bucks for a year of material that's more than one or two decades old. But that investment helps us make the archiving project possible. We fully intend to get it done one way or another. The only real question is how long it'll take.

Dear *2600:*

I am the proud owner of the account @2600 at app.net. Since I never used it after I created the account (I just followed a few people, but did not even read the timeline), I decided that I should give this away to you. If you are interested, please let me know. I ask for nothing in return, but if you want to do something, I suggest you donate a real good sleeping back to a random homeless, or something like this?

Deal?

Best regards and thanks for all the good work.

Dennis

Thanks for the offer - it sounds like a fair deal. (We assume you mean "sleeping bag" as we have no idea what donating a "sleeping back" would entail.)

Contributions

Dear *2600:*

I am interested in purchasing some annual digests in PDF format. May I suggest a subscription type product?

For a $260 single payment (same as for my lifetime print subscription), the buyer gets PDFs of all the annual digests currently available and, for life, each additional digest (both filling in old ones and adding new each year) as they become available. By purchasing this "subscription," the reader makes a significant financial contribution to the project of digitizing all past issues.

What do you think?

sol

That's a damn good idea and one we're going to seriously consider. But it would have to apply only to the PDF version as we have no access to customer information for the Kindle version. We're curious what others think of this creative solution.

Dear *2600:*

I have been a faithful reader for the past ten years. Keep up the inspiring and innovative work! Also, for the past ten years, I have been a malware analyst and an inventor. For the past two years, I have been working on my XE-2600b malware interceptor, which will hopefully allow me to capture malicious code trying to attack my test network for studying and reverse engineering. I was wondering if there were any such projects already in development. Keep up the good work.

flames

As there is no shortage of malicious code trying to attack networks, there is an abundance of creative types looking for ways to counter that and provide a valuable tool to the community. You can read about their exploits (pun intended) here or find others who would be interested in this sort of thing at hacker conferences and 2600 *meetings. We look forward to following the progress.*

Investigations

Dear *2600:*

I have no one else to turn to. Authorities won't help me and I have tried my best to find this person. He has been harassing my sisters and now one of my friends. And now he's threatening to expose her on every social page. I want his address and his name. If you are interested in helping me, I will give you his phone number and the email address.

Jeremy

First off, if you have someone's phone number and email address, that's enough information in most cases to track them down if they are truly posing a threat. Have you asked yourself what you would actually do if you knew exactly who and where this person was? And, having answered that, is it really a good idea? It's easy to get wrapped up in this kind of crap and make it a whole lot bigger than it really is. You can block phone numbers and filter out email addresses. Most of the time, it's the reaction that fuels a harasser. Take that away and they tend to lose whatever power they're holding onto.

Dear *2600:*

I recently had a firsthand experience with social engineering. My significant other began playing a popular word game with an ex. At the time, we shared a mini-tablet and it seemed innocuous enough at first, but some messages came through that proved otherwise. In a small fit of anger, I put a keystroke logger on a semi-shared laptop. I got three of the four relevant passwords that I needed to more closely monitor the situation. The fourth proved to be quite stubborn since it was for an account that wasn't accessed on the laptop, but on my SO's phone. Here's where the engineering came in. I have a PoS phone that I have complained about regularly. I used this to my advantage. I requested to be sent a photo that was only on the laptop and

further required that it be sent from said fourth account. This was under the guise that "I can't save photos to this PoS phone that are sent from any other account." Voila! (I will add that the logger was eventually discovered due to user error, but by then it had well served its purpose.)

P.S. Do you send notification if a letter will be published, or if it will just be ignored?

pathos.ethos

We look forward to seeing this story play out on an afternoon talk show, hopefully with flying chairs and phones. As for notification of letter publication, you're looking at it. Hopefully.

Dear *2600:*

Recently, while working on a client's computer, I was asked to install a Wi-Fi adapter dongle that has no markings as to make or model. The device itself would not install device drivers onto the computer. Having left my Ubuntu Live USB drive at home, which normally is able to tell me deeper information of hardware on a device, and only being left with my R&D laptop (Toshiba Portege M405) with a base install of Windows Vista, I had to resort to other methods. The dongle has Wi-Fi N on the top and, past that, any user trying to determine what this was to get a driver for it would have had to do an image search and hope they found what they were looking for. In the process of looking through the scant documentation, I noticed that the chipset is a Ralink RT5370 which, after a quick Google search, brought me to www.ralinktech.com which has recently merged with MediaTek. Both of these companies I have personally never heard of, but I was able to click through and find proper drivers for it and was able to finish my task. Thought this would be an interesting little tidbit for anyone going through the same issues that I just went through.

Love the magazine - glad to have such a great source of technical information at my fingertips that is created by the readers.

--handle-need-not-apply--

Dear *2600:*

I don't know if you all get these kinds of requests or not, but I'm a student at one of the local community colleges here in Denver. I've been reading your mag for quite some time. I know you all ask to subscribe to the mag and have a subscription if you want to even think about posting, but in truth I was a little skeptical about who all had access to that information. Now, years later, I've kind of cleaned up my act and am trying to move over to the other side of the hats. Hence, the schooling. Now for the main reason for this rant... I'm doing a report mainly based on privacy and, hence, am including info on SOPA and PIPA. The thing is, most of my works cited are conglomerate BS, if you know what I mean. I was wondering if maybe anybody there at headquarters might be willing to help me out with any info regarding privacy and how it affects society today that you might have in that vast library of yours. If so, I would be more than grateful.

Joseph

The Electronic Frontier Foundation has a real treasure trove of material online that should help you get a sense of the history and the significance. You can also find quite a bit from the American Civil Liberties Union and the Electronic Privacy Information Center. Through all of these, you'll undoubtedly find more.

Please don't worry about who may find out what you're reading, at least not to the extent that it changes what you read. The more people who refuse to take this seriously, the less serious it can be.

Suggestions

Dear *2600:*

I want to ask you if you'd be interested in publishing an article about our latest discovery: how to scam *2600 Magazine* and gain free subscriptions, magazines, t-shirts, email bounce backs, etc. This should work worldwide. By the way, I belong to an intergalactic white hat, elite hacking, super illuminated, certified, white hat hacking federation called White Jacket Hacking Group Worldwide. LOL

Bob Hardey's Mom

Let's see if we can guess. You send us an article which details how to get free stuff from us by writing articles to get free stuff and then we send you some free stuff in exchange for the article. If you can put something together that goes on for more than a sentence or two, it might be worth it.

Dear *2600:*

In a recent tidy-up, I found some old 28.8k dial-up modems. I remember experimenting with them years ago, and I discovered that if you connected two modems to the same phone line (by plugging them into a double-adapter and plugging that into the wall socket), they could be told to connect without making a phone call - very exciting at a time when this was your only means of connecting two computers together!

However, if you just connected the modems together (into the same double-adapter but not plugging it into the wall socket), it wouldn't work. The modems couldn't hear each other without a live phone line being involved, even though the line was not used to dial out.

I have always wondered why this was the case. Do modems require line voltage to be present before they can communicate? If one was feeling nostalgic and wanted to experiment again, could a phone line be "faked" by just sticking -48VDC onto the cable connecting the two modems?

Malvineous

You are absolutely correct, line voltage must be present. In fact, you've stumbled upon an old, inexpensive method for connecting two computers together for simple point-to-point networking or file transfer. For this reason, many companies

sold "phone line simulators." Not only was their primary purpose for testing telephone equipment, but they were also very useful for connecting two computers together via modems within the same building over much greater distances than a simple null modem serial cable would allow, given the higher voltage and current of the (simulated) phone line. A Google search will reveal commercial phone line simulators for a wide price range, in addition to simple, no-frills, do-it-yourself versions for as cheap as ten bucks in parts.

Dear *2600:*

Freespeechme.org deserves a serious look. It's based off of Namecoin, and the idea behind it has been out for a while now. I believe Aaron Swartz was eyeing it at one time. In the end, it's a really cheap way to register a domain (Dot-Bit for mere pennies) that has jack squat to do with ICANN (totally different, almost "bulletproof" infrastructure). See how you guys size it up.

Chris

This is the kind of thing we like to see. We want to know if our readers have been making use of this and, if so, what their experience has been.

Observations

Dear *2600:*

One personal realization I've come to during this NSA debacle is that security is like gaming: it stops being fun when someone cheats. While the tech giants are surely scrambling to capture their customers' trust, and more importantly their shareholders' appeasement, I hope the subversion of security - through methods which deserve no merit - doesn't extend this disturbance to those among us who contain the true hacker spirit: the mindset and capability of overcoming the odds using ingenuity rather than unlimited resources and show-of-force. To them I say: don't give up! And to those other guys I say: cheaters never win.

Potissimum Libertas.

Justin

Dear *2600:*

This does not warrant a full article, but I just wanted to point out to your readers through you, that if they love their privacy, an old-school technology can help them.

Being a privacy lover myself, I grew concerned to learn, through the Snowden revelations, of the extent of surveillance on cell phone users. I remembered that pagers don't have transmitters, and discovered that there are still two nationwide paging companies (USA Mobility and American Messaging). Deals can be had through a few online resellers if you are willing to pay for several months upfront.

New to this old technology on the back-end, you can have copies of pages emailed. This is great if you want to create redundancy to a cell phone for spotty reception situations. That's an option they charge you for, but you can have a free recording of yourself when the pager company answers their number.

I've also found that I can eliminate voice mail, which I find quite inconvenient, by forwarding my wire-line phone to the pager company. This also eliminates robo-calls, campaign calls, etc. Auto-dialers are baffled by the pager company, which is great, IMHO.

I hope some in our community will welcome this old-school, but private, technology.

DeepGeek

Dear *2600:*

I came across this while reading *Love and Math: The Heart of Hidden Reality* by Edward Frenkel. This may be the quintessential essence of hacking! In reference to Galois' approach to solving polynomial equations: Galois did not solve the problem of finding a formula for solutions of polynomial equations in the sense in which it was understood. He hacked the problem! (circa 1820) He reformulated it, bent and warped it, looked at it in a totally different light. And his brilliant insight has forever changed the way people think about numbers and equations. You'll need to read the book to learn more about the Langlands program, a transformational unified theory of mathematics.

William

Dear *2600:*

This is in regards to Steam's (the largest computer game marketplace out there) Valve Anti Cheat now mining your DNS cache history to see which domains you've pulled files from (whether that be an image loading or a page load).

I don't condone cheating in online games. In my personal opinion, based on my tens of thousands of hours of gaming online, I'd have to say that the majority are out to make up a lack in their life by acting on sociopathic impulses (trolling and griefing).

That said, Privacy should always be written with a capital P.

In the near future, someone will write an app to automatically clear the DNS cache on a computer, and evolutions of that will hopefully be truly "protected storage" in the form of locking it down and making it unreadable, spoofing the data so that only what the user wants to be seen is shown to any third-party application reading it, or hell, knowing about hardware boot-kits, software root-kits, and NSA's PRISM, the operating system too!

Maybe one of you readers will be that someone.

Distributed DNS, Undernets, AlterNets, and the like aren't a reality yet. They're not "vaporware," but they're not "good software" yet either. So in the meantime, some security specialists need to get cracking on some of the concepts I outlined in "Anonymity and You, Firefox 17 Edition" (30:4) and this. Preventing insecure local data storage that can currently be abused from staying open

to such attacks is a priority. Don't trust the hardware, don't trust the operating systems, and sure as hell don't trust software, even if it's something *you* or someone you trust wrote.

There are plenty of factors in play now that we've seen. Examples of this are rootkits in Linux distributions put there by intelligence agencies, backdoors in hardware and operating systems put there by manufacturers or "Men In The Middle", as well as *huge* third party software vendors like Valve.

How would you like the United Kingdom's "Ministry of Truth" reading your DNS cache every time you run a BBC news applet? Flagging a user to be banned from their ISP for using a VPN to read blocked content or things not available in their country is not just possible, it's likely. This applies everywhere, though the U.K.'s recent "efforts" to block more than just pornography and copyrighted content are visible in the media at the moment, so it makes a great example.

Get to net work, folks!

L0cke

Dear *2600:*

The craziest thing happened today at Target. Wife and I went to see what a friend's gift card issue was since they couldn't use it at the restaurant (how embarrassing). Anyway, we went to the return center, had a little chit chat, and they would've given us a replacement IHOP card but, unfortunately, they were out. So we got a different card of equal value and went back to the return center. The lady over there had a return ticket already prepared for us to do an exchange. She held up the ticket to scan it, *beep,* then all of a sudden the register crashed and forced a reboot. She was like "uh oh, the register crashed, let's try a different one." She went to register #2, *beep,* same thing. Once again frustrated, she tried register #3 in the same returns area. *Beep,* same thing. The returns area was now out of registers, so basically my wife and I shut down the returns area without even lifting a finger. We eventually had to go to the checkout area. So on register #4, the lady entered her worker ID, the password, then *beep!* You guessed it! *It crashed!* I was laughing up a storm deep down inside thinking that Target actually generated a return ticket that made their point-of-sale systems crash. It would have been hilarious if I had gotten a hold of that ticket and published it in the magazine. In reality, I was a bit aggravated that it took so long to exchange a gift card. I just thought it was worth mentioning that a simple ticket being scanned caused reboot chaos across four registers.

CasperGemini

As if Target hasn't had enough problems lately, this is something they really ought to lose sleep over. We'd like to hear some theories as to what may have been going on here. Now that we know such a thing is possible, we're sure all kinds of experimentation will ensue, not only at this retailer but at many others. Bad software allows for so many possibilities.

Dear *2600:*

I understand that "three letter government agencies" by law cannot collect the facial recognition information, but can "buy" it from Walmart and other entities. Walmart tries to match "faces" with credit card information, which then will give them names and addresses. Even if you usually pay cash, if you paid by credit card or check even once, they gotcha! It also appears that Walmart and Target collect information from RFID tags placed in high end clothing like expensive jeans (under labels) and other clothing customers are likely to wear again while shopping at the store. Walmart calls their program EPC. There is a sign on the door saying you can look up EPC at walmart.com if you want to know more. I did find a funny looking RFID tag in some underwear I bought at the Walmart in Franklin, Tennessee. It appeared to be over an inch wide and half an inch high. Had a chip in the middle with big wings attached on either side. I asked a relative that works in IT security what I should do with it. He said to find a cart in the parking lot and tape it to a not so prominent part and they'll be tracking that cart forever. I would send it to you, but it got lost in the car trunk.

Boxholder

Versification

Dear *2600:*

Of copper, light, and waveform spawned,
The Argus' gaze pierces from beyond.
All man's deeds simultaneously recorded,
Myriad strands of data, all hoarded,
To this multi-eyed and mindless being,
Was given the gift of being all-seeing.
A mass of sensors, ubiquitously extended,
Regardless of source, all feeds comprehended.
Bentham's design, reaching greater height,
Achieved not by brick but by patterns of light.
In omnipresence, there can never be break,
For when one eye lies sleeping, another's awake.

With such density of bits flowing through the wire,
Increasingly murky is the Boolean mire.
Yet there remains hidden, despite highly sought,
No datagram yet can encapsulate thought.

Evan Krell

Dear *2600:*

My computer is not a tool. It is a person, just as I am. If I treat it like a person, it will treat me like a person. My enemy is strong, but I am stronger. My enemy brags about his ten gigaflop computer, but I am more powerful with my 30.68 gigaflops of fury.

I must know my computer, inside and out. I must know its hardware, its software, its networks, and its capabilities. If I am one with my computer, my computer will be one with me. I must destroy my enemy, and he will be nothing but a pile of bullshit and shitty computer parts. I swear by this creed and my country should stop all of this "hackers are criminals" stuff.

Neo Anderson

Convocations

Dear *2600:*

How does one go about getting added to the meeting list? I was told by the meeting organizer that he had submitted to be added, but it has been months, and we're still not on the list. Is there a submission process? Any information you could provide would be awesome.

There are no other meetings listed even remotely nearby this area so it would be great to be added. Thank you.

Johnson

It can take months because we're a quarterly publication and meetings are updated for each issue. Plus, just submitting a meeting location is only the first step towards getting listed. There has to be follow-up as well, letting us know if the meeting took place, how many people showed up, if everyone was bailed out, etc. We're happy to say that your meeting is now on the list.

Dear *2600:*

I saw that you have taken out the Orlando meetings at the Fashion Square Mall at the Panera Bread. Could you please tell me why?

youssef

Due to numerous complaints about nobody showing up and a dialogue right here in the letters section, we felt it was best to remove it until and unless it becomes more organized.

Dear *2600:*

I'm a technology enthusiast down here in Costa Rica and wanted to start a *2600* meeting. What are the requirements for me to get listed down here?

B

We're happy to say that you've already met them. By posting the proposed meeting location, holding a couple of meetings, and letting us know how they went, you've given us enough reason to believe that this is going to be taken seriously and our sending people in your direction won't be a waste of time for them. We wish you the best of luck and hope to hear more.

Dear *2600:*

I've tried to make contact with the organizer for the Ann Arbor meeting and haven't had any luck. I've tried to catch them in IRC, etc. as well to no avail. Are you aware of the current status of the meeting? If it's dead, which it really seems as though it is, I'm located in Detroit and am considering starting up a group here since one doesn't even exist for Detroit. Please let me know what you know!

Matt

Regardless of whether or not Ann Arbor is still happening, a meeting in Detroit is something we'd support. We'll look for more reports on the status of the Ann Arbor meetings and act accordingly.

Dear *2600:*

Friday between 5 and 8 pm is difficult for both religious Jews and Muslims, as Friday until sunset is a holy day for Muslims, and Friday evening to Saturday evening is the Jewish Sabbath. Is there any room for scheduling a meeting Saturday night instead, a common time off for everyone? (Sunday is a regular work day here.) I know it's a long shot, but because of the surrounding circumstances as far as the work week and religious issues, I had to ask.

S

This was discussed a bit in our last issue and, as it turns out, a group has put together a meeting in Israel that doesn't conflict with the Sabbath. So, for the first time, we have 2600 *meetings that don't take place on a Friday evening. Since such a sizable amount of the population wasn't able to participate on that day, this exception makes sense in this situation. Below are some details on how it all went.*

Dear *2600:*

Here is a summary of the first *2600* meeting in Israel:

5:40 pm: guy walks by "2600 mah ze" and says to his companion ("what's 2600?"), responding to the fact that I perched the magazine at the end of our table in the food court. The guy keeps walking.

5:51 pm: balloon popped, echoing around the food court and scaring everyone. Whew, not a bomb. No, it wasn't an attempt to drum up some attention for our meeting.

6:25 pm: The same guy approaches and again asks me in Hebrew (not his companion this time) "what is 2600??" Told him briefly, showed him the magazine, he thought it was cool even though he couldn't read English, shook my hand.

7:05 pm: second meeting attendee shows up.

We discussed IPv4 compared to IPv6, related security issues, how to promote the meeting, and the best flavor of ice cream milkshake at McDonalds. (Having a kosher McDonalds around is always a treat.) The meeting was held in English.

The second person attended through word-of-mouth and not because of my posting online.

Incidentally, I posted different versions of the following:

"Putting together a *2600* meeting, getting it off the ground according to the suggestion in the latest issue of *2600* to hold it Thursday instead of Friday (Shabbos). Therefore, it will be on Thursday, February 6, 2014 from 5-8 pm in the *big* Fashion Mall in Beit Shemesh, second floor, food court. (Mall is

across from the Beit Shemesh train station.) Please print and hang on bulletin boards, repost online, and spread the word."

S

Congrats and please keep us updated.

Dear *2600:*

This recent Friday was the first meeting I attended, and I brought a friend with me. Unfortunately for us (and I don't want to call anybody out), we went to the meeting location listed at dc2600.com. This week, the meeting organizer(s) decided to try a new location which they announced on Twitter (which I saw after arriving and two other new people showed up as well) but didn't update the website. The new location was several metro stops away and by the time we would have gotten there, the meeting would likely have been over already. So those of us at the old location had a nice dinner and chat, and I jokingly christened us the 5200. We all now know the meetings are in a new location and will be there next time, but that's my report from the underground.

Matt

We're sorry this happened, but to the best of our knowledge the meeting location in Washington DC hasn't changed. Twitter really isn't the best way to announce a change, especially if there's an existing website. Hopefully, this was an anomaly, but if there is a change to the location, we'll be sure to publicize it.

Appreciation

Dear *2600:*

I finally did it. After living the minimum wage lifestyle for so long, I managed to find actual, meaningful employment in the IT field. A week ago, I was stacking cans of soup in the world's sleaziest health food store. Today, I was restructuring the VoIP system of a corporation with offices around the country. Once I get my first paycheck, I'll be making exactly twice as much per month as I was at my last job. At the very least, after three months I'll be able to say I'm an IT guy and never return to retail. I sincerely want to thank the editors, contributors, and readers of *2600* for making a worthwhile magazine which kept me sharp and ambitious while I wasted my life doing pointless work. To anybody out there who can relate, and who knows what it's like to be stuck in the meaningless cycle of unskilled labor, I want to say that there is a light at the end of the tunnel. Even if you lack degrees, certifications, or ten years of "real" experience, there's an opening out there somewhere where you can break into the field. Use the same hacker skills you've used your whole life to outsmart the other nine candidates who've applied for the job, because you're absolutely capable of doing so. Be ambitious and confident, and just go do it.

Anonymous

We appreciate the words, but the credit goes to you for not losing sight of the potential that's always out there. While there are many "meaningless" jobs, what's unique in all of us is our imagination, something that all of the oppression, boredom, and discouragement of the world isn't able to crush. That uniquely human characteristic is the shining light that brightens the daily drudgery and which can often lead us out of it. The point is never to give up on yourself or on the potential for change.

Dear *2600:*

I have read your magazine for quite some time and I love what you do. After my computer teacher introduced me to your magazine, I have wanted to learn hacking because it sounds like a fascinating field. The trouble is, I have searched far and wide for directions on how to begin learning about hacking. While the information in your magazine is intriguing, I admit that most of it is incomprehensible to me due to my lack of experience. Can you tell me where and how to begin?

A loyal reader

You have already begun. The thing to remember is that there's always going to be material that appears to be incomprehensible to you. This is true of everyone, whether they care to admit it or not. The more you read, obviously, the more familiar you'll become with the subject matter. But even in those articles that you believe are shooting well over your head, we believe you can grasp the overall meaning of them, even if the particulars escape you. Otherwise, why would you be even slightly interested? So, since we've established that you have in fact already started to learn quite a bit about hacking, the best way to keep or increase your momentum is to become more a part of the community, whether it be by going to meetings, becoming part of a local hackerspace, or engaging in a dialogue here. You will never know it all. But you're in as good a position as anyone to get a firm appreciation and overall understanding of what the hacker world is all about. That alone puts you ahead of just about every elected official and media pundit out there.

Dear *2600:*

A couple of comments. First, while none of us either like DRM or the MPAA/RIAA Mafia or much of their illegal actions, we cannot change them without letting our money and doing business with them speak loudly. Next, having been a victim of ID theft twice, I do not do online transactions. Since Amazon will not do business with you unless you provide both personal information and do it all online where you are again subject to exposure, I simply will not do business with them.

That is why I have a subscription to *2600*. I enjoy it immensely. I laugh more often than not as I read the letters, sometimes from the content and frequently from *2600's* reply. I love the sardoni-

cism and often excellent sarcasm. I do not always agree with you, though, and I think you are making a mistake with Amazon as Amazon is well on the way to pushing out many small bookstores, as I learned while searching for two books recently. I paid almost triple to avoid Amazon and I don't regret it. I will continue to keep up my subscription to *2600* and pray that they are not forced to deal only with Amazon as time goes on and as a number of small bookstores have been forced to. They won't take checks and if you won't pay online, they "can't" do business with you, and that includes credit cards over the phone. That's what the market e.g. Amazon requires.

I particularly enjoy the fact that *2600* encourages the younger folk in more positive ways than my generation did. I don't get all the techie stuff but I get enough that my network hasn't been intruded upon since 2006, and I haven't been tagged with malware in nearly as long.

Captain V. Cautious

We have supported local bookstores from our beginnings and we're always happy to be carried in one. As a magazine, however, we want to make our publication available in as many places as we can, in print and through digital methods. Expanding into chain stores years ago helped us gain many new readers. More recently, through the Kindle, we've managed to reach many thousands of people who we may otherwise never have reached, as well as reestablished links with readers who, for one reason or another, were no longer able to find us in their local stores. We're now reaching even more people through Google Play. Our goal is to have as many options available to readers as we can, so that if one doesn't work for you for whatever reason, there will always be another.

Dear *2600:*

This comes to you from DownUnder and via snail mail - guess you could say I am of the old school and sit on the fence between old and new with a foot in each.

Having said that, I have to say how delighted I am to find such a tome as yours - sort of unlocks the Pandora's Box of computers for me. Your zine came by way of my new Kindle - an unexpected birthday present which opened a new window in my "reading soul" and showed me the way to San Jose and back with a cache full of books I never expected to access. Glorious fun for a bookworm such as I!

So, worming my way through the many categories and subcategories, I spied the Internet and Technology section... whereupon I thought "well, this could be interesting...."

Oh, the delight of my left brain! Here it is, the way through the maze and labyrinth of "how it bloody well works." Hats off to all ye hacker folk who delve deeply where angels fear to tread.

Seriously, my knowledge of the deep stuff is limited, but *2600* has given me a new lease on life to go where my angels said "no, not there, 'tis the devil's playground." Devil be damned, so Volumes 28 and 29 were added to the cache and thus it is I begin my new lessons.

My original lessons began on a Mergenthaler Linotype as I decoded the art of good old fashioned typesetting in the days when you had to learn what all the fonts looked like or were going to look like in a given text, how much kerning and leading were required, not to mention the art of drawing a line. That required an x and y coordinate plus the point size, and hopefully you did not end up with an elephant's footprint rambling across the page to infinity. All done, of course, looking through the glass darkly on a screen which simply blinked in black and green. Keying in the daily horse racing guide required the use of left hand mouse action, right hand typing, and a memory which contained the endless parameters for font changes, lines, dots, white space, and alignments! Would I thus be correct in surmising that the HTML used today is a child of the original typesetting codes and parameters? Has to be, methinks.

Well, I am still finding my way to get "more private" in my computer land and the many hints I've found in *2600* are inspiring to say the least. For some reason, I always felt the need to use totally different passwords - just seemed to make sense and I simply keep a hard copy. A clever little "grimoire" in an alpha-lingo of my own creatings....

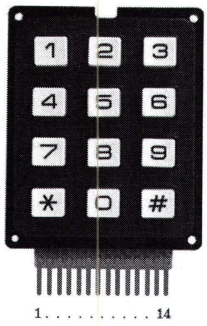
So, as left brain would have it, I figured I needed to educate myself on a bit of terminology and, to that end, discovered techterms.com, a great resource of just about any bit of lingo housed in computer land. Guess you have to start somewhere and, as much as I loved the articles, I was stymied because I didn't have a clue what most of it meant.

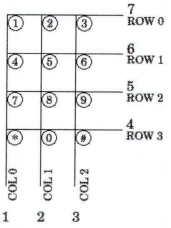
Of a couple of articles which grabbed my attention, one was on bitcoins (I had been looking for an online business and found reference to this, but left it alone). Then came along *2600*, and said article inspired me once again. Thanks for that! I kinda feel this bitcoin thing is important in the coming time. Most folk are aware that worldwide finance is a bit of a mess and the crunch will come. Things in Australia are not too bad, but poor old New Zealand is like the testing ground for what will come here. I sense that bitcoins will put money power back into the hands of ordinary folk. Maybe it will morph in the years to come, but it certainly rings a bell in my mind as being something to watch - and hopefully get into!

Well, that's it from DownUnder... go all ye hackers, go! Never stop inquiring and light the wise fire of divine intelligence which we all possess. I go now to disciplined study and uncover the hacker within....

Ed
Australia

Brute-forcing PIN Code Keypads Using Combinatorial Mathematics





by Alva Ray

Where I live - and probably in many parts of the world - most residential houses are guarded at entrance by the simple mechanism that is the four-digit PIN code. By pressing buttons on a numeric keypad in the correct order, the door will unlock, and all residents share that single code. Many of these numeric keypads have the same couple of flaws that make them more vulnerable to brute-force attacks: First, there is no confirmation button that needs to be pressed after having entered four digits. Second, the last four entered digits will always be accepted, instead of the pad resetting after an incorrect PIN code.

Now, brute-forcing a keypad of this kind only involves a maximum of 10,000 codes to begin with. While this may seem a large number, it's actually quite small compared to the possible number of combinations when brute-forcing, for example, a computer password. (A four-letter password using lowercase a-z means 456,976 combinations.) The big difference between brute-forcing a computer password and trying PIN codes on a physical keypad is, of course, that the latter can't easily be automated, meaning it will be very slow.

To go through all possible PIN codes, you could start at 0000, 0001, 0002, etc., and try them all in order. You would be looking at a maximum of 40,000 key presses, hoping for the correct PIN code to be early in the sequence. Being a skilled keypad operator able to try one PIN code per second, this method would still mean up to three hours of hard work and sore fingers.

But because of the flaws mentioned in the beginning, you don't have to press that many buttons. After having tried the first four PIN codes (0000, 0001, 0002, 0003) you have actually already tried ten different ones, since the pressed sequence also contained 0010, 0100, 1000, 0020, 0200, and 2000. By this principle, the number of required key presses is only a quarter of that initial 40,000. If you can keep up the same speed as previously, this means "only" about 40 minutes of work. However, the process in this case will probably be slower since the pressed sequence will not just be an ordered set of increasing numbers - something that otherwise favors physical brute-forcing since it can be carried out in a more systematic and thus faster fashion.

So, what shortened sequence might that be? In other words, what is the shortest possible sequence of digits containing all of the four-digit PIN codes from 0000 to 9999? Luckily, combinatorial mathematics can answer that for us, in the form of so called "De Bruijn sequences." Named after the Dutch mathematician Nicolaas Govert de Bruijn, attributing it to Camille Flye Sainte-Marie, Tanja van Aardenne-Ehrenfest, and himself, such sequence is according to Wikipedia:

"[A] cyclic sequence of a given alphabet A with size k for which every possible subsequence of length n in A appears as a sequence of consecutive characters exactly once."

In the case of keypad PIN codes, the alphabet has a length of ten (the digits 0-9) and the subsequence a length of four. Every De Bruijn sequence has a length of k^n, so this one will be 10,000 digits, plus an extra three zeroes at the end to cover all PIN codes, since the sequence is cyclic. Concluding this short mathematical excursion, all four-digit PIN codes can be expressed through a 10,003 digit number.

It turns out this string of numbers fits on approximately two A4 pages, meaning it could be printed double-sided on a single sheet, small enough to always be carried around in your toolbox/bag/wallet/pocket/hidden compartment. Any savants out there might find it useful to just memorize the whole thing. While still implying anywhere between one and several hours of number punching, this sequence will ensure the absolute minimum number of key presses.

Some possible scenarios: Finding yourself locked in, guessing a PIN code your only

escape, this will definitely save you valuable time and oxygen. Forgetting or losing the PIN code to your rented storage space or garage, it will save you the money for having the code reset by an operator. You could even save some stamp money by delivering all your mail yourself! OK, that last one was a joke, but you get the point.

Speaking of mail, the chances of hitting a correct PIN code early on in the sequence at any given residential house entrance are in fact higher than one in 10,000. At least over here, keypads accept additional PIN codes used exclusively by letter-carriers, codes that are often shared throughout entire neighborhoods. By going through the entire sequence on a less prominent keypad in your area, maybe in batches to avoid suspicion, you might find multiple working PIN codes. In that case, one of them is likely a service-type one - a skeleton key among PIN codes. Nota bene, you should not do this for any space you are not allowed access to in the first place, but that goes without saying.

I want to end this article with an idea for an invention:

It was said earlier that trying PIN codes on a physical keypad is not easily automated. However, it would be interesting to do just that, by building a small device with a set of mechanical "thumbs" that can be held against the keypad. It would then run through the optimal 10,003 digit PIN code sequence, pushing the buttons much faster than any human could. If the device could try even just ten PIN codes per second, it would take at most 16 to 17 minutes to guess the right one. If lucky, and if there are multiple correct codes, it would take a much shorter time than that. The device could be run by an Arduino board or similar, having some software on it that could calculate De Bruijn sequences itself given PIN code length, and remembering its position in the sequence when deactivated. If written so, and if activation of the device happens simply by pushing it against the keypad and deactivation occurs by releasing it, you would have a very stealthy piece of brute-force machinery. You could visit a keypad for just a minute at a time over the course of several hours or even days, always continuing where you left off. Bonus points for coming up with some clever way to make the thumbs flexible enough to be fitted on any keypad layout (4-3, 5-2, etc.). The advanced hardware hacker could even add a sensor to the device that can notice a green light, the common keypad mechanism for signaling that the correct pincode was entered. With a built-in GPS and wireless, the device could save its location and the correct PIN code and, when connected to the Internet, report this data to a shared database.

Without further ado, and using some Python code found on Wikipedia, I've generated for you the 10,003 digits making up the shortest possible sequence containing all PIN codes between 0000 and 9999 exactly once. Cut it out and save it, because you never know when it might come in handy:

```
0000100020003000400050006000700080
0090011001200130014001500160017001
8001900210022002300240025002600270
0280029003100320033003400350036003
7003800390041004200430044004500460
0470048004900510052005300540055005
6005700580059006100620063006400650
0660067006800690071007200730074007
5007600770078007900810082008300840
0850086008700880089009100920093009
4009500960097009800990101020103010
4010501060107010801090111011201130
1140115011601170118011901210122012
3012401250126012701280129013101320
1330134013501360137013801390141014
2014301440145014601470148014901510
1520153015401550156015701580159016
1016201630164016501660167016801690
1710172017301740175017601770178017
9018101820183018401850186018701880
1890191019201930194019501960197019
8019902020302040205020602070208020
9021102120213021402150216021702180
2190221022202230224022502260227022
8022902310232023302340235023602370
2380239024102420243024402450246024
7024802490251025202530254025502560
2570258025902610262026302640265026
6026702680269027102720273027402750
2760277027802790281028202830284028
5028602870288028902910292029302940
2950296029702980299030304030503060
3070308030903110312031303140315031
6031703180319032103220323032403250
3260327032803290331033203330334033
5033603370338033903410342034303440
3450346034703480349035103520353035
4035503560357035803590361036203630
3640365036603670368036903710372037
3037403750376037703780379038103820
3830384038503860387038803890391039
2039303940395039603970398039904040
5040604070408040904110412041304140
4150416041704180419042104220423042
4042504260427042804290431043204330
4340435043604370438043904410442044
3044404450446044704480449045104520
4530454045504560457045804590461046
```

```
204630464046504660467046804690471O   122612271228122912321233123412351 2
472047304740475047604770478047904 8   361237123812391242124312441245124 6
104820483048404850486048704880489O   124712481249125212531254125512561 2
491049204930494049504960497049804 9   571258125912621263126412651266126 7
905050605070508050905110512051305 1   126812691272127312741275127612771 2
405150516051705180519052105220523O   781279128212831284128512861287128 8
524052505260527052805290531053205 3   128912921293129412951296129712981 2
305340535053605370538053905410542O   991313141315131613171318131913221 3
543054405450546054705480549055105 5   231324132513261327132813291332133 3
205530554055505560557055805590561O   133413351336133713381339134213431 3
562056305640565056605670568056905 7   441345134613471348134913521353135 4
105720573057405750576057705780579O   135513561357135813591362136313641 3
581058205830584058505860587058805 8   651366136713681369137213731374137 5
905910592059305940595059605970598O   137613771378137913821383138413851 3
599060607060806090611061206130614O   861387138813891392139313941395139 6
615061606170618061906210622062306 2   139713981399141415141614171418141 9
406250626062706280629063106320633O   142214231424142514261427142814291 4
634063506360637063806390641064206 4   321433143414351436143714381439144 2
306440645064606470648064906510652O   144314441445144614471448144914521 4
653065406550656065706580659066106 6   531454145514561457145814591462146 3
206630664066506660667066806690671O   146414651466146714681469147214731 4
672067306740675067606770678067906 8   741475147614771478147914821483148 4
106820683068406850686068706880689O   148514861487148814891492149314941 4
691069206930694069506960697069806 9   951496149714981499151516151715181 5
907070807090711071207130714071507 1   191522152315241525152615271528152 9
607170718071907210722072307240725O   153215331534153515361537153815391 5
726072707280729073107320733073407 3   421543154415451546154715481549155 2
507360737073807390741074207430744O   155315541555155615571558155915621 5
745074607470748074907510752075307 5   631564156515661567156815691572157 3
407550756075707580759076107620763O   157415751576157715781579158215831 5
764076507660767076807690771077207 7   841585158615871588158915921593159 4
307740775077607770778077907810782O   159515961597159815991616171618161 9
783078407850786078707880789079107 9   162216231624162516261627162816291 6
207930794079507960797079807990808O   321633163416351636163716381639164 2
908110812081308140815081608170818O   164316441645164616471648164916521 6
819082108220823082408250826082708 2   531654165516561657165816591662166 3
808290831083208330834083508360837O   166416651666166716681669167216731 6
838083908410842084308440845084608 4   741675167616771678167916821683168 4
708480849085108520853085408550856O   168516861687168816891692169316941 6
857085808590861086208630864086508 6   951696169716981699171718171917221 7
608670868086908710872087308740875O   231724172517261727172817291732173 3
876087708780879088108820883088408 8   173417351736173717381739174217431 7
508860887088808890891089208930894O   441745174617471748174917521753175 4
895089608970898089909091109120913O   175517561757175817591762176317641 7
914091509160917091809190921092209 2   651766176717681769177217731774177 5
309240925092609270928092909310932O   177617771778177917821783178417851 7
933093409350936093709380939094109 4   861787178817891792179317941795179 6
209430944094509460947094809490951O   179717981799181819182218231824182 5
952095309540955095609570958095909 6   182618271828182918321833183418351 8
109620963096409650966096709680969O   361837183818391842184318441845184 6
971097209730974097509760977097809 7   184718481849185218531854185518561 8
909810982098309840985098609870988O   571858185918621863186418651866186 7
989099109920993099409950996099709 9   186818691872187318741875187618771 8
809991111211131114111511161117111 8   781879188218831884188518861887188 8
111911221123112411251126112711281 1   188918921893189418951896189718981 8
291132113311341135113611371138113 9   991919221923192419251926192719281 9
114211431144114511461147114811491 1   291932193319341935193619371938193 9
521153115411551156115711581159116 2   194219431944194519461947194819491 9
116311641165116611671168116911721 1   521953195419551956195719581959196 2
731174117511761177117811791182118 3   196319641965196619671968196919721 9
118411851186118711881189119211931 1   731974197519761977197819791982198 3
941195119611971198119912121312141 2   198419851986198719881989199219931 9
151216121712181219122212231224122 5   941995199619971998199922223222422 2
```

52226222722282229223322342235223622372238223922432244224452246224722482249225222552256225722582259226222652266226722682269227322772278227922802281228222832284228522862287228822892290229122922293229422952296229722982299230223032304230523062307230823092310231123122313231423152316231723182319232023212322232323242325232623272328232923302331233223332334233523362337233823392340234123422343234423452346234723482349235023512352235323542355235623572358235923602361236223632364236523662367236823692370237123722373237423752376237723782379238023812382238323842385238623872388238923902391239223932394239523962397239823992400240124022403240424052406240724082409241024112412241324142415241624172418241924202421242224232424242524262427242824292430243124322433243424352436243724382439244024412442244324442445244624472448244924502451245224532454245524562457245824592460246124622463246424652466246724682469247024712472247324742475247624772478247924802481248224832484248524862487248824892490249124922493249424952496249724982499250025012502250325042505250625072508250925102511251225132514251525162517251825192520252125222523252425252526252725282529253025312532253325342535253625372538253925402541254225432544254525462547254825492550255125522553255425552556255725582559256025612562256325642565256625672568256925702571257225732574257525762577257825792580258125822583258425852586258725882589259025912592259325942595259625972598259926002601260226032604260526062607260826092610261126122613261426152616261726182619262026212622262326242625262626272628262926302631263226332634263526362637263826392640264126422643264426452646264726482649265026512652265326542655265626572658265926602661266226632664266526662667266826692670267126722673267426752676267726782679268026812682268326842685268626872688268926902691269226932694269526962697269826992700270127022703270427052706270727082709271027112712271327142715271627172718271927202721272227232724272527262727272827292730273127322733273427352736273727382739274027412742274327442745274627472748274927502751275227532754275527562757275827592760276127622763276427652766276727682769277027712772277327742775277627772778277927802781278227832784278527862787278827892790279127922793279427952796279727982799280028012802280328042805280628072808280928102811281228132814281528162817281828192820282128222823282428252826282728282829283028312832283328342835283628372838283928402841284228432844284528462847284828492850285128522853285428552856285728582859286028612862286328642865286628672868286928702871287228732874287528762877287828792880288128822883288428852886288728882889289028912892289328942895289628972898289929002901290229032904290529062907290829092910291129122913291429152916291729182919292029212922292329242925292629272928292929302931293229332934293529362937293829392940294129422943294429452946294729482949295029512952295329542955295629572958295929602961296229632964296529662967296829692970297129722973297429752976297729782979298029812982298329842985298629872988298929902991299229932994299529962997299829993000300130023003300430053006300730083009301030113012301330143015301630173018301930203021302230233024302530263027302830293030303130323033303430353036303730383039304030413042304330443045304630473048304930503051305230533054305530563057305830593060306130623063306430653066306730683069307030713072307330743075307630773078307930803081308230833084308530863087308830893090309130923093309430953096309730983099310031013102310331043105310631073108310931103111311231133114311531163117311831193120312131223123312431253126312731283129313031313132313331343135313631373138313931403141314231433144314531463147314831493150315131523153315431553156315731583159316031613162316331643165316631673168316931703171317231733174317531763177317831793180318131823183318431853186318731883189319031913192319331943195319631973198319932003201320232033204320532063207320832093210321132123213321432153216321732183219322032213222322332243225322632273228322932303231323232333234323532363237323832393240324132423243324432453246324732483249325032513252325332543255325632573258325932603261326232633264326532663267326832693270327132723273327432753276327732783279328032813282328332843285328632873288328932903291329232933294329532963297329832993300330133023303330433053306330733083309331033113312331333143315331633173318331933203321332233233324332533263327332833293330333133323333333433353336333733383339334033413342334333443345334633473348334933503351335233533354335533563357335833593360336133623363336433653366336733683369337033713372337333743375337633773378337933803381338233833384338533863387338833893390339133923393339433953396339733983399340034013402340334043405340634073408340934103411341234133414341534163417341834193420342134223423342434253426342734283429343034313432343334343435343634373438343934403441344234433444344534463447344834493450345134523453345434553456345734583459346034613462346334643465346634673468346934703471347234733474347534763477347834793480348134823483348434853486348734883489349034913492349334943495349634973498349935003501350235033504350535063507350835093510351135123513351435153516351735183519352035213522352335243525352635273528352935303531353235333534353535363537353835393540354135423543354435453546354735483549355035513552355335543555355635573558355935603561356235633564356535663567356835693570357135723573357435753576357735783579358035813582358335843585358635873588358935903591359235933594359535963597359835993600360136023603360436053606360736083609361036113612361336143615361636173618361936203621362236233624362536263627362836293630363136323633363436353636363736383639364036413642364336443645364636473648364936503651365236533654365536563657365836593660366136623663366436653666366736683669367036713672367336743675367636773678367936803681368236833684368536863687368836893690369136923693369436953696369736983699370037013702370337043705370637073708370937103711371237133714371537163717371837193720372137223723372437253726372737283729373037313732373337343735373637373738373937403741374237433744374537463747374837493750375137523753375437553756375737583759376037613762376337643765376637673768376937703771377237733774377537763777377837793780378137823783378437853786378737883789379037913792379337943795379637973798379938003801380238033804380538063807380838093810381138123813381438153816381738183819382038213822382338243825382638273828382938303831383238333834383538363837383838393840384138423843384438453846384738483849385038513852385338543855385638573858385938603861386238633864386538663867386838693870387138723873387438753876387738783879388038813882388338843885388638873888388938903891389238933894389538963897389838993900390139023903390439053906390739083909391039113912391339143915391639173918391939203921392239233924392539263927392839293930393139323933393439353936393739383939394039413942394339443945394639473948394939503951395239533954395539563957395839593960396139623963396439653966396739683969397039713972397339743975397639773978397939803981398239833984398539863987398839893990399139923993399439953996399739983999400040014002400340044005400640074008400940104011401240134014401540164017401840194020402140224023402440254026402740284029403040314032403340344035403640374038403940404041404240434044404540464047404840494050405140524053405440554056405740584059406040614062406340644065406640674068406940704071407240734074407540764077407840794080408140824083408440854086408740884089409040914092409340944095409640974098409941004101410241034104410541064107410841094110411141124113411441154116411741184119412041214122412341244125412641274128412941304131413241334134413541364137413841394140414141424143414441454146414741484149415041514152415341544155415641574158415941604161416241634164416541664167416841694170417141724173417441754176417741784179418041814182418341844185418641874188418941904191419241934194419541964197419841994200420142024203420442054206420742084209421042114212421342144215421642174218421942204221422242234224422542264227422842294230423142324233423442354236423742384239424042414242424342444245424642474248424942504251425242534254425542564257425842594260426142624263426442654266426742684269427042714272427342744275427642774278427942804281428242834284428542864287428842894290429142924293429442954296429742984299430043014302430343044305430643074308430943104311431243134314431543164317431843194320432143224323432443254326432743284329433043314332433343344335433643374338433943404341434243434344434543464347434843494350435143524353435443554356435743584359436043614362436343644365436643674368436943704371437243734374437543764377437843794380438143824383438443854386438743884389439043914392439343944395439643974398439944004401440244034404440544064407440844094410441144124413441444154416441744184419442044214422442344244425442644274428442944304431443244334434443544364437443844394440444144424443444444454446444744484449445044514452445344544455445644574458445944604461446244634464446544664467446844694470447144724473447444754476447744784479448044814482448344844485448644874488448944904491449244934494449544964497449844994500450145024503450445054506450745084509451045114512451345144515451645174518451945204521452245234524452545264527452845294530453145324533453445354536453745384539454045414542454345444545454645474548454945504551455245534554455545564557455845594560456145624563456445654566456745684569457045714572457345744575457645774578457945804581458245834584458545864587458845894590459145924593459445954596459745984599460046014602460346044605460646074608460946104611461246134614461546164617461846194620462146224623462446254626462746284629463046314632463346344635463646374638463946404641464246434644464546464647464846494650465146524653465446554656465746584659466046614662466346644665466646674668466946704671467246734674467546764677467846794680468146824683468446854686468746884689469046914692469346944695469646974698469947004701470247034704470547064707470847094710471147124713471447154716471747184719472047214722472347244725472647274728472947304731473247334734473547364737473847394740474147424743474447454746474747484749475047514752475347544755475647574758475947604761476247634764476547664767476847694770477147724773477447754776477747784779478047814782478347844785478647874788478947904791479247934794479547964797479847994800480148024803480448054806480748084809481048114812481348144815481648174818481948204821482248234824482548264827482848294830483148324833483448354836483748384839484048414842484348444845484648474848484948504851485248534854485548564857485848594860486148624863486448654866486748684869487048714872487348744875487648774878487948804881488248834884488548864887488848894890489148924893489448954896489748984899490049014902490349044905490649074908490949104911491249134914491549164917491849194920492149224923492449254926492749284929493049314932493349344935493649374938493949404941494249434944494549464947494849494950495149524953495449554956495749584959496049614962496349644965496649674968496949704971497249734974497549764977497849794980498149824983498449854986498749884989499049914992499349944995499649974998499950005001500250035004500550065007500850095010501150125013501450155016501750185019502050215022502350245025502650275028502950305031503250335034503550365037503850395040504150425043504450455046504750485049505050515052505350545055505650575058505950605061506250635064506550665067506850695070507150725073507450755076507750785079508050815082508350845085508650875088508950905091509250935094509550965097509850995100510151025103510451055106510751085109511051115112511351145115511651175118511951205121512251235124512551265127512851295130513151325133513451355136513751385139514051415142514351445145514651475148514951505151515251535154515551565157515851595160516151625163516451655166516751685169517051715172517351745175517651775178517951805181518251835184518551865187518851895190519151925193519451955196519751985199520052015202520352045205520652075208520952105211521252135214521552165217521852195220522152225223522452255226522752285229523052315232523352345235523652375238523952405241524252435244524552465247524852495250525152525253525452555256525752585259526052615262526352645265526652675268526952705271527252735274527552765277527852795280528152825283528452855286528752885289529052915292529352945295529652975298529953005301530253035304530553065307530853095310531153125313531453155316531753185319532053215322532353245325532653275328532953305331533253335334533553365337533853395340534153425343534453455346534753485349535053515352535353545355535653575358535953605361536253635364536553665367536853695370537153725373537453755376537753785379538053815382538353845385538653875388538953905391539253935394539553965397539853995400540154025403540454055406540754085409541054115412541354145415541654175418541954205421542254235424542554265427542854295430543154325433543454355436543754385439544054415442544354445445544654475448544954505451545254535454545554565457545854595460546154625463546454655466546754685469547054715472547354745475547654775478547954805481548254835484548554865487548854895490549154925493549454955496549754985499550055015502550355045505550655075508550955105511551255135514551555165517551855195520552155225523552455255526552755285529553055315532553355345535553655375538553955405541554255435544554555465547554855495550555155525553555455555556555755585559556055615562556355645565556655675568556955705571557255735574557555765577557855795580558155825583558455855586558755885589559055915592559355945595559655975598559956005601560256035604560556065607560856095610561156125613561456155616561756185619562056215622562356245625562656275628562956305631563256335634563556365637563856395640564156425643564456455646564756485649565056515652565356545655565656575658565956605661566256635664566556665667566856695670567156725673567456755676567756785679568056815682568356845685568656875688568956905691569256935694569556965697569856995700570157025703570457055706570757085709571057115712571357145715571657175718571957205721572257235724572557265727572857295730573157325733573457355736573757385739574057415742574357445745574657475748574957505751575257535754575557565757575857595760576157625763576457655766576757685769577057715772577357745775577657775778577957805781578257835784578557865787578857895790579157925793579457955796579757985799580058015802580358045805580658075808580958105811581258135814581558165817581858195820582158225823582458255826582758285829583058315832583358345835583658375838583958405841584258435844584558465847584858495850585158525853585458555856585758585859586058615862586358645865586658675868586958705871587258735874587558765877587858795880588158825883588458855886588758885889589058915892589358945895589658975898589959005901590259035904590559065907590859095910591159125913591459155916591759185919592059215922592359245925592659275928592959305931593259335934593559365937593859395940594159425943594459455946594759485949595059515952595359545955595659575958595959605961596259635964596559665967596859695970597159725973597459755976597759785979598059815982598359845985598659875988598959905991599259935994599559965997599859996000

Building a Community Forum

by Freaky

We've seen user-input communications in various incarnations for years, from bulletin boards and newsgroups to mailing lists and forums. I have been building communities for the vast part of a decade and have found success in building communities covering a wide range of subject matter from coffee and medical research to technology and hobby sites. This simple guide will get you on your way to building your own community.

Choosing Your Subject

If you already have a website, your subject is probably obvious, but in either case it's important to pick a subject that's familiar, sparks interest, and is something you are passionate about that you think other people will talk about.

An example of a community that grew out of interest is one of the largest lockpicking enthusiasts' websites, LockPicking101.com. This community began with local hackers expressing an interest in learning about lockpicking. The forum provided a place for hacker/lockpicking enthusiasts to share informative tidbits they learned about lockpicking with each other. Soon it became apparent that others were also interested in lockpicking and how locks worked. The site recently celebrated its ten year anniversary!

Once you have your subject, research possible names for the community. See what exists already as you don't want your site confused with someone else's site. If you already have a site and you're creating a new domain name just for your forum, it's vital that you choose a memorable name for your community. Consider keywords that are directly associated with your subject and your target demographic. In addition, it's a necessity that you determine which title would be the most search-friendly. Luckily there are many keyword research tools that can provide you with the pertinent and popular search words!

Select Your Software

There are quite a few web-based software solutions for your needs, the most popular being vBulletin and phpBB. When selecting your forum software, it's important to select one that is updated and maintained. Running software that isn't maintained can lead to hacked sites and servers. Some hosting providers help install the software. Others update the software for you, but if you're running your own services, it's on you to keep it up to date!

Once you make your selection and install it, plan on sticking with it for a while. It's rather hard to migrate to different forum software, especially when the site grows. CellPhoneHacks.com is one example of migrating from phpBB to vBulletin. The site was run on phpBB for years, constantly being updated, so when it was migrated to vBulletin, the automated tools weren't as automated as we wanted and were riddled with errors that took a great deal of work to get running again. Sometimes migration goes smoothly, other times it doesn't, so always have backups of your database and files!

Choose Main Topics of Discussion

A forum with too many sections and no posts is like a ghost town. When visitors hit the site, seeing everything empty, they tend to press the back button because they feel their post won't get seen. When selecting your main sections of discussion, start with a couple and make sure you get some great posts in the section, so people see they are active and the community is alive. You can always add more as your community grows, but it's good to start with just a handful. This was experienced first-hand with UndercoverFiles.com, a community for conspiracy and doomsday preparations. As you can imagine, there are so many topics that could be covered, but we had too many at first and had to scale back and combine subjects.

Once your main subjects are created, you're going to want to seed your community. Start by making some posts yourself and ask your friends to make a post or two and get involved! Ideas for starter topics include rules and introduction topics. Reach out to other sites that may want to get involved to help the site grow! Remember you need that warm feeling that the site is alive and active.

Adapt to Your Audience

Your audience speaks and you will be able to see what's of interest to them and what they're talking about. Even though you may start a forum with one person in mind, you may realize you actually attracted a different kind of user. BaristaForums.com, which was previously espressoforums.com, was intended to be a site for coffee lovers to talk about coffee. Once traffic started coming to the site, we realized it was full of coffee shop owners and employees looking to grow their business, talk b2b, and learn about the latest tech in their industry. The site was adapted and the caffeinated talk is still buzzing! Keep your ear to the ground and adapt to your community.

Promote Your Community

Books have been written on promoting websites; the key thing is you need to promote your site, and you have a lot of tools and resources at your fingertips, including tons of blog posts and community sites like webmasterworld.com which I first started out on. Without spending any more money, you can research search engine optimization (SEO) and start writing better posts that will attract more search engine traffic.

Don't spam other sites. Remember, you're trying to build a community and you probably don't want people spamming your site, so make sure you don't spam other people's sites. Many sites allow you to have signature tags and have link sections. But the best kind of traffic isn't from one link, it's from a recommendation. Start making friends and get involved in other communities and other sites by doing guest posts and writing great content.

We've promoted communities locally by printing flyers and business cards and posted at colleges, coffee shops, and other businesses to help drive traffic to the sites. Make these flyers available to other users on the forum so they can help spread the word! While some sites we've promoted via social media, others are promoted as paid advertisements on Google or banners on other websites.

Keep Your Community Clean

We've seen sites get obliterated by spam bots, so it's important to keep your forums clean, updated, and protected against the spam bots. There are different methods you can try to keep spam out, including enabling captcha to stop automated registration and posts, but there are also third party solutions like blockscript.com which allows the webmaster to input a bit of code to check to see if the connection is made via proxies, known IPs of spam, or certain countries and then rejects the connection.

Your number one protection against spam is your own community users and forum administrators. You will want to select a moderator or administrator to help keep the community clean, someone who is active in the community and has the free time to help keep it clean. Some forum software allows users to report posts. When enough users report a post, it is removed automatically and put under review. Your moderators and administrators may move on to other things over time or get too busy to be as active as they were, so keep an open line of communication and be aware of what is needed. You don't want to neglect your community. It can easily be overtaken by spam. At that point, it's best to shut it down or disable new posts if you want to keep the old content accessible to users as reference.

Procedural Worlds Statistical Analysis Image Processing and PRNG Exploitation for the Lulz — or Why IMDb Got a Captcha

by sam

In February 2005, the GNAA devised a cunning plan to troll IMDb users using various fancy hacks. This is what happened.

The Plan

It was suggested on the #gnaa IRC channel that the movie *Gayniggers from Outer Space* (*GNFOS*), from which the organization takes it name, be upvoted to the IMDb Top 250 as an emotional tribute to this cult movie. The GNAA, not being 4chan, did not have an army of idiots to carry out their deeds; they had to use skills and technology instead.

The first attempt was simple: everyone voted for *GNFOS*, and asked people they knew to vote as well. It went slowly. In order to vote several times, a person had to go through a heavy process: only registered users can vote on IMDb, and a valid email address is required in order to register an account. Manual account creation was slow. The GNAA therefore decided to automate the IMDb account creation.

Creating a Procedural World of People

The following observations and guesses were made about the IMDb voting process:

- For the Top 250, only votes from "regular voters" were considered. This probably meant that in order to have an impact on the vote, they needed to A) vote for several movies in addition to *GNFOS* and B) have the same accounts vote again in the following days.
- A "weighting system" was applied to the votes, which probably included disfavoring votes from the same IP address, so they needed to use as many different IPs as possible.
- Multiple email addresses from the same domain were more likely to attract attention, so using as many different domains as possible would make it more difficult to deduce which other accounts were created using this process.
- New users needed to fill out a form with their gender, birth year, country, postal code, etc. Randomizing this information would reduce the odds of being detected through statistical analysis.
- So the GNAA wrote an account creation library that, given a random seed, would create a unique identity comprising of:
- Full name, using data from the most common female and male names, as well as surnames in the U.S.
- Email address, using the full name combined with a variety of free email providers such as spam.la, mailinator.com, fastmail.us...
- Gender, country, year of birth, postal code.
- A preferred password for use on websites.

Generated identities would then look like this:

```
SEED, FULL NAME, GENDER,
➥ EMAIL ADDRESS, PASSWORD
3480, Tracy Gilbert, F, Tracy
➥Gilbert@spamhole.com, 26ACTR41
3481, Rene Reid, M, Rene_Reid@
➥runbox.com, Re96RE14
3482, Sandra Silva, F, SANDRA63@
➥swiftmail.com, UA75ED11
3483, Terrence Bowman M, terren
➥cebowman@spamhole.com, en29TETE
3484, Ian Wade, M, WADE5946@po
➥boxed.com, 59DE28WA
3485, Barbara Burke   F, barbara
➥_burke@spam.la, rb86BA13
```

People taking part in the operation would then be responsible for a seed range. For instance, Gary would run a script with seeds 1400 to 1499 for several days. But if Gary became busy, someone else could run the script with the same seed range and continue where he had left off. There was no need to create a central database because all of the identity information was generated procedurally.

Operation imdbtroll

The GNAA combined the identity creation library with additional anonymizing features such as a regularly updated list of public HTTP proxies (Tor was barely usable back in 2005), and web user agent randomization. The imdbtroll.py script was created.

People on IRC started running the script with a seed range assigned to them. The script went through several iterations, but the final version worked roughly as follows:

1. Choose a seed from the provided range, and create the corresponding identity.
2. Check whether the identity's email address is activated, by logging in if necessary. For instance, a spam.la account didn't require any subscription. But a mailinator.com account did. If the email address is not active, register an account at the email provider.
3. Check whether the IMDb account is present, by logging in if necessary. If the IMDb account is not present but there is a confirmation email in the mailbox, activate it. If the IMDb account is not present and there is no email, create an IMDb account and wait for a confirmation email in the mailbox.
4. Log in to IMDb.
5. Vote for movies from IMDb's Top 250, from the bottom 100, or using its built-in search engine; random search words included "troll," "communists," or "nazis."
6. Vote for *Gayniggers from Outer Space,* giving that movie 8, 9, or 10 stars.

The script also tried hard to simulate a real human using a real web browser, pausing between pages, using valid referrer information, clicking on links, sometimes not even voting for *GNFOS*....

It worked well. The weighted average vote for *GNFOS* went from 5.9 stars to 8.7.

```
FEB 2ND   FEB 3RD   FEB 4TH
5.9/10    7.5/10    8.7/10
```

And here are the voting details:

```
      FEB 2ND, FEB 3RD, FEB 4TH
10    605 (68.0%), 1391 (81.2%),
➥ 2913 (81.8%)
9      26 (2.9%), 60 (3.5%),
➥ 224 (6.3%)
8      24 (2.7%), 25 (1.5%),
➥ 85 (2.4%)
7      28 (3.1%), 28 (1.6%),
➥ 55 (1.5%)
6      28 (3.1%), 29 (1.7%),
➥ 51 (1.4%)
5      33 (3.7%), 33 (1.9%),
➥ 51 (1.4%)
4      18 (2.0%), 18 (1.1%),
➥ 37 (1.0%)
3      27 (3.0%), 27 (1.6%),
➥ 35 (1.0%)
2      30 (3.4%), 30 (1.8%),
➥ 37 (1.0%)
1      71 (8.0%), 71 (4.1%),
➥ 72 (2.0%)
```

Bantown Trolls the GNAA

On February 4th, Bantown, a rival trolling group, got ahold of the GNAA's script by lurking on the IRC channel and using powerful hacker tools such as wget to retrieve the publicly posted script updates.

Bantown started running imdbtroll.py, too, with their own secret seed ranges. They just made one single modification to it: instead of giving *GNFOS* ten stars, they were giving it one star.

A race had begun. It was obvious that Bantown was running more instances of the script than the GNAA, so that they could completely cancel the GNAA's efforts. One solution was to run even more instances than Bantown, but a weapon escalation could only mean the eventual detection of unusual behavior by IMDb admins.

But the GNAA had a secret weapon: a logic bomb hidden in plain sight, right inside imdbtroll.py.

The GNAA Trolls Bantown Back

The library used for IMDb access had a lot of features, including changing a user's password. It was not used by imdbtroll.py, but it was fully functional. The GNAA therefore created a new script, fuckbantown.py, which did the following:

- Create a new identity from a random seed.
- Log into IMDb using the identity.
- Change the user's password so that the account becomes unusable for Bantown's running scripts.
- Change the vote for GNFOS from 1 star back to 10.

There was only one small problem: the GNAA did not know what random seeds Bantown had been using. They would have to potentially log in to billions of possible accounts in order to find out which users were created. That was not only guaranteed to raise alarms at IMDb, but it was also practically unfeasible in a reasonable amount of time.

But there was another way, thanks to spam.la. Some of the identities were using that

domain for their email address.

One prominent feature of spam.la was that *all* emails sent to a spam.la address appeared on the website. ("All email sent to any_address@spam.la is publicly readable right here" is what was said on their site.) So the GNAA only had to monitor that website and look for unknown IMDb account activation emails! Then, if the confirmation email was sent to, say, TRACEY49@spam.la, they only had to brute-force the Python pseudorandom number generator in order to find the seed that had created such an address. That still meant testing all possible seeds, but without having to connect to any server. If the seed was 215045, it probably meant that a Bantown person was using seeds 215000 to 215999.

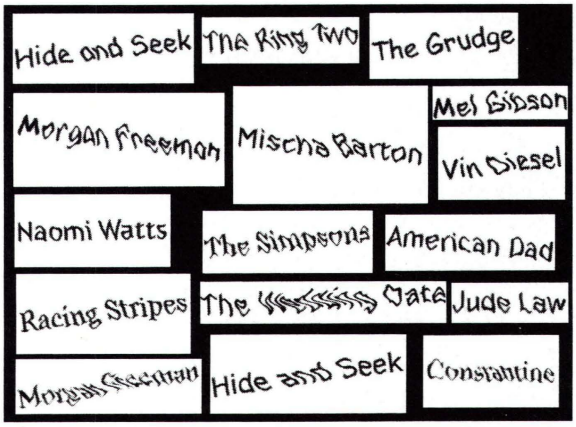
Little by little, the GNAA secretly changed the votes for the users that Bantown had spent hours creating.

The IMDb Captcha

Understandably, the Bantown people felt butthurt. On February 5th, they decided to put an end to the whole operation and they alerted IMDb. A wave of panic swept over the admins and one of them quickly set up a captcha composed of a random movie or actor name to protect account creation from automated scripts.

Back in 2005, captcha breaking was rather uncommon. Some tools existed, but they only targeted simple captchas with minor image distortions. The one used by IMDb was considered hard to break.

However, the captcha had an unexpected weakness. It took the GNAA some time to understand it but, with a few samples, it had become visible:

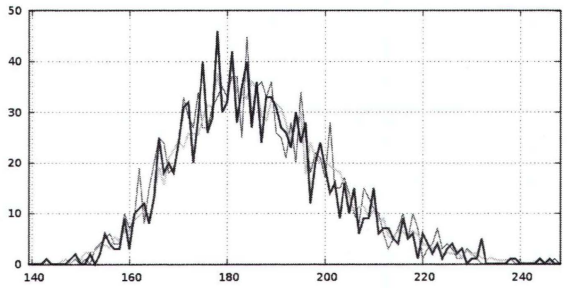


Can you see it? "Morgan Freeman" and "Hide and Seek" appeared twice each. What were the odds that, given 16 movie and actor names chosen at random, two of them would appear more than once? Pretty small, wouldn't you agree? Well, yes, unless the list of movies and actors was unexpectedly short. And a small dictionary is a serious captcha weakness.

In order to guess the size of the dictionary, the GNAA gathered 192 captcha samples and counted how many times duplicates appeared:

66 names appeared once
39 names appeared twice
10 names appeared 3 times
3 names appeared 4 times
1 name appeared 6 times

They then performed a statistical analysis and managed to compute the probability that the above distribution would appear given various dictionary sizes:



The most probable dictionary sizes were between 170 and 190. As expected, that was *small* and allowed for a captcha breaking attack that did not involve OCR. Given the size of the corpus, they only had to count characters instead of decoding them. For instance, four characters followed by seven characters could be "Pulp Fiction," "Ryan Gosling," or "Teri Hatcher." Since three tries were allowed to solve the captcha, that one would always be successfully guessed. In average, this led to a captcha breaker that had more than 60 percent efficiency.

Operation imdbtroll could carry on.

Epilogue

A few hours after the captcha breaker was integrated into imdbtroll.py, someone on #gnaa pointed out that the IMDb Top 250 only allowed movies that ran for more than 45 minutes.

GNFOS was a short movie. It would never enter the Top 250.

The whole operation had been in vain, but science progressed and lulz were had.

At Home Malware (and Online Ads) Protection

by Ashes

As we know, even the most security-aware person can be subject to redirects, mis-clicks, etc. So when I found a host file online containing known malware websites, I immediately wanted to load this file onto my Ubuntu machine to protect it. However, I have a lot of other devices on my network as well, including my media computer for streaming movies and wireless devices such as tablets and phones. Loading a hosts file onto each one of these devices and updating them every time the malware hosts file was updated online would be more work than I wanted to do.

Having DD-WRT on my home router would be the answer to zero work after the initial configuration. To implement my solution, I used SSH to connect to my router. In the root's home directory I then wrote the following script:

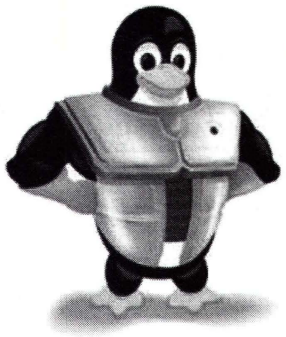
```
#!/bin/sh

wget -O ~/malware_hosts.txt
➥ http://www.malwaredomainlist
➥.com/hostslist/hosts.txt

wget -O ~/ad_hosts.txt http://
➥www.winhelp2002.mvps.org/hosts
➥.txt

cp -f /tmp/hosts /tmp/hosts.bkp

cat ~/malware_hosts.
txt > /tmp/hosts
cat ~/ad_hosts.txt >> /tmp/hosts

rm -f ~/malware_hosts.txt
rm -f ~/ad_hosts.txt
killall dnsmasq
dnsmasq --conf-file=/tmp/dnsmasq.
➥conf
```

To explain this script to those who may not understand, the script downloads the updated malware hosts file from www.malwaredomainlist.com and, for good measure, another list with advertising domains. It then creates a backup of the current hosts file and copies the contents of the downloaded malware hosts file and advertising hosts file into the proper hosts file to be read by the operating system. After this happens, the script then removes the two downloaded files, kills the current DNS service, and restarts it so that the hosts file can be properly read.

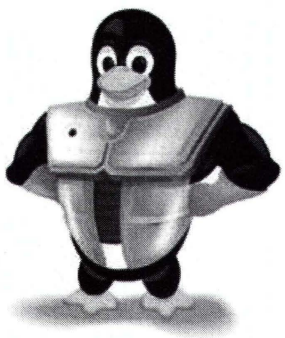
I then ran the script to ensure that it did not error out as well as making sure the malware and advertisement website list was copied into the hosts file. After it did not error out and everything was a go, I created a weekly cron job. I added a file "update_malware_blocks" into the /tmp/cron.d directory with the proper configuration so that it runs weekly.

Some additional notes on this configuration:

1. The /tmp directory gets reset every time the router is rebooted. If you have to reboot your router, you will have to re-implement the steps above.
2. The home directory for root on DD-WRT is in the /tmp/root directory.
3. Your clients must be set to use your router as your DNS server. Then, of course, use OpenDNS servers to further resolve requests by putting their IP addresses into your router settings via the web GUI.



Automated Target Acquisition



by r00tNinjas: 0rbytal & blerbl
0rbytal@burntmail.com &
theblerbl@gmail.com

"Invincibility lies in the defense; the possibility of victory in the attack." - Sun Tzu

Whether you are tired of hackers messing with your server (defense), you've got mad hacking skills and no targets (offense), or perhaps both, this article should interest you. I will briefly explain a brilliant system set up by my friend blerbl because he's a techno-hacker genius who doesn't like to write, and I'm a fairly decent writer who thought my *2600* brethren would love to replicate his defensive web server configuration. But first, the standard disclaimer:

This information is strictly for educational purposes. You should not try this outside of your own personally owned and operated test network. Any consequences resulting from your application of the knowledge shared in this article are your own fault. Do not try this at home.

blerbl runs his own web server, mostly as a front lobby to host various files he wants to access from any remote location with Internet access. As any web administrator who monitors their server will notice, the number of automated scans occurring across the Internet is prodigious. He doesn't mind being scanned, but he'd prefer they not launch remote file inclusion attacks to enlist him in their botnet.

Like most savvy web administrators, blerbl uses a robots.txt file on his server to politely ask the courteous web crawlers to refrain from searching or indexing specific directories. Of course, blerbl also knows that the cunning hackers look for "robots.txt" files on web servers because they often contain the file paths that are much more interesting than what is published on the server. With this in mind, blerbl makes sure to include in his "robots.txt" file paths to tantalizing pages like "/myadmin.php" as sort of a "honey pot" for the nefarious hackers and inconsiderate web crawlers. To avoid copying the honeypot page a thousand times, renaming it as every permutation of myadmin.php, blerbl used the mod_rewrite engine of Apache to help him accomplish his goal.

When a user requests "/myadmin.php" on his website, the user's IP address is added to a special log file. He added a rule to his Apache configuration that will compare all requests with requests filed in the special log. If the request matches a logged IP from the special log, the request is transparently modified to become a request for the trap page... again. To reinforce his intent, blerbl added a rule at the top of his configuration file that compares the requestor's IP address with the special log file and serves an error page if the requestor has ever previously accessed the trap page.

This routine prevents malicious users from accessing his server from that IP address, as was blerbl's intent, but this method isn't just an effective defensive measure... remember that when the user is blacklisted, his IP address is logged by the server in the special log file. Most casual Internet users will only browse the pages that are linked, and have no interest in a "robots.txt" file, or any page listed in it. Who has any interest in browsing pages and files listed in the "robots.txt" file? Hackers.

The special log file containing the blacklisted IP addresses can now be used as a targeting list! Clever and careful hackers won't hack directly from their own IP address... they use somebody else's. So, the blacklisted IP addresses likely belong to either (A) noobs who don't really know what they're doing, (B) script kiddies who disregard stealth, or (C) compromised systems. Regardless of the type of user that scanned the web server, the admin can now scan the scanner with a fair probability they can gain access (if the admin had the time/interest). It's kind of like being an active agent of karma, teaching hackers the golden rule through the most effective (and often merciless) teacher: experience.

Another benefit to this defense implementation is that the web admin can add rules to discriminate based on user agents that script kiddies often use, or any other screening parameter. Plus, the blacklist can be modified or manually updated without having to restart the server. The customization possibilities are endless. Below is the code for the auto-blacklisting files you can use to defend your web server, or to automate your target acquisition.

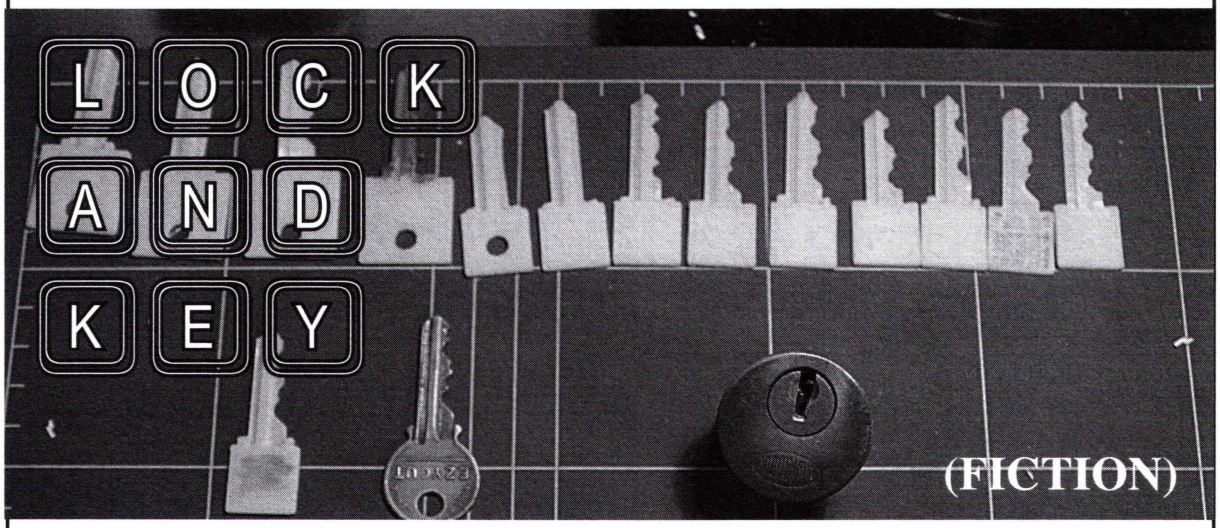
Hack *All* The Things!

[security.conf]

```
#include the desired site's
➥ conf file
RewriteEngine ON
#RewriteLog rwlog.log
#RewriteLogLevel 5
## BLACKLIST IPS ##
RewriteMap ipslist txt:/etc/
➥security/blacklistip
RewriteCond %{REMOTE_ADDR} ^(.*)$
RewriteCond ${ipslist:%1|white}
➥ ^black$ [NC]
RewriteRule (.*) - [F]
## TRAP REQUESTS ##
RewriteMap reqlist txt:/etc/
➥security/bad_requests
RewriteCond %{REQUEST_URI}
➥ ^/*(\S+)/*$ [NC]
RewriteCond ${reqlist:%1|white}
➥ ^black$ [NC]
RewriteRule (.*) "/trap.php" [L]
## RFI Prevention ##
RewriteCond %{THE_REQUEST} GET\
➥ ((http)|(ftp))(://|s://)+.*
RewriteRule (.*) "/trap.php" [L]
```

[trap.php]

```
<html>
        <title> Oh my </title>
        <body>
        <center><p>Now what ???
➥</p></center>
        </body>
<?php
    $bl_filename = "/etc/security
➥/blacklistip";
    $f = fopen($bl_filename,'a');
    $msg = $_SERVER['REMOTE_ADDR
➥']."\tblack\n";
    fwrite($f,$msg);
    fclose($f);

    $bl_filename = "/etc/security
➥/offenses.log";
    $f = fopen($bl_filename,'a');
    $msg = $_SERVER['REQUEST_
➥TIME']."\t".$_SERVER['REMOTE_
➥ADDR']."\n";
    fwrite($f,$msg);
    fclose($f);
?></html>
```



by Robert B. Schofield

I was at my local hackerspace finishing up my Arduino powered 3D LED grid. It was currently displaying a 3D falling rain of light, when I heard a voice over my shoulder.

"Nice. Very nice," it said, in an accent I didn't recognize.

I turned to see a skinny man in jeans and a leather jacket, with an English driving cap that bore the Union Jack across the top. He had a bushy mustache, and a small triangular beard. His accent was not British.

"Thanks," I replied, turning back to my project. Next was a wall of light going front to back, left to right, then top to bottom.

"My name is Boleslav," the man behind me said.

"Lock," I said, turning around.

"That is an interesting name, and very

appropriate for what I would like to discuss with you."

"And what is that?" I asked.

"This." He reached into his jacket and slowly pulled out a key. It was made of glass.

"What?"

He held it out and I took it. I immediately realized it was not glass, but clear plastic, and there were tiny fiber optic lines inside. I also noticed that the teeth of the key were all at the minimum key depth. It was a bump key. A clear plastic fiber optic bump key. What in the world?

"Can I buy you a drink?" Boleslav asked.

Two beers later at the local dive a few blocks down, and I asked Boleslav, "Where did you get that?"

"No. No questions about that," he said, holding up a hand. "Yes, it is unique. Very special. And I have a proposition for you. Would you like to work with it?"

"Sure. I mean, maybe. Work with it how?"

He grinned. I think he knew he had me. Still, I was wary. This was not something you find online, not even on Silk Road, before that was shut down.

He pulled out the key again and held it between us. "Notice the op-tiks," he said, in his strange accent.

I did. They led from each of the teeth to a small connector on the bow of the key.

"It connects to a ka-mara," he said. "A digital ka-mara."

I shrugged.

"You know what a bump key is?" he asked.

"Of course. The teeth are all at the lowest possible point. You insert the key, then back it out a notch. You tap the key with a bump hammer while you turn it slightly, and if you're lucky, it opens the lock.

Boleslav grinned. "Exactly. If you are lucky." He held up the key. "This is for something better, to eliminate the luck." He took a big swig of beer. "You have a MakerBot at your haker-space. You put this key in a lock, bump and take picture inside the lock. Ha! Your name!" He patted my shoulder. "Then you turn picture into real key. What do you say?"

Interesting. It seemed possible, at least in theory. A unique challenge. If the key really worked. "Maybe," I said, thinking. "Not an Arduino, it'll need a computer."

"Must be small," Boleslav said.

"A Raspberry Pi," I replied. "But I don't have one, or the time to work on it."

"I have Bitcoins," he said. "I have been mining since the start. Ten now, and ninety more when it works."

I took the key from him, turned it around in my fingers, and nodded.

Of course this was not on the up and up. I was not that naive. But it was a challenge. And what a unique key. I cashed the Bitcoins and got the Raspberry Pi. The key and small digital camera that connected to it that he gave me worked. When I bumped the key in a lock the camera flashed the inside of the cylinder and pins and took about a hundred pictures. I wrote some Python to calculate the proper length of the cylinders based on the pictures and then convert those to key teeth height. A friend of mine, Sh0kwave, helped me turn that data into a MakerBot file, and easier than I expected, it worked! I could print a working, plastic key that was easily strong enough to work in a lock.

Bump, snap, calculate, print, and you had a working key.

Boleslav met me at the bar, very excited, when I contacted him. "Very good! Excellent!" he said when I showed him the plastic key and how it opened the test lock I'd used. "Give to me and I will transfer the rest of the Bitcoins."

"Are you sure you'll transfer them?" I asked.

"Yes, of course! You have my phone number, my email."

I hesitated a moment, then said, "OK," and handed over the equipment.

I checked the next day, but of course there was no transfer. I didn't really expect it. I'd traced his email through the header and his phone number and knew they were both throw-aways. That was OK. It had been a fun challenge. I still got ten Bitcoins to help fund my next project. I knew Boleslav was a crook, or a spy, or something. That was OK too, because the next time Boleslav, or whoever he worked for tried to use the gear I'd given him they would get a nice little surprise. It would make a small object, but not a key. They would slowly see a very small human fist appear. And as they continued to watch, they would see that the fist had a single finger extended.

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 10-13
Notacon 11
Cleveland Marriott East
Warrensville Heights, Ohio
www.notacon.org

April 17-21
Easterhegg 2014
Kulturhaus Arena
Stuttgart, Germany
eh14.easterhegg.eu

April 26-27
Maker Faire UK
Centre for Life
Newcastle, England
www.makerfaireuk.com

May 16-18
CarolinaCon 10
North Raleigh Hilton
Raleigh, North Carolina
www.carolinacon.org

May 17-18
Maker Faire Bay Area
San Mateo Event Center
San Mateo, California
www.makerfaire.com

June 5-6
RVAsec
Commonwealth Ballroom
Virginia Commonwealth
University Campus
Richmond, Virginia
rvasec.com

June 13-15
CircleCityCon
Hyatt Regency Indianapolis
Indianapolis, Indiana
circlecitycon.com

June 28-29
Nuit Du Hack
Disneyland Paris Convention
Center
Paris, France
www.nuitduhack.com

June 28-29
Maker Faire Kansas City
Union Station
Kansas City, Missouri
www.makerfaire.com

July 9-13
ToorCamp 2014
Hobuck Beach Resort
Neah Bay, Makah Indian
Reservation, Washington
toorcamp.org

July 18-20
HOPE X
Hotel Pennsylvania
New York, New York
x.hope.net

July 26-27
Maker Faire Detroit
The Henry Ford
Dearborn, Michigan
www.makerfaire.com

August 7-10
DEF CON 22
Rio Hotel and Casino
Las Vegas, Nevada
www.defcon.org

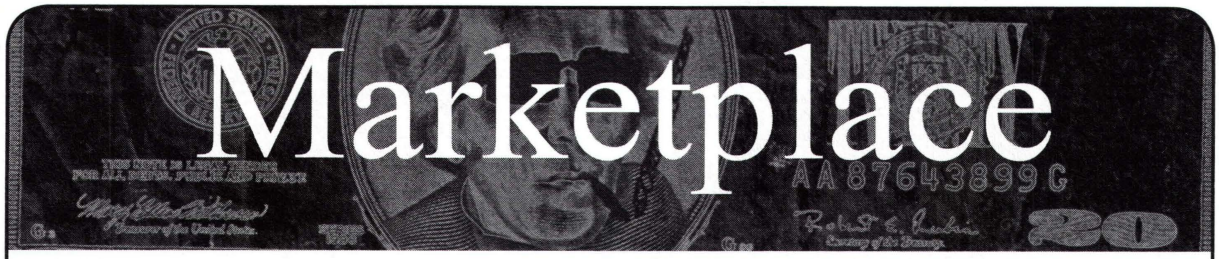
September 20-21
World Maker Faire New York
New York Hall of Science
Queens, New York
www.makerfaire.com

September 24-28
DerbyCon
Hyatt Regency
Louisville, Kentucky
www.derbycon.com

October 16-17
GrrCON
DeVos Place
Grand Rapids, Michigan
www.grrcon.org

December 27-30
Chaos Communication Congress
Congress Center Hamburg
Hamburg, Germany
www.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

HOPE X. *2600* presents the tenth Hackers On Planet Earth conference at New York City's HOtel PEnnsylvania July 18-20, 2014. Visit xxx.xxxxxxxxxxxxxxxxxx.xxx or x.hope.net for the latest news, travel info, special hotel rates, etc. Speakers wanted: email speakers@hope.net. Volunteers wanted: email volunteers@hope.net. Vendors wanted: email vendors@hope.net. Projects wanted: email projects@hope.net. You get the idea. You can help define what HOPE X focuses on and be a real part of hacker history, right in the middle of midtown Manhattan, across the street from the busiest train station in America. You can also join our announcement mailing list from the main page of our websites. Call (212) PEnnsylvania 6-5000 for the special conference room rate.

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.

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.

CAPT'N CRUNCH WHISTLES. Only a few left. THIS IS *THE ORIGINAL* WHISTLE from Capt'n Crunch cereal box. Brand new, unused, mint condition! Join the elite few who own this treasure! Once the remaining few are sold, that's it - there will never, *ever,* be another one offered again. Key chain hole for easy insertion on your key ring. Identify yourself at meetings, etc. as a *2600* member by dangling your key chain and saying nothing. Cover one hole and produce *exactly* 2600 hz. to beep-off a long distance call so you can then Multi Freq. another if your telephone office uses in-channel long distance equipment. Cover the other hole and you get another frequency. Use both holes to call your dog, dolphin, concubine, or hamster. Also, ideal for telephone remote control of your own electronic remote devices. Price includes mailing. $59.95. Not only a rare collector's item but a VERY USEFUL and unique device which is easy to carry with you at all times; nobody will ever know, except you, how it is used for remote control! Cash/money order only. Mail to: WHISTLE, P.O. Box 28992 (ST); CC, Missouri 63132.

BLUETOOTH SEARCH FOR ANDROID searches for nearby discoverable Bluetooth devices. Runs in background while you use other apps, recording devices' names, addresses, and signal strength, along with device type, services, and manufacturer. This is a valuable tool for anyone developing Bluetooth software, security auditors looking for potentially vulnerable devices, or anyone who's just curious about the Bluetooth devices in their midst. Exports device data to a CSV file for use in other programs, databases, etc. If you've used tools like btscanner, SpoofTooph, Harald Scan, or Bluelog on other platforms, you need Bluetooth Search on your Android device. More info and download @ http://tinyurl.com/btscan.

CLUB-MATE is now easy to get in the United States! The caffeinated German beverage is a huge hit at any hacker gathering. Now available in two quantities: $36.99 per 12 pack or $53.99 per 18 pack of half liter bottles plus shipping. Bulk discounts for hacker spaces are quite significant. Write to contact@club-mate.us or order directly from store.2600.com.

A TOOL TO TALK TO CHIPS. It's the middle of the night. You compile and program test code for what must be the 1000th time. Digging through the datasheets again, you wonder if the problem is in your code, a broken microcontroller... who knows? There are a million possibilities, and you've already tried everything twice. Imagine if you could take the frustration out of learning about a new chip. Type a few intuitive commands into the Bus Pirate's simple console interface. The Bus Pirate translates the commands into the correct signals, sends them to the chip, and the reply appears on the screen. No more worry about incorrect code and peripheral configuration, just pure development fun for only $30 including world wide shipping. Check out this open source project and more at DangerousPrototypes.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

Announcements

WHISTLEBLOWER EDWARD SNOWDEN is currently in Russia where he has been granted temporary asylum. The United States government is exerting substantial pressure on Russia and other countries in an attempt to force Mr. Snowden to the United States where he will face decades in prison or worse. Mr. Snowden's legal defense and its associated public campaign will be a long and expensive journey which will only be overcome with your financial help. Support the right to know. Support Edward Snowden. https://wikileaks.org/freesnowden Donation methods include online credit card or PayPal. Checks can be mailed to Derek Rothera & Company, Chartered Accountants, Units 15 & 16, 7 Wenlock Road, London N1 7SL, United Kingdom. Bitcoins can be sent to 1snowqQP5VmZgU47i5AWwz9fsgHQg94Fa.

Help Wanted

BE A TACTICAL TELEPHONE INSTALLER. Successful telephone service provider Shadytel is seeking applicants for the position of tactical lineman at ToorCamp 2014. Approximately four positions are available. Applicants must be able to attend Toorcamp, scheduled for 09 through 14 July 2014, at Neah Bay, Washington, USA. We will provide landline service to people in tents. You'll be responsible for helping to make that happen: taking orders, laying and terminating cable, configuring switchgear, tearing down the network afterwards, et cetera. On-site training will be provided. Qualified applicants will display a hobbyist interest in cable management, record keeping, and (optionally) customer service. Strong applicants will be able to recite the Shadytel Core Values and demonstrate

familiarity with telephone industry regulations. No monetary compensation is offered. You will receive some of our schwag, possibly to include: Shadytel polo shirt, Shadytel hardhat, Shadytel branded hand tools, other Shadytel branded items. Send email to careers@shady.tel to apply. Include relevant information. Also include a phone number and preferred times to talk (Seattle time or UTC).

I'M LOOKING FOR A GRAPHIC DESIGN ARTIST to help me design a "logo" /vehicle sign for a solar installation business I plan on doing in the near future. I can send a photocopy of a hand drawn rough draft. The logo will include log cabins, a stream, wind turbines, solar panels, and such. If you have any examples of your work, especially regarding anything like this, please send it my way with pricing info. This is for a small first time business owner who is planning for the future. I can pay a reasonable amount. Designs will have to be printed and mailed to me. The final can be sent to a home address on a flash drive. Payment will be made by BoP check. Please contact me at Solomon B. Kersey #87754-020, Federal Corrections Complex - Low, P.O. Box 5000, Yazoo City, MS 39194.

Wanted

INTRODUCING GSCSI - Global Strategic Cyber Studies Institute: We are a startup with solid senior leadership and a mission that calls for change to the current mentality regarding the negative connotations associated with the term "Hacker". We are all hackers in one way or another and we want to put forward and proudly carry the wisdom behind some incredibly talented individuals. In fact, we don't hire anyone who "doesn't get it". We need help to grow and develop a revenue stream, and are seeking volunteers for positions in curriculum development, instructional design, instructors (virtual and classroom). We also are looking for any interested candidates to serve on our Advisory Board. Also, if you are interested in public speaking at Cyber events and are willing to travel the globe, let us know. Please send any questions or expressions of interest to 2600team@gscsi.org. Please help us reshape the cyber world and thinking one mind at time, if need be.

Services

GET YOUR HAM RADIO LICENSE! KB6NU's "No-Nonsense" Study Guides make it easy to get your Technician Class or General Class amateur radio license. They clearly and succinctly explain the concepts, while at the same time give you the answers to all of the questions on the test. And the best part is that they are free from www.kb6nu.com/tech-manual. E-mail cwgeek@kb6nu.com for more information.

DIGITAL 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 digital 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 recover data from many sources, including computers, external media, and smartphones. 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's digital forensic examiners all hold prestigious forensic certifications. Our principals are co-authors of *The Electronic Evidence Handbook* (American Bar Association 2006) and of hundreds of articles on digital forensics and electronic evidence. They lecture throughout North America and have been interviewed by ABC, NBC, CBS, CNN, Reuters, many newspapers, and even Oprah Winfrey's *O* magazine. For more information, call us at 703-359-0700 or email us at sensei@senseient.com.

INTELLIGENT HACKERS UNIX SHELL: Reverse.Net is owned and operated by intelligent hackers. We believe every user has the right to online security and privacy. In today's hostile anti-hacker atmosphere, intelligent hackers require the need for a secure place to work, compile, and explore without big-brother looking over their shoulder. Hosted 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/

NOPAYCLASSIFIEDS.COM - Free advertising - 50 countries! Free business directory, classified ads (6 free photos) with link to your website to help you expand your business and improve search engine placement. Search over 35 million classified ads (mostly USA) to help you find what you want. Thank you for being part of our online audience!

SECURE UNIX SHELLS & HOSTING SINCE 1999. JEAH.NET is one of the oldest and most trusted for fast, stable shell accounts. We provide hundreds of vhost domains for IRC and email, the latest popular *nix programs, access to classic shell programs and compilers. JEAH.NET proudly hosts eggdrop, BNC, IRCD, and web sites w/SQL. *2600* readers' setup fees are always waived. BTW: FYNE.COM (our sister co.) adds free WHOIS privacy to all domains registered or transferred in!

BASEMENT TECHIE* AND *THE DYSTONAUT: Two great tastes that taste great together! Better than a kick in the ass with a steel toe boot! DIY - Dystopias - Poor Hackers playing with Electronics and RF - Living Outside The System - by Ticom - http://www.oberonsrest.net/

WANT SOMEONE'S FBI FILE? Check out GetGrandpasFBIfile.com, a site that shows you how to get the FBI files for any dead person. Or use GetMyFBIfile.com, the site that shows you how to get your own FBI file.

THOUSANDS OF GOVERNMENT DOCUMENTS are published at GovernmentAttic.org. New material available each week. Click on the Documents homepage link to start.

Personal

BORED COMPUTER HACKER locked up in Oregon prison. There's currently a book based on my past exploits called *@Large*. I am currently serving a concurrent state and federal sentence. I am disabled with no source of regular income. I am currently seeking any donations of books/magazines/cash. You can donate money via http://www.accesscorrections.com or jpay.com. I can be emailed via accesscorrections.com too. All books/magazines have to have a return address from a bookstore/Amazon, etc. I can also receive newspapers, brochures, catalogs, photos. Feel free to send me any manga from off the net or any interesting articles/pictures. I am interested in fantasy/scifi books and all areas of technology and computer security. You can snail mail me at the following address: SRCI, Timothy Bach, SID#18928639, 777 Stanton Blvd., Ontario, OR 97914.

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. If you're an electronic subscriber, please send us a copy of your subscription receipt. Send your ad to *2600* Marketplace, PO Box 99, Middle Island, NY 11953. You can also email your ads to subs@2600.com. **Deadline for Summer issue: 5/21/14.**

HOPE X

PREREGISTRATION IS OPEN!

Announcing the newest Hackers On Planet Earth conference, to be held at the recently rescued Hotel Pennsylvania in New York City, July 18, 19, and 20, 2014

Preregistration is easy! Just visit store.2600.com and order your tickets. You'll get an email confirmation and you'll be set. Thanks to the hard work of so many great people, we're able to continue to keep the price extremely low for a conference of this kind in the middle of one of the busiest places in the world. (Check x.hope.net for deals on hotel rooms for conference attendees.)

We will once again have in excess of 100 speakers and talks, break-out sessions, workshops, concerts, all sorts of villages (hackerspace, lockpicking, hardware hacking, and the like), Segway rides, art displays, contests, retro computing, and new things still being developed!

Interested in speaking? HOPE X wants to hear from you! Just email speakers@hope.net and let us know in a few paragraphs what you want to address, who you are, and other relevant info. Guidelines are at x.hope.net.

None of this would be possible without the hundreds of volunteers who pitch in to make it all happen. If you want to be a part of that, send an email to volunteers@hope.net and let us know if there's something specific you can do or if you're able to simply be sent where you're needed.

Finally, our biggest challenge as always remains getting the word out. We don't have a big PR team, just a magazine, radio show, website, and lots of friends. But we would be thrilled to have the word spread before the conference so that more new people get to experience this and not simply read all the amazing press we get after it's all over. If you can help, email press@hope.net and give us your ideas.

HOPE X is our 10th conference, the 20th anniversary of our first conference, and the 30th anniversary of *2600*! It's all lining up so perfectly. We hope to see you there.

xxx.xxxxxxxxxxxxxxxxxx.xxx (likely the coolest domain name EVER)

or x.hope.net (for those who can't or won't visit the .xxx domain)

NEWSFLASH! HOPE X is now accepting bitcoins! We're the first hacker conference in America to do this and one of the few conferences anywhere to have ever opened the doors to this innovative form of digital currency. Visit bitcoin.hope.net to be a part of this.

"Everybody gets so much information all day long that they lose their common sense."
- Gertude Stein, 1946

Editor-In-Chief
Emmanuel Goldstein

Associate Editor
Bob Hardy

Layout and Design
Skram

Cover
Dabu Ch'wald

Office Manager
Tampruf

STAFF

Infrastructure
flyko

Network Operations
phiber

Broadcast Coordinator
Juintz

IRC Admins
beave, koz, r0d3nt

Inspirational Music: GreenJolly, Juno Reactor, Lorde, The Kleptones, Elzhi, ODB, Shame On Blue

Shout Outs: Henry Hickmen, Lunar Freeze, Evan Doorbell, David Kennedy, John Huntington, Blackphone, SpoofCard

Big Welcome: Olivia

***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 2014, Volume 31 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. & Canada - $27 individual,
$50 corporate (U.S. Funds)
Overseas - $38 individual, $65 corporate

BACK ISSUES:
1984-1999 are $25 per year when available.
Individual issues for 1988-1999
are $6.25 each when available.
2000-2013 are $27 per year or $6.95 each.
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**

MEETINGS

ARGENTINA

Buenos Aires: Bar El Sitio, Av de Mayo 1354.

AUSTRALIA

Melbourne: Southgate Shopping Complex, outside food courts.
Sydney: The Crystal Palace Hotel, 789 George St. 6 pm

AUSTRIA

Graz: Cafe Haltestelle on Jakominiplatz.

BELGIUM

Antwerp: Central Station, top of the stairs in the main hall. 7 pm

BRAZIL

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

CANADA

Alberta

Calgary: Food court of Eau Claire Market. 6 pm
Edmonton: Elephant & Castle Pub, 10314 Whyte Ave, near big red telephone box. 6 pm

British Columbia

Kamloops: Student St in Old Main in front of Tim Horton's, TRU campus.
Vancouver (Surrey): Central City Shopping Centre food court by Orange Julius.

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

CHINA

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

COSTA RICA

Heredia: Food court, Paseo de las Flores Mall.

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.
Grenoble: EVE performance hall on the campus of Saint Martin d'Heres. 6 pm
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: Bar le Golden Gate, Rue St Georges a Rennes. 8 pm
Rouen: Place de la Cathedrale, benches to the right. 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 payphones beside the Dublin Tourism Information Centre on Suffolk St. 7 pm

ISRAEL

***Beit Shemesh:** In the big Fashion Mall (across from train station), second floor, food court.

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

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
Trujillo: Starbucks, Mall Aventura Plaza. 6 pm

PHILIPPINES

Quezon City: Chocolate Kiss ground floor, Bahay ng Alumni, University of the Philippines Diliman. 4 pm

SWEDEN

Stockholm: Starbucks at Stockholm Central Station.

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. 6 pm

Arizona

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

Arkansas

Ft. Smith: River City Deli at 7320 Rogers Ave. 6 pm

California

Los Angeles: Union Station, inside main entrance (Alameda St side) between Union Bagel and the Traxx Bar.
Monterey: East Village Coffee Lounge. 5:30 pm
Sacramento: Hacker Lab, 1715 I St.
San Diego: Regents Pizza, 4150 Regents Park Row #170.
San Francisco: 4 Embarcadero Center near street level fountains. 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

Loveland: Starbucks at Centerra (next to Bonefish Grill). 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

Fort Lauderdale: Undergrounds Coffeehaus, 3020 N Federal Hwy. 7 pm
Gainesville: In the back of the University of Florida's Reitz Union food court. 6 pm
Jacksonville: O'Brothers Irish Pub, 1521 Margaret St. 6:30 pm
Melbourne: Trephub, 907 E Strawbridge, #103. 5:30 pm
Sebring: Lakeshore Mall food court, next to payphones. 6 pm
Titusville: Krystal Hamburgers, 2914 S Washington Ave (US-1).

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: Tomlinson Tap Room in City Market, 222 E Market St. 6 pm

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
Worcester: TESLA space - 97D Webster St.

Michigan

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

Minnesota

Bloomington: Mall of America food court in front of Burger King. 6 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

Elko: Uber Games and Technology, 1071 Idaho St. 6 pm
Reno: Barnes & Noble Starbucks 5555 S. Virginia St.

New Hampshire

Keene: Local Burger, 82 Main St. 7 pm

New Mexico

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

New York

Albany: SUNY Albany Transfer & Commuter Lounge, first floor, Campus Center. 6 pm
New York: Citigroup Center, in the lobby, 153 E 53rd St, between Lexington & 3rd.
Rochester: Interlock Rochester, 1115 E Main St, Door #7, Suite 200. 7 pm

North Carolina

Charlotte: Panera Bread, 9321 JW Clay Blvd (near UNC Charlotte). 6:30 pm
Greensboro: Caribou Coffee, 3109 Northline Ave (Friendly Center).
Raleigh: Royal Bean Coffee Shop, 3801 Hillsborough St (next to the Playmakers Sports Bar and across from Meredith College). 7 pm

North Dakota

Fargo: West Acres Mall food court.

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.
Youngstown (Niles): Panara Bread, 5675 Youngstown Warren Rd.

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, food court outside Taco Bell.
Pittsburgh: Tazz D'Oro, 1125 North Highland Ave at round table by front window.
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 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 (Plano): Fourteen Eighteen Coffeehouse, 1418 Ave K. 6 pm
Houston: Ninfa's Express seating area, Galleria IV. 6 pm

Vermont

Burlington: The Burlington Town Center Mall food court under the stairs.

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
Richmond: Hack.RVA 1600 Roseneath Rd. 6 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 (a * indicates a meeting that's held on the first Thursday of the month). Unless otherwise noted, *2600* meetings begin at 5 pm local time. To start a meeting in your city, send email to meetings@2600.com.

Payphones of the World

Czech Republic. This is a fairly basic model found in Prague. It's definitely seen a good amount of use but looks like it can handle quite a bit more.

Photo by Matt Anderson

Mexico. Found in an underground washroom hall in Playa del Carmen, this phone clearly benefits from spending all of its time indoors.

Photo by Jorge

Peru. This is a name we should all become familiar with. A subsidiary of América Móvil, a Mexican company, Claro Americas can be found in just about every Central and South American country, plus the Caribbean. Certainly among the most cheerful looking phones out there.

Photo by Leonel H. Ramos Chang

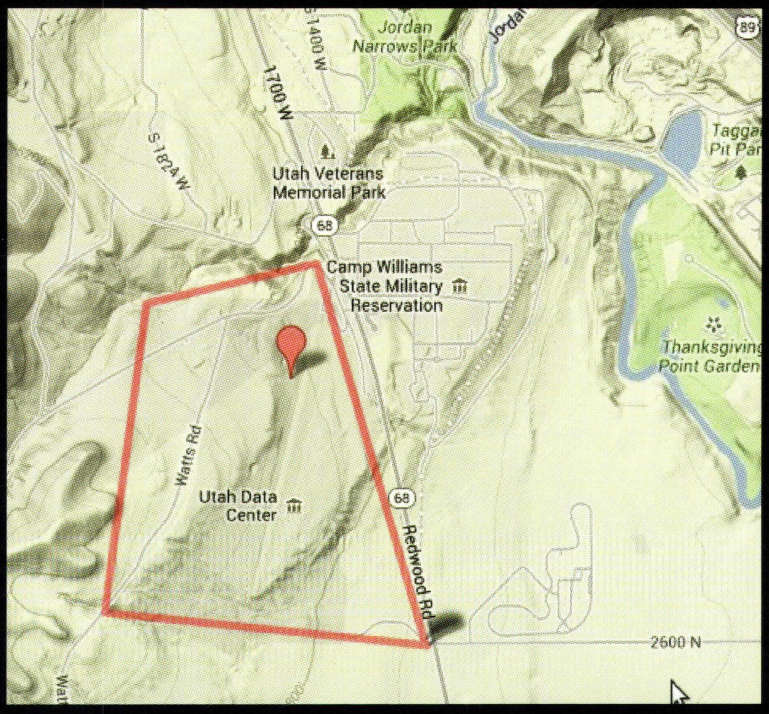
Ecuador. A decidedly less cheerful model, but Claro is still the lead operator in this country with almost nine million subscribers. América Móvil ran the company when it was known as Porta, but switched the name to Claro, which translates to "bright" or "clear" in Spanish.

Photo by TProphet

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

On the web, "404" and "missing" are pretty much synonymous, which means anyone halfway familiar with the net will be reading this sign as "Missing Hair Design" every time they walk past this place in Edinburgh, Scotland. Discovered by **sarx**, this is probably not the best marketing strategy for this establishment, even if it happens to be their address.



Reader **Steve** found a rather weird fact while Googling Bluffdale, Utah. It seems the massive NSA data center over there just so happens to have a road named "2600" heading straight to it. We have to assume this is a coded invitation.

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.