Volume Thirty-Five Number One!
DIGITAL EDITION Spring 2018

2600

The Hacker Quarterly

African and Asian Payphones

Morocco. Found outside of an Afriquia gas station in Errachidia, this is the standard type of payphone seen throughout the country, though not many are on the outside of buildings like this one.

Photo by Gabriel Dean

Taiwan. This phone was discovered inside the Chung-Shan building (which can be seen on the back of the 100 New Taiwan Dollar bill and was the venue of the National Assembly) which wasn't open to the public until recently.

Photo by John Skilbeck

Saint Helena. Seen in the capital city of Jamestown, this is a fairly basic model with a somehow otherworldly feel to it, much like the country itself.

Photo by Babu Mengelepouti

Indonesia. This apparently long-forgotten phone can be found by the Gelora Bung Karno Stadium in Jakarta. A real fixer-upper.

Photo by Michael McPhail

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 (Do not send us links as photos must be previously unpublished.)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)

COMMUNIQUES

inject your soul with liberty

Embracing Empowerment

It was inevitable.

Whenever people are pushed, victimized, or stifled, there always arrives a time for backlash. It could come quickly or it could take years, even generations. But it always happens at some point. It's who we are as humans. And when that opportunity to fight back comes, it's almost impossible to restore the status quo. This is the natural order of things.

There is little more inspiring to us than to see those who were once subjugated to powerlessness step forward with a renewed spirit in front of a populace willing to listen. Whether it be the people of an oppressed nation, an impassioned group of idealistic high school students, or the victims of abuse, these newly found voices need to be celebrated and encouraged by all of us.

In the hacker world, we generally consider ourselves to be open to the views of everyone. For decades, we've said that "we exist without skin color, without nationality, without religious bias" (as stated so eloquently in the "The Hacker Manifesto"). We defend the right of all to speak, no matter how distasteful the words may be. Anonymity is our friend, allowing all to put forth an opinion without fear of being exposed and held accountable. And these values eventually found their way into the mainstream and onto the net, where they more or less became the default attitude. As the famous cartoon in *The New Yorker* said, "On the Internet, nobody knows you're a dog."

But clearly, this is an idealistic view that's only partly true. While theoretically we're all equals and we're judged only by the words that come from our keyboards, in reality the same ugliness that pervades society trickles down into our community as well. We've always known this.

One has only to pore through old IRC logs, Usenet posts, or BBS archives to quickly find examples of sexist, homophobic, xenophobic, and racist dialogue, things we all simply accepted as normal without really considering how crippling these attitudes have been to others. It's only now, as the people on the other side of these words come forward, that many realize how wrong it was to just tolerate this or to believe that it was all in good fun and there were no real victims. There were. There are. And, as long as we allow it to continue, there always will be.

By witnessing the extent of the #MeToo movement, everyone can realize how pervasive sexual abuse has been in our culture. No institution has escaped this. Religion, politics, entertainment, corporate boardrooms, sports, media, and yes, tech. Yes, even the hacker community. When the sickness is this profound, there are very few places where you can find refuge from it.

So what's changed? Why do we know better now? Primarily, it's because of the empowerment that has finally been realized by those who were taught to have no hope of things ever being different. They have raised their voices, banded together, and forced the rest of us to confront the ugliness and finally start doing something about it. Sometimes, all it takes is the courage of a single person to open up the floodgates. Other times, it's a prolonged effort that's resisted at every turn. But once it begins in earnest, it really can't be stopped. And if this makes you uneasy, you should ask yourself why.

Change is always difficult, even when it's essential. Something as innocent as moving from one school to another can seem impossible to accomplish, and as children we might resist vehemently. The more serious changes - granting equal rights to the oppressed, accepting different and changing

cultures, acknowledging one's mistakes on both an individual and collective level - are almost always resisted to some degree, even by good people who ultimately know better. When the tipping point comes - and the tipping point always comes - it seems unconscionable that we ever allowed such an oppressive environment to exist in the first place. It becomes easy to see how wrong we were and we all rush to make judgments since we now live in a more enlightened era. And that's where the cycle begins all over again. We don't see *ourselves* as part of the problem. It's always someone else.

Once we've come to terms and realized that we have in fact been on the wrong side of history, it's very easy to do the right thing. It's so much harder to apply that moving forward, to consider that we may *still* be acting in an unjust and unfair manner. The way we tackle this is by listening - and by never assuming a challenge to one's beliefs is a threat.

It takes a tremendous amount of courage to come forward and confront an injustice, whether it's on a personal or a systemic level. The very fact that we saw so many people doing this in recent months is a clear illustration of just how intimidating the entire process is. Imagine having to live with the secret of being abused for years or decades because the entire process of seeking justice would likely backfire and cause you even more pain. We see how many people this has affected, but we're only seeing the cases that affect people already in the limelight. How pervasive is this and how many more ordinary, non-famous people are also victims? It's hard to even fathom.

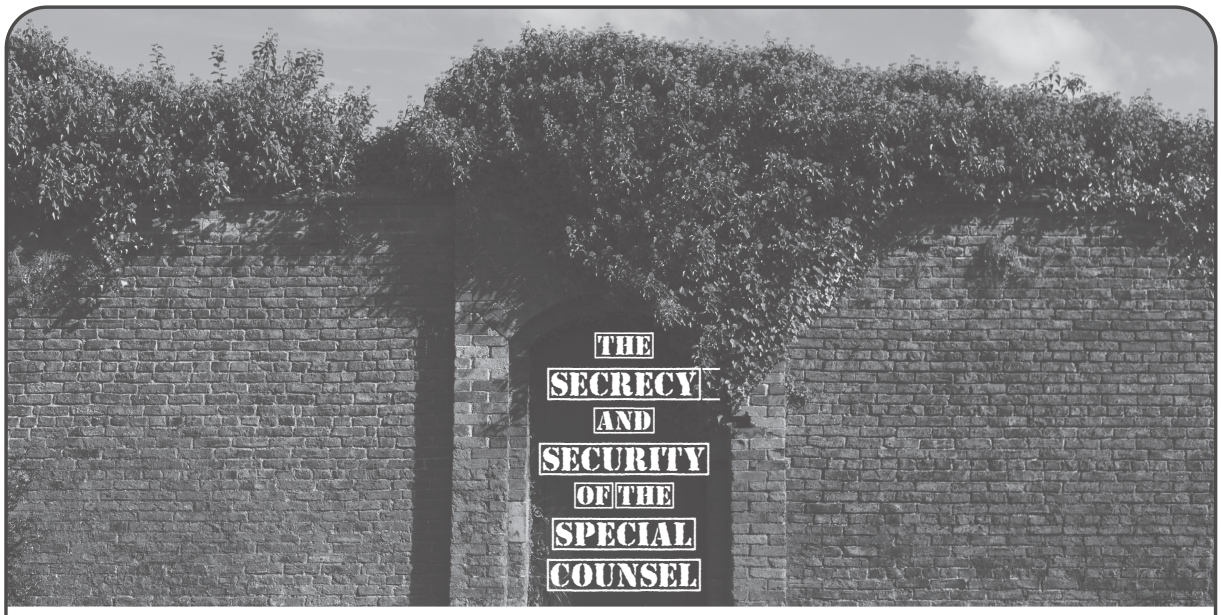
Of course, any system can be abused and, now that the voiceless are finally being heard, we expect a wave of opportunists to sweep in, using this forum to settle scores or profit in some way. This is not something to be feared, as long as we don't fall into the same trap of fact-free judgment we're so often drawn to. We know there will be mistakes and injustices. But this is inevitable in any scenario; not confronting the demons amongst us is simply not an option anymore.

The hacker community has always been particularly thoughtful and introspective. That is why we are well suited to look critically at ourselves and figure out how best to make improvements. We must avoid the arrogance of knowing we're right and remain open to the possibility of being completely wrong. Much like we handle technology itself, we need to always consider new ways of approaching old issues, embracing the modern without rejecting the history. Change is essential, but it's not always correct. We need to experiment, try new methods to solve a problem, admit our mistakes, and share our results. It's only when we dig our heels in and refuse to consider other ideas that we begin to stagnate. And that is the first step downwards.

Every time we see someone stand up to the system, whether it's a kid in school, someone who's handicapped, a victim of bias, or a challenger of abuse, that is a moment to cheer and to strongly support. Many times, all that the rest of us need do is listen. Take the person seriously. Respect where they are coming from. We're often so used to getting this ourselves that we forget how many others have to fight to be heard. And what they have to say could change everything.

But most importantly, we have to avoid becoming the people who are the problem. We do that through communication and by setting examples. It's not too late even if you added to the problem in the past. Understanding how people go down a bad road is key to not making similar mistakes in the future.

There is a reason kids today are so perceptive and much more morally conscious with regard to social issues, the environment, alternative cultures, and using their voices. They've seen the results of getting it all wrong... greed, pollution, prejudice, bullying. They've witnessed the abhorrent behavior of those who want those "good old days" to return, and they know how to use social networks to build movements to fight back. Sure, you will find exceptions to this and many will argue that the youth of today is as close-minded and self-centered as ever. But we believe there is a measurable change in the air, brought about by a growing amount of frustration, anger, and support. As hackers and those who build and rebuild technology, this is something we've seen before. When we're pushed so far in the wrong direction, the inevitable pushback momentum takes us much further in the right direction than if we had simply gotten there on our own. In that way, the proponents of injustice and abuse have been key motivators in our making the world a better place. All we have to do is react.



by Alexander Urbelis
f/k/a Neon Samurai
alex@blackstone-law.com

Within the hacker community there is healthy disagreement about most everything. But one topic on which we can all agree is that keeping information secure is hard work. The more sophisticated the adversary, the harder this work becomes. And the more time an adversary has to mount and launch an attack, the more likely it is that an attacker will identify a vulnerability ripe for exploitation.

After nearly a year of clandestine yet intensive investigative work, Special Counsel Robert Mueller has been steadily unraveling the information warfare campaign waged by Russian agents to influence the outcome of the 2016 U.S. presidential election. Mueller's work has confirmed three critical points of this information operation: (i) Russians sought to sow political discord within the United States by exploiting Americans' addiction to social media (and the pleasure inducing algorithms that have kept us addicted to social media), (ii) Russians targeted not only the Democratic National Committee but also State voter rolls and voter registration databases, and (iii) the objectives of these operations were to promote the candidacy of Donald Trump and discredit Hillary Clinton. Carefully and methodically plodding along day in and day out, Mueller and team continue to make astonishing progress in ways no political pundit could have predicted. Mueller's indictment of 13 Russian nationals and three Russian entities for their role in the information operations above relied on primary source material - including email communications to and from indicted Russian nationals - kept closely and securely guarded during the investigation, and the indictment itself told an intricate tale of intrigue and skullduggery ranging from operations carried out from nondescript office buildings in St. Petersburg to Russian operatives' visits to at least ten U.S. states.

Add to this the indictment and surrender of Paul Manafort and Richard Gates, and the arrests, collaboration, and guilty pleas of Michael Flynn and George Papadopoulos, and one begins to wonder: how did Mueller accomplish this despite being arguably the single most targeted person on this planet when it comes to information security threats? Classic threat modeling - the process of enumerating and prioritizing hypothetical threats - would confirm this: as the special counsel charged with spearheading the investigation of the President of the United States' possible collusion with Russian operatives during the 2016 election and a criminal probe into the issue of whether the President or his confidantes engaged in the obstruction of justice, he faces the most sophisticated of adversaries on a daily basis.

Mueller's adversaries range from the experienced intelligence operatives of the Russian GRU, Pakistani ISI, Israeli Mossad, German BND, to the nearly 70 other active intelligence agencies worldwide, to the world's media organizations, and to the President himself, all of which actively seek even a bread crumb of data that could give insight into Mueller's targets, plans, and findings. We expect Mueller

and his team to operate securely, without unnecessary leaks, and to prevent these sophisticated prying eyes from compromising his investigation. (To be sure, the American people and media would crucify Mueller and his team if a foreign power compromised their security protocols and, say, their internal communications ended up in a data dump to Wikileaks.) When it comes to the operational security of Mueller and his team, there is no room for error. A single misstep could compromise the entire investigation. And the vectors of attacks are so numerous and so inherent to the technology, that we all take for granted on a daily basis that Mueller and team must forego nearly all forms of modern collaboration.

Unpacking a small subset of the vulnerabilities and threats associated with everyday technology is illuminating in and of itself, explains why Mueller's investigation must exist in a black box, and demonstrates why we should expect it to continue for some time.

Email

Mueller and his team should never *ever* be sending emails across networks to each other. As pervasive as email is in our lives, it is one of the least secure methods of communication. Email travels across networks in plain text, which means that anyone who has access to that network could arguably "sniff" out the data packets and reassemble the entire communication.

Moreover, email passes through or resides on a mail server. That means that these mail servers themselves present another vector for compromise. And the same goes for any device on which email resides, whether it be a computer, phone, or tablet. There is no doubt that Mueller's adversaries are lying in wait for misaddressed confidential emails sent to, for example, an auto-inserted email address associated with a team member's employer. In short, the use of email for anything related to the special counsel investigation would be tantamount to a breach or leak.

Encrypted Email

A reasonable assumption would be for Mueller to rely on encryption technology for sending confidential emails, but this too would fail when dealing with the sophisticated adversaries Muller and team face.

Encrypted email has its shortcomings. For example, only the body of a message is encrypted, leaving critical metadata such as the sender, recipient, subject line, and *header* information exposed. In addition, it is very easy to make a mistake and forget to encrypt an email attachment when using encryption tools. A single unencrypted email attachment sent across a public network could be disastrous for the investigation.

Further still, decrypting emails requires the protection of one's *private key*, which is generally stored on one's computer. If one's computer - or one's network - is not secure, then using encrypted email is utterly futile.

SMS, Text Messages, and 2FA Vulnerabilities

Surely, Mueller and team should be able to send quiet bursts of text to each other via SMS or text message? This is certainly *not* the case. SMS and text messages are only as secure as the telecommunications providers' networks over which they are relayed. SMS messages have been subject to, among other vulnerabilities, social engineering attacks, whereby an attacker convinces a wireless carrier to redirect SMS traffic to a separate SIM card.

Similarly, phone number hijacking scams that rely primarily on an attacker's ability to social engineer the port of a telephone number from one carrier or SIM card to another would effectively compromise Mueller's SMS and text message communications. If successful, an attacker would have access to 2FA authentication codes sent to his mobile, and the ability to perform password resets for any and all accounts tied to the ported phone number.

End-to-End Encrypted Messaging

There has been a great deal of talk about end-to-end encryption being implemented in apps such as WhatsApp and Signal. And it is true that end-to-end encryption prevents eavesdropping on messages while in transit, but this type of encryption does not protect a message once it resides on a device. If an end point, e.g., a phone or computer, can be compromised, then end-to-end encryption is useless.

The limits of end-to-end encryption were seen all too well in the aftermath of the failed Turkish coup, whose organizers relied on Whatsapp for planning, and whose arrest and interrogation required them to hand over their phones and passwords, thereby allowing the

authorities to access unencrypted Whatsapp conversations.

Phone Calls

Even using a mobile phone to discuss the special counsel investigation could result in a compromise. A pervasive vulnerability in what is known as Signaling System No. 7 (SS7) enables attackers with access to an SS7 system to acquire nearly all data from a mobile phone. This means that a mobile user's calls can be forwarded through an intermediary and recorded, SMS messages intercepted, and the location of a mobile tracked so long as it is powered on. And while the SS7 vulnerability affects nearly everyone with a mobile phone, it is an attack vector reserved for extremely high-value governmental targets, a category into which Mueller and team undoubtedly fall.

Zero-Day Exploits

A vulnerability that is known only to an attacker and can be exploited at will is called a zero-day exploit. By definition, a zero-day exploit utilized against Mueller or his team's devices could compromise the entire investigation, especially if the team was operating on the erroneous assumption that email or text message encryption was effective security, or if any systems warehousing communications of the investigation were connected to the Internet.

Zero-day exploits are highly sought after and guarded secrets, fetching as much as 1.5M USD for certain types of exploits targeting iPhones. It would be difficult to think of a target worthier of a zero-day attack than Mueller and his team. Thus, it is imperative that for the operational security of the special counsel investigation that no data whatsoever from the investigation can ever reside on any devices used to access the Internet.

Home Networks

Nothing today is off limits when it comes to breaches. Mueller and team must be extremely cautious with their data and actions on any network to which they are connected, especially home networks using consumer grade equipment. The failure to secure a router or upgrade firmware presents a simple attack vector that could result in the breach of communications, web browsing habits, and personal files. Home networks are fertile ground for adversaries looking for information to compromise or blackmail Mueller or his team. And, as was widely reported and commonly misunderstood, the compromise of the WPA2 Wi-Fi protocol (used to secure nearly all WiFi networks) suggests that targeted attacks against unpatched high-value governmental targets will be forthcoming.

* * *

What this means for Mueller and team is that they cannot rely on the technology we take for granted on a daily basis, because a single misdirected email, text message, or unsecured phone call could compromise many months of arduous investigative work. The full body of the work of Mueller and team must reside only in what is known as a Sensitive Compartmented Information Facility (SCIF). SCIFs are essentially information vaults, access to which is highly controlled. The internal networks on which Mueller and team communicate should be air-gapped, meaning the network should not be accessible to the outside world by any means. No data from the investigation should ever reside on a device that accesses the Internet, electronic means of communications should be kept at a minimum, and most of the work and collaboration must be anachronistically accomplished by face-to-face meetings and discussions. Thankfully, given that a great deal of Mueller's team is drawn from the ranks of the FBI and DOJ, most should be familiar with the travails of working within the confines of a SCIF. To date, there can be no question that Mueller and his team have run a surprisingly tight ship in terms of operational security, and their practices should be viewed as an exemplar of what it takes to keep critical information out of the hands of adversaries.

What this means for us, the public and the media, is that if we expect Mueller and team to continue to give the security of their information the attention it deserves, then we must become accustomed to a slow but steady show of progress. When what is at stake is the legitimacy of a sitting President of the United States, a counterintelligence investigation into that President's election tactics, and the unraveling of a hostile foreign nation's attacks on the most critical safeguard of our democracy - the vote - we are well to be reminded that patience is a virtue.

by RAMGarden

You see this question asked almost daily from those who want to learn for fun, get a job, or just learn more about hacking with software and writing their own tools and scripts. I taught myself how to code Applesoft BASIC when I was 12. I just happened to find a guide book in our tiny elementary school library one day. I asked the teacher during computer lab if I could try some of the examples in the book using one of the Apple IIe machines (that's all we had in our school back then). Once I made the computer ask me for my name then say "Hello <name>", I was hooked! I wish the valuable resources and tutorials available on the Internet now were around back then! Today, you can go to `www.w3schools.com` and easily learn Javascript, for example.

But instead of me telling you which language is best for a beginner or which is best for hacking, I'll give you my best advice from my two decades of programming for fun and profit: learn the basics that most languages have in common. Then, learning a different language is just looking up how those pieces should be typed out (or what that language *syntax* is for that particular piece). You'll also need to determine which language works best for the task, such as which ones work for the server side of a web app or which ones can run on mobile phones versus desktops natively, etc. But that's usually figured out with a quick Internet search.

These basic parts will make up most of your code and should be what you think of when designing your program before writing any real code. A great way to design is to write down pseudocode - code parts written in plain English that just lay out the basic statements and logic flow without any real, working code. Then you should be able to pick from a few different languages to write the program following that pseudocode. There's a ton of stuff I could go into for all the documents and planning and design reviews that go on with professional software engineering, but the most attention always seems to go to the pseudocode section of our documentation. There's also things that narrow down the list of languages to use, like if you will be extending other software using an API or application programming interface. Then you have to use whatever language is supported by that API. The documentation for the API will tell you which language or languages can be used. The Esri ArcGIS API supports Java, C#, VB.net, and python, for example.

Once you learn the basic parts and the things most languages have in common, learning a new language to take on a new programming project should be easier than just learning from scratch. You also don't have to memorize it all since you can just do quick Internet searches for something like "c# for loop syntax" or "VB.net if statement syntax" to get quick examples.

Just a few of the main basic parts are:

Variables

Store and retrieve numbers, words, whole sentences, objects, and more. You see these in algebra class and you see them a whole lot more in programming.

Examples:

```
catTax = 6.8
sumTotal = 42
grandTotal = sumTotal * catTax
```

The Ada 95 language uses this syntax (colon before the equal sign):

```
catTax := 6.8;
```

Look up Assignment (computer science) to learn more: `https://en.m.wikipedia.org/wiki/Assignment_(computer_science)`

If Statements

This is also known as if-then or if-then-else or boolean logic flow control. These are

the "choose-your-own-adventure" novels in computer form that will make up a large portion of your code. They are based around a boolean expression that just comes down to TRUE or FALSE. Things like "is this checkbox checked?", "did the user click yes or no?", "is the total greater than some maximum number?" will be asked with these "if statements". Note that every language has some way to mark the lines of code that will be executed if the statement is true. Sometimes it's curly braces, sometimes it's the words "end if", and for Python it goes by the fact that the lines "inside" the if statement are indented below it.

Examples:
Javascript, C#, Java, PHP syntax:

```
if (livesRemaining == 0) {
    alert("game over";
}
else {
    alert ("Ok!");
}
```

Python syntax (I had to google this with "python if statement"!):

```
if livesRemaining == 0:
    print "game over"
else:
    print "ok!"
```

Visual Basic .net:

```
If (livesRemaining == 0) Then
    Console.Write("game over")
Else
    Console.Write("ok!")
End If
```

Loops

Another feature that makes computers so great is how they can do the same or similar thing over and over again millions of times without getting bored or complaining. You can use various types of loops to tell the computer to repeat a whole section of code with as little or as much complication as you want. Watch out for the dreaded "infinite loop" though! If you tell it to repeat until something is true, but that thing never becomes true, then it will get stuck looping forever until you turn off the computer or kill the program. Sometimes apps or games can "hang" because they're waiting forever for something and that thing never happens. There are several kinds of loops like FOR loops, WHILE loops, and FOR EACH loops.

FOR loops are normally used to loop through a known number of items in an array or list of things like numbers.

Example: loop through all the cats in my list and print them out:

```
FOR (i=0; i < catList.count; i++)
{ print catList[i]; }
```

WHILE loops are used to loop until some condition becomes false. This is the one that can repeat forever!

Example: loop until I have 100 in my cat count:

```
WHILE(catCount < 100) { catCount
= catCount + 1; }
```

FOREACH loops are used to loop through all the items in a list and is only implemented in certain languages. This is different from the FOR loop in that instead of keeping up with some index then using that inside the loop to get the item from the array or list at that index, the FOREACH statement will assign each item in the list to the given variable for you so you can read or change the item directly and type less code.

Example: print out each cat's name in a list of cat names:

```
FOREACH (string catName in
catNameList) { print catName; }
```

Each time the loop goes through, it will change out the value stored in catName to the next one automatically. Less code to write!

These are just some of the basic parts of software and by no means anywhere near the full list, as that is covered in many programming books and API help documentation. I would recommend finding your nearest hackerspace/makerspace and asking if they have any code jams or programming classes to learn from. These are normally free or very low cost - along the lines of a small donation to the space to help cover their costs for rent, etc. If there aren't any of these near you, then I would say that going through any programming tutorials you can find on the Internet is always another great way to start. And don't forget to sign up for `stackoverflow.com` to post and answer questions from fellow programmers who get stuck making computers do great and wonderful things! I wish you good luck and hope you learn to bend silicon to your will.

Breaking Standards

by bartitsu59

Greetings from France. I think I've always wanted to write this article. Maybe because I'm very tall (6 feet 7 inches), and got mocked about that, or maybe because I grew up more and more as a hacker with a non-traditional view of the world.

From early on, I got pissed when everyone, back in the 90s, embraced a flawed machine called the PC, leaving behind them more beautifully engineered machines (like the Atari ST, the Commodore Amiga, or The Acorn Risc Machine).

Now, even with the Mac back in the landscape (which shows anyway a very similar hardware), we are bound by a number of standards without even thinking about it.

A Bit of Historical Context

Maybe you don't know about it, but in 2014, Carnegie Mellon University spent several months to extract the data and retro engineer the original drawing software from 20-year-old Amiga floppy disks. On those disks were exclusive digital drawings from Andy Warhol.

This story gave me a lot of insights. Not only was it quite funny and interesting to read about, but it made me realize also that using old technology, or at least a non-standard one, was a good way to conceal your data. After all, most of us prudent hackers have USB keys with, at the bare minimum, encrypted files, or VM images. Yet, putting someone else's USB key in any modern computer will reveal the nature of the data stored, with the file extension plus a beautiful icon.

For the most concealed files, say, one with no extension, a quick look at the first bytes (the signature of the file) would leave not much doubt on what kind of data is in there.

If the nature of the file is still unknown to you, then you can rely on several types of forensics software, for example Apache Tika, which will happily identify a thousand file formats. But it will fail with some....

How many of us know the Magic Shadow Archiver format, used on exotic OSes for the now vintage Atari line of computers? It's a practical format used for archiving Atari ST floppy images, supported by most Atari ST emulators.

Those fun facts can lead us to creative solutions to hide our data.

Out of Sight, Out of Mind

Imagine now that you write your password vault on an emulated ST, in a simple text file, and store it on an emulated floppy disk. Ideally, you would have chosen a compressed MSA floppy image (with .MSC extension, or no extension to complicate things).

Another option: I have on my desk a beautiful gray box called a "MiST Computer." This beast has an FPGA inside and can dynamically adapt its hardware behavior, depending on the selection of a soft core (programmed using a Hardware Description Language) copied on an SD card. Basically, I can switch from a 100 percent accurate hardware replica of an Atari ST, to an Amiga, an Amstrad, a Spectrum (see the link at the end of the article for the numerous possibilities)....

So, I could also store my password vault into a disk partition of this little machine, which is furthermore offline, and shares the same screen of my regular computer (which makes it easily available).

Alternatively, the floppy image trick will work as well, since these retro-modern machines do support those images as well.

With no physical access to the network, and an uncommon file structure on the machine's mass medium (in my case, an SD card), I don't see how one could get his hands on my sensible data.

Don't expect to be limited by the apparent lack of power of those options. Back in the days, those machines were able to cope with mundane tasks (like playing, writing, drawing, budgeting) with a few MHz and Mbytes. First, because most applications were written in assembly, but also because the operating systems were far simpler than the ones we are used to today, thus leaving all the horsepower for the user land. (Most were stored in ROM, so the RAM was left untouched.)

Of course, this applies to any decent emulator or FPGA-based computer. A nice example of an emulator, in fact a true virtual machine (that makes use of the full power of the host machine) is ARANYM (Atari

Running On Any Machine). Amiga forever and RPCEmu are also very good picks that will offer you tons of options to store your data, with the ease of use of any regular PC virtual machine.

In any case, you are strongly advised to use a hard disk image. Saving your data in this image will obfuscate it quite a bit.

If you are a PC addict and don't want to learn about alternative architectures (poor you), you could still rely on long forgotten file formats, using no-longer-supported old softwares, such as Wordstar, Ashton-Tate Framework, or DataPerfect. To use them, I would suggest downloading DOSBox, a very accurate DOS emulator.

Using Steganography

The icing on the cake consists of using steganography on an old image format.

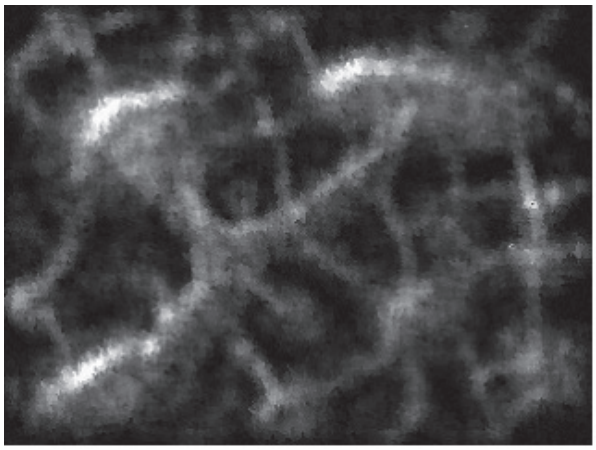
For this example, I will use an image format called "Degas Elite" (well-known Atari ST drawing software) in its extended flavor.

Indeed, Degas Elite was only able to handle the three standard resolutions of an Atari ST: low, medium, and high, for which the .PI1, .PI2, and .PI3 file extensions were respectively defined.

When more powerful Atari computers arrived on the market, some other drawing software (FuckPaint) extended the Degas file format to handle superior resolutions.

So, our image will be in PI9 format, with a resolution of 320x240 and a palette of 256. (You will see that this image is quite abstract, which is nice, since this technique will alter the palette.)

The technique I will use is kept ridiculously simple to just give you a primer on how steganography works. At the end of this article, you will find a link to a website describing a file format used by a fantasy console (pico8), on which I took inspiration.



The PI9 file format is really simple: you have the first 256*3 bytes describing the palette in RGB format. The rest of the file contains the bitmap uncompressed.

With only a few Unix commands, we will take a user:password couple, swap each couple of bytes (so that it does not appear in clear in a hex editor), and replace the first color declarations with it.

In our case - very simple with ten characters in total - we will then replace the four first colors with our data. Of course, the longer the data, the more the rendering will be altered. That's why I'm advising you to take an abstract scene, for which a change of colors will not be seen as suspicious.

For longer data to store, you need something more evolved, such as the technique used by pico8 and its special PNG format.

```
# First encode our user@password
➥ in hexadecimal, swap the byte of
➥ each 16bits word
# reverse the xxd command and write
➥ back to a 'header' temporary file
echo -n "2600@rules" | xxd -p |
➥ sed 's/\(.\)\(.\)/\2\1/g' |
➥ xxd -r -p > header
# A PI9 file has a constant size of
➥ 77824 bytes, our user@password
➥ couple is 10 characters long
# so write the whole source file minus
➥ ten bytes into a 'body' temp file
tail -c 77814 COLOURF.PI9 > body
# concatenate 'header' and 'body' to
➥ get the resulting image with the
➥ first4 colors altered (each color
➥ takes three bytes)
cat header body > COLOURB.PI9
```

I'm then using the online version of a tool called "recoil" to check that my image is: first, not corrupted; then, that it is properly shown with, at most, a minor impact on the palette. In our case, I'm seeing no difference between the original image and the new one.

With the same image, I was able to store three user:password couples for a total of 71 characters with no visible difference. This is explained by the fact that this image does not use the 24 first colors of the palette (24*3 color components = 72 bytes).

To retrieve the password, you proceed with a reverse approach:

```
$ head -c 10 COLOURB.PI9 | xxd -p |
➥ sed 's/\(.\)\(.
```

TELECOM INFORMER

by The Prophet

Hello, and greetings from the Central Office! Spring means that everything is blooming and the Pacific Northwest is even more green. Of course, I'm allergic to all of it. My nose turns into a faucet and I blow my nose with paper towels, not even bothering with Kleenex. Worse than all of that, though, are the cottonwood trees next door. While the trees are on the neighbor's side, the roots are on our side and they're absolutely relentless at breaking into our sewer line. Our toilets are backed up, the sinks are full, and there is a Porta Potty out in front of the building while the company decides whether indoor plumbing is actually required by our union contract. These days, nothing is done or fixed unless it's either required by law, contractually obligated, or will drive revenue. And whatever is fixed is done as cheaply as possible, after a long, slow, and deliberative process. I'm hoping by the time I write the fall column, we'll have running water again.

The problem of SS7 fraud is similar. It's an issue that, like the cottonwood trees, is well known. It's one that could have been prevented with some investment in maintenance. However, it's now a problem that is responsible for the seven calls (all of which were spoofed) that I have received so far today touting timeshares in exotic El Salvador. I wrote about the spoofed call problem in detail in the Winter 2017-2018 issue of *2600: The Hacker Quarterly*. If you didn't see it, the problem in a nutshell is that SS7 is (more or less) completely unauthenticated, so it's possible for anyone who has access to the network to claim that they're calling from any phone number they'd like to impersonate. What's more, even if I know that a call is totally bogus (for example, a call coming from an international gateway that claims to be a number assigned to my Central Office), I'm not allowed to block it because both policy and tariffs require me to deliver all calls. And this, after all, makes sense. Delivering calls usually means revenue to the company. Rejecting them means we'd not only have to spend money on recognizing and rejecting bogus calls, but we'd also lose out on the revenue.

When I wrote my last column, I didn't think the FCC would take any action that would stop robocalls. However, there has been a big change in the landscape: some debt collectors and IRS scammers started calling with spoofed numbers that pointed to Public Safety Answering Points (PSAPs). This, while possibly effective, was a major strategic miscalculation on the scammers' part. PSAP phone numbers are essentially a "back door" to 911. While the National Emergency Number Association (NENA) has been making efforts to lock down access to PSAP phone numbers (to the point where they charge $5,000 per year for access to a comprehensive database), a lot of these are publicly available. For example, one state publishes the addresses, phone numbers, and points of contact for every PSAP in the state.

Predictably, 911 operators are now being flooded with people returning missed IRS scam and other junk calls, which is now impairing the ability of public safety agencies to answer legitimate calls. There aren't many things that drive a hopelessly divided government to action, but failure of 911 services is one of them. The FCC issued a proposed order in November, and will vote in March. This order will allow phone companies to do the following:

- Block calls claiming to be from numbers that are not configured to place outgoing calls (such as PSAPs).
- Block calls claiming to be from phone numbers that are obviously bogus, such as disconnected numbers.

Naturally, this is tougher to implement than you might expect because, although they could be adapted, SS7 call flows weren't really designed for this use case. In fact, the whole telephone system is designed to *deliver* calls, not block them. It's possible to send calls through with missing or incorrect CN and CPN and, in fact, carriers are required to deliver all calls as long as the SS7 mandatory fields are valid. Not all fields are mandatory, though, and many fields are missing and invalid.

A few years ago, we actually got pretty close to fixing this before it all fell apart. Starting in the mid-2000s but reaching a fever pitch around 2010, rural wireline carriers got very interested in fixing one part of the problem: "phantom traffic." This is traffic that was intentionally obfuscated to avoid paying access charges. It got to the point where around 20 percent of calls delivered to rural, high-cost areas lacked the appropriate billing information.

This was done by providers using VoIP switches

that allowed SS7 fields to be modified en route. Obviously, this was an activity that was never contemplated by the original design of SS7. When a long distance call is placed, it is handed off from your local phone company to an interexchange carrier (typically your long distance company). If you're using a VoIP calling service, the process is essentially the same. However, interexchange carriers don't always route calls over *their* own network. Now that traffic is carried by VoIP on the back end, it can easily be routed using a "least cost routing" table.

What is the least cost routing for any call? When you're delivering the traffic as a local call and not paying access charges, of course! Unscrupulous carriers began modifying the CN (Charge Number) field at the time calls were delivered to the tandem closest to the destination, substituting a local number for the originating number. And like magic, there were no access charges!

Well, if you want to get the attention of phone companies, *mess with billing*. By 2008, lobbying by rural phone companies was intense. There was even a Congressional hearing. The issue reached a fever pitch in 2010, with loud protests from rural carriers who were being shorted. In the middle of all of this, Congress passed the Truth in Caller ID Act, which addressed spoofed and bogus calls (a different part of the SS7 problem).

With this much momentum, the FCC had a real opportunity to (mostly) phase out SS7, limit who could access the network, and transition the phone system to 21st century technology. Instead, they decided to muddle through. The existing networks remained in place and nothing got fixed, but the FCC issued an order requiring carriers to accurately report CN information and maintain it throughout the entire call path. And while there were high initial hopes, the Truth in Caller ID Act was impossible to build any real implementation rules around because of technical problems and loopholes in the law.

If you have been a longtime reader of this column, you probably remember that rural carriers could once profit handsomely by generating large volumes of incoming calls, which gave rise to free conference calling services, free voicemail, and other services operating - improbably - from small towns in rural states. This had been a thorn in the side of long distance companies for a long time, and although a two decades-long game of cat and mouse ensued (spanning complaints from rural carriers ranging from long distance carriers throttling and failing to complete calls to delivering phantom traffic), it became clear that revenue based on voice minutes was declining and no longer reliable. The FCC, in one sweeping order, rendered the whole issue moot. Access charges, a scheme in use since 1984, were to be phased out for large carriers by the middle of 2018, and for small carriers by the middle of 2020. The Universal Service Fund would be maintained, but funded in other ways and prioritized around the build-out of broadband services.

Unfortunately, the phase-out of access charges meant that there wasn't any real long-term incentive to improve the architecture of SS7; billing was only temporarily threatened, so it wasn't worth the investment. Carriers all over the country began applying for (and receiving) waivers from new CN delivery rules. In all fairness, older telephone switches don't support this; some parts of rural Alaska still don't even use SS7! However, the FCC also signed a consent order with Level 3, which was the largest offender in delivering phantom traffic.

Once again, the FCC is revisiting an issue for which the design and implementation of SS7 is the root cause, and once again there is a chance to make real improvements to the phone system. We'll see what the new rule looks like, and how carriers agree to implement it. Most are lobbying for a watered-down ruling that *allows* them to block bogus calls in the two specific categories referenced above, but doesn't require them to do so. If there is no requirement, then expect the phone companies to show up with a begging bowl and stories about hardship and difficulty in implementing the feature.

And with that, it's time to bring another column to a close. Have a wonderful spring, and I'll see you again in the summer!

References

- http://transition.fcc.gov/Daily_➥Releases/Daily_Business/2017/➥db0302/DOC-343731A1.pdf - Notice of proposed rulemaking from the FCC allowing "do-not-originate" and obviously bogus calls to be blocked
- https://apps.fcc.gov/edocs_➥public/attachmatch/FCC-11-➥161A1.pdf - FCC order switching to "bill and keep" from access charges and adding rules to address the phantom call problem
- https://www.utc.wa.gov/_layouts➥/15/CasesPublicWebsite/Get➥Document.ashx?docID=13&year=➥2011&docketNumber=110866 - Great presentation from the Washington Utilities and Transportation commission from 2011 on the phantom traffic problem, including network diagrams
- https://www.cs.rutgers.edu/➥~rmartin/teaching/fall04/cs552➥/readings/ss7.pdf - Very readable and understandable introduction to SS7
- https://www.gpo.gov/fdsys/pkg/➥CHRG-110shrg75047/html/CHRG-➥110shrg75047.htm - Congressional hearing on phantom call problem

The Censorship Resistant Internet
Part 2: How to Run an I2P Hidden Service

by p4bl0
2.6k@uzy.me

0x0 - Introduction

This is the second part of a series of articles explaining how to run censorship resistant services on the Internet. The first one, which was about the Tor technology, appeared in the Summer 2017 issue, and I assume here that the reader is familiar with it[1]. As the title says, this time I will talk about I2P[2], the Invisible Internet Project.

Tor and I2P are comparable technologies in that they are made to enable anonymous usage of the Internet. They differ in their threat-model and thus in their design. From a user's point of view, we can simplify things and say that Tor is better as a proxy (anonymously going out of the mixnet) while I2P is better at hidden services (anonymously staying inside the mixnet). Most other differences are more technical and I will discuss them as needed in the rest of the article.

To transmit data anonymously, I2P uses what is called "garlic routing," which is a loose name in reference to onion routing and to the fact that messages can be bundled (like cloves inside a garlic bulb). The principle of garlic routing is that potentially bundled messages are sent to their destination (a cryptographic key) by going through tunnels. Unlike Tor circuits, I2P tunnels are attached to a router and are unidirectional. A router is simply a running instance of an I2P implementation: I2P being fully decentralized (contrary to Tor which needs centralized directories (that I incorrectly called "distributed hash table" in the previous article, sorry!)), each client is also a router and participates to the network. In addition to the participation to other routers' tunnels, each router is responsible for the creation of inbound and outbound tunnels (remember that tunnels are unidirectional) for itself and for its local destinations (understand "services"). Each tunnel is composed of one or more participants, which includes the local router itself. Of course, a tunnel with less participants offers a weaker anonymity guarantee (down to no anonymity with a single participant). The first participant of a tunnel is called the gateway and the last participant is called the endpoint. For inbound tunnels, the local router is the endpoint, and for outbound tunnels it is the gateway. I do not have enough space here to detail how tunnels are built, but it is sufficient for our purpose to know that there is a distributed database maintained by a subset of all routers (those which opted-in) called the netDb which contains routers' contact information (called RouterInfos, which, for example, contains the public address of the router), and public destinations' contact information (called LeaseSets, which, for example, contains the list of inbound gateways to the destination). Each router has complete knowledge of the RouterInfos of the participants of its tunnels, but a participant of a tunnel does not know much about it: it only knows which router it receives data from and which one it has to send data to and, thus, it is also aware if it is an inbound gateway or an outbound endpoint, but that's all.

Now let's have a quick look at how tunnels are used. When a message is sent in I2P, it necessarily goes through an outbound tunnel of its source and then through an inbound tunnel of its destination. The outbound gateway is responsible for preprocessing the message which involves splitting it into fixed-size (1KB) fragments and iteratively encrypting them along with delivery instructions for layered decryption by the outbound tunnel participants. After that, each fragment is forwarded to the next participant. Each participant decrypts the fragments it receives and forwards them to the next, until the endpoint is reached. At this point, once the outbound endpoint has decrypted the fragments, it reassembles them to recover the preprocessed message, which is then forwarded to the inbound tunnel gateway. In turn, the inbound gateway splits the message in fixed-size (still 1KB) fragments, but this time it only encrypts them once, before forwarding them to the next participant. Each participant encrypts the fragments it receives and forwards them to the next, until the endpoint is reached. Then, the endpoint iteratively decrypts the fragments

and reassembles them. The message arrives at the destination.

I'm simplifying things here as my goal is only to give an overview of how I2P routing works. This way of transmitting messages allows I2P to use packet switching. This is actually a major difference with how Tor circuits work: it means that I2P can take advantage of the existence of multiple tunnels, not only for resilience but also for bandwidth (by balancing loads), and that tunnels can be short-lived (I2P renews them every ten minutes by default) rather than long-lived like Tor circuits, which makes traffic analysis harder.

Just like Tor, I2P can be used for so many things besides running and using anonymous services. The default install comes with a web server and a BitTorrent client, for example.

Now that we have seen a quick overview of how I2P works under the hood (more curious readers are encouraged to take a look at the awesome I2P technical documentation), let's start working on our goal: how to run an I2P hidden service.

0x1 - Installations

There are several implementations of I2P. The two major ones are the original I2P (written in Java) and the more recent i2pd (in C++)[3]. The latter probably has a smaller footprint than the former and its configuration is easier to manage (flat files instead of web UI), but it still crashed way too often the last time I tried to switch to it (I confess it has been almost a year).

Just like in the previous part on Tor, I will be giving instructions for Debian stable (which, unlike last time, is now Stretch).

So, to install I2P, create a new file called "/etc/apt/sources.list.d/i2p.list" with this content (you need to be root or use "sudo"):

```
deb https://deb.i2p2.de/ stretch
➥ main
deb-src https://deb.i2p2.de/
➥ stretch main
```

Save it and then add the GPG key that signs I2P packages to "apt" by issuing the following commands:

```
$ wget https://geti2p.net/_
➥static/i2p-debian-repo.key.asc
$ sudo apt-key add i2p-debian-
➥repo.key.asc
```

The first one will retrieve the key and the second one will add it to "apt". You can now issue the usual "sudo apt-get update" and it will retrieve the list of packages from the I2P repository. Then, install I2P and the I2P keyring so that the necessary GPG keys will be kept in sync and you do not have to worry about that later:

```
$ sudo apt-get install i2p i2p-
➥keyring
```

That's it. If you encounter any trouble regarding the use of HTTPS, installing the "apt-transport-https" package should fix it.

0x2 - The I2P Console

Once I2P is installed, your router should be running and its console waiting for you on "http://127.0.0.1:7657/". If it is not running, you can issue the command "sudo service i2p start" to start the router.

When you point your browser on the I2P web console, you will be able to see the network status in the sidebar. At the beginning, I2P is starting and searching for peers to build tunnels. If, after a little while, the network status is not "OK", you can click on it and you will be redirected to a page which explains the problem and provides potential leads on how to fix the issue (for example, if you are behind a firewall and need to configure it).

By default, the I2P configuration on bandwidth usage is pretty conservative; you can go to "http://127.0.0.1:7657/config" to change the default to better suit your network connectivity. Remember that by sharing more, you improve your anonymity, as more traffic that is not yours will go through your router.

After that, you are all set. I strongly encourage you to explore what the I2P console has to offer, not only in terms of configuration and information, but also in terms of services.

0x3 - Setting Up Your Hidden Service

Please refer to the previous part about Tor for my recommendations on where to run your hidden service. If you install I2P on a remote host (e.g., a VPS), you will still need to access its web console. For that, we are going to use SSH port forwarding ("-L"). Let's call your remote host "vps" and assume that your username there is "user". The following SSH command:

```
ssh -C -N -L 7757:127.0.0.1:7657
➥ user@vps
```

will open a local socket on port 7757 that forwards traffic to and from "127.0.0.1:7657" on "vps". The "-C" flag enable compression, and the "-N" tells SSH to not execute any

remote command so that it only does the port forwarding and does not open a remote session.

Now you can point your browser to "http://127.0.0.1:7757/" and you will see your remote I2P console. Your local I2P console (if any) is still accessible on "http://127.0.0.1:7657/". I will now use "console" as a shortcut for the console host and port - please adapt to your particular situation (local or remote).

To set up a hidden service, you need to go to the tunnel manager, which you will find at "http://console/i2ptunnelmgr". There you will be able to manage both client tunnels and hidden services (which could also be called "server tunnels"). By default, there are multiple client tunnels. You will see that each has a type. The "standard" type is equivalent to Tor hidden services: you can use it for any TCP service. Another important type is "Streamr", which can be used to tunnel UDP traffic through I2P (which Tor is incapable of). All other types are derived from the standard type to specialize for specific services. For example, HTTP and IRC types of service respectively filter HTTP headers and IRC commands to minimize risks of breaking your anonymity.

An interesting client tunnel that exists by default is "I2P HTTP Proxy", which binds "127.0.0.1:4444" to, well, an I2P HTTP proxy. This means that if you tell your browser to use this proxy (in Firefox you go to `Preferences > Advanced > Network`, click the "Settings" button of the "Connection" section, choose "Manual proxy configuration" and fill out the host and port for "HTTP Proxy"), you will be able to browse EepSites (".i2p" websites), but more on that later.

An interesting hidden service that exists by default is "I2P webserver". Indeed, I2P comes bundled with a web server which is by default binded to "127.0.0.1:7658" and serves files from "/usr/share/i2p/eepsite/docroot/". If you want to use that to host an EepSite, just put your HTML files in the "docroot" directory and you are almost all set. You may need to manually start it (the "Start" button in the "Control" column of the table listing your hidden services) and you probably want to edit the configuration of the service (by clicking on its name) and check the "Automatically start tunnel when router starts" box.

I am personally not using the bundled web server but rather the same setup as for my .onion (please refer to[1]). So, before going further on making the EepSite available to all, let's see how to set up an arbitrary service. I will use the same simple service as last time with Tor, but keep in mind that the process is the same for an SSH server, for instance.

As a reminder, the little service initializes the "counter" variable at 0 and then forever increments "counter" by one, waits for a connection on port 2600, answers with a single line saying hi and displaying the number of connection to the service since it has been (re) started.

```
counter=0
while true; do
counter=$((counter + 1))
echo "Hi, 2600 reader! Counter:
➥ "$counter"." | busybox nc -l
➥ -p 2600
done
```

In the I2P tunnel manager, we create a new hidden service by clicking the "Create" button at the bottom of the hidden services list after having selected the "Standard" type in the dropdown menu next to it. This takes us to a rather long form, but do not worry - most of the important stuff is at the top. We chose a name and a description for the service. Check the "Auto start" box. The host is "127.0.0.1" (localhost) and the port is "2600" (the one where our little service is waiting for connections). A default filename for the private key file is filled - I suggest that you customize it to be able to recognize that file later. Leave the "Local destination" field blank. The rest of the parameter has sane defaults, but I encourage you to check out the different options (most of it is either self-explanatory or you probably should not modify it). Remember that you can choose to encrypt the LeaseSet of your service before it is sent to the netDb, which will only allow people you share your key with to access it.

Important: your private key file corresponds to the cryptographic identity of your hidden service. You want to have a backup of it, as you will need it to move your hidden service onto another machine or to reinstall it after a crash. It is located in the "/var/lib/i2p/i2p-config" directory (you will need to be root to access it). To reinstall a hidden service, you can simply copy your private key file in this directory and indicate its name in the hidden service creation form.

0x4 - Accessing Your Hidden Service

To access I2P hidden services, you will need to go through a client tunnel. For EepSites, you can configure the I2P HTTP Proxy in your web browser as explained above and then go to `http://i2p-projekt.➥i2p/` for example (this is an EepSite mirror of I2P project's website). If it does not work at first, try not to despair. Sometimes I2P can be a bit slow to get all the necessary information and set up everything to work properly.

While your browser uses I2P HTTP Proxy, it will use the default outproxy ("false.i2p") if you visit the classical web. As said in the introduction, I2P is not intended for that, so you may want to use a different browser or switch back to no proxy settings. A good solution could be to create a Firefox profile that always uses I2P proxy that you would use only for accessing EepSites. To do that, launch Firefox with the command "firefox -no-remote -ProfileManager". The "-no-remote" argument tells Firefox to ignore any running instance of itself (otherwise it would ignore the command line arguments and simply open a new window of the running instance). Once in the profile manager, create a new profile that you name I2P, select it, and start a Firefox instance for this profile. Configure it to use the I2P proxy. Now when you want to browse EepSites, you can launch "firefox -no-remote -P I2P" (the "-P" selects the desired profile). Note that Firefox profiles do not share extensions or preferences. This is good privacy-wise, as a vanilla browser is harder to uniquely fingerprint than a strongly customized one. A less optimal but more convenient solution is to use an extension such as FoxyProxy which can tell Firefox to use different proxy settings depending on the URL, so you could tell it to use Tor when the URL matches this regexp "^https?://[^/]*\.onion/.*" and I2P when it matches the same regexp but with "i2p" instead of "onion".

To access other types of I2P hidden services, such as an SSH server or our own little greeting-and-counter service, the easier way is to create a SOCKS client tunnel. Go back to "http://127.0.0.1:7567/i2ptunnelmgr" and create a new client tunnel of type "SOCKS 4/4a/5". Name it something like "I2P SOCKS Proxy", choose if you want to start it automatically when the router starts, and assign it a port, e.g., 5555. The rest of the options already have sane defaults, but as for the creation of the hidden service, I encourage you to look at everything. Once the SOCKS client tunnel is created and started, you can use it just like we did last time with Tor.

The attentive reader will notice that we still do not know the ".i2p" name of the hidden service we created! To get it, you need to go back to the tunnel manager at "http://console/i2ptunnelmgr" and check your list of hidden services. For services of type HTTP such as the default EepSite, you will have a "Preview" button. It is actually a link so you can right-click it and choose "Copy Link Location" in the context menu to get its address in your clipboard. For other types of services, you will directly be given its "Base32 address" instead of the button. For our little service, this address is "khpazz3f747z5zet72s6g3dccw53bfdqyhxt-5da4sv7ouve5veuq.b32.i2p". Yes, this is quite long. It is the I2P equivalent of Tor's ".onion" names - the Base32 address of a service is derived from the public key of the destination. The good thing is that I2P has a mechanism for getting a .i2p domain that you can freely choose, but before going into how that works, let's connect to our service:

```
$ nc -X 5 -x 127.0.0.1:5555
khpazz3f747z5zet72s6g3dccw53bfdq
➥ yhxt5da4sv7ouve5veuq.b32.i2p 1
Hi, 2600 reader! Counter: 1.
```

Notice that I give "nc" port number 1. That is because I2P does not care about the port number. Apart from that, the SOCKS proxy works like any other, including Tor's, so you can configure your SSH client to automatically use it when the host ends in ".i2p" in the same way that we did last time for Tor.

0x5 - Getting Your Own .i2p Domain

There are three desirable properties that a naming system should meet: it should be decentralized, names should be meaningful, and names should also be securely unique. The theory is that you can only get two out of three (this is called Zooko's triangle). There actually are some sketches of solutions to get the three at the same time, such as Namecoin or GNU Name System, a part of GNUnet. Tor's naming system is decentralized and secure but not human readable. I2P is the same at the level of Base32 addresses, but has an additional layer which is decentralized and human-meaningful, but where names are not necessarily unique.

The idea is that each I2P router has its own address book, which you can access at "http://console/dns". The address book associates ".i2p" domains with destination keys. There are several parts of the address book: the local part (which includes a private part that will never be published even if your address book is public), and the subscriptions part. By default, your local address book is empty and your only subscription is "http://i2p-projekt.i2p/hosts.txt". You can get more subscriptions from registry websites such as "inr.i2p", "no.i2p", "stats.i2p/i2p", or "identiguy.i2p". For example, you can add "http://inr.i2p/export/alive-hosts.txt" to your subscriptions.

To make your domain usable by others, either they have to manually add an entry for your domain in their address book, or you will need to submit it to services that provide subscription lists that people are actually subscribed to. The first step for that is to verify that your name is not already taken by someone else (otherwise the registry service will not accept it as the main ones work on a "first come, first served" basis). Then, we will submit it to one or more of the main registries (listed above). The procedure is similar for all of them, so we will use "inr.i2p". Go there and use the search box to check that the domain you want is not already in use. If it is not, click "Register a domain" in the menu. Then enter your desired domain. It's a Base64 hash (you can find it in the "local destination" field when you edit your hidden service configuration) and a description of your hidden service.

If it gets accepted and is indeed alive when tested, it will be added to the host file of the service you submitted it to (and it will probably be picked up by the other services). There. You completed the final step!

0x6 - Conclusion

I hope you enjoyed reading this article and that you will put the freedom and the privacy provided by I2P to good use. Next time, we'll learn how to use IPFS, the InterPlanetary File System, to host a decentralized website.

0x7 - References

[1] *2600 Magazine*, Issue 34:2. If you missed it, the article is now also available on my web page (`https://pablo.rauzy.name/outreach.html`).

[2] I2P. `https://geti2p.net/`

[3] i2pd. `http://i2pd.website/`

[4] Namecoin. `https://namecoin.org/`

[5] GNUnet. `https://gnunet.org/`

Bitcoin or Bit Con?

One Newbie's Adventures in Cryptoland

by XtendedWhere

Depending on who has your ear, Bitcoin is either the greatest invention for humankind since the Internet, or the greatest financial bubble since Tulip Mania bloomed and wilted back in the 1630s.

Readers of these pages likely need no introduction to the infamous cryptocurrency, which first appeared in a white paper and software program released in January of 2009, shortly after the U.S. economic collapse and massive financial system bailout. Authored by an enigmatic and still anonymous persona, Satoshi Nakamoto, Bitcoin solved the "double spending problem" which had long hobbled systems of anonymous digital currency. In it, a publicly shared ledger, or blockchain, records all transactions, and copies of the ledger may be maintained by anyone, with all copies having to remain in perfect agreement.

I read about Bitcoin soon after its creation and found the technology interesting, but I had no use for it at the time, and gave it little further thought. Fast forward to the spring of 2016, when a speaker at a gathering of technology entrepreneurs in Los Angeles touted the wonders of cryptocurrencies and the amazing financial gains it had brought him. Rather than trading for pennies, the price of a Bitcoin had climbed to around $700 each. My curiosity piqued, I decided to dig in and learn more.

Searching the Internet for information and tutorials revealed much out-of-date, conflicting, and rather alarming information: tales of black market dealings and money laundering via the Silk Road website, the capture of its founder Ross Ulbricht and seizure of 144,000 Bitcoin by the U.S. Justice Department in 2013, and the collapse of Bitcoin exchange Mt. Gox and the disappearance of more than $400 million from its holdings in 2014. The many suspi-

cious sounding developments made me think twice about getting involved. Yet, seven years after its creation, Bitcoin persisted. What new and innovative technology doesn't face early hurdles and stumbles on the way to success? So I persisted as well.

Reading "How Money Got Free: Bitcoin and the Fight for the Future of Finance" by Brian Patrick Eha (Oneworld Publications, 2017) put the many Bitcoin stories in perspective and showed me how it operated behind the scenes. The primary drivers behind cryptocurrencies have great appeal: a decentralized system of exchange not controlled by any government or political group, the ability to make anonymous purchases and donations, worldwide reach, fast transactions, and minimal fees or "friction" in the system. In all, Bitcoin seemed like a promising technology for our increasingly digital world.

At last I felt knowledgeable enough to take action. Meanwhile, during my long "research" phase, Bitcoin's price had climbed from $700 to over $6,000! Clearly I'd missed out on some amazing financial gains.

Because reading about how to ride a bicycle and actually learning to ride are very different things, my time had come to climb on the cryptocurrency bike and actually buy some Bitcoin.

The next phase of my education had three goals:

1. Exchange some U.S. dollars for some Bitcoin.
2. Purchase some "thing" with Bitcoin.
3. Exchange some Bitcoin back to dollars (hopefully after the price went up).

The old saying goes "don't invest more than you're willing to lose." Had I bought $700 of Bitcoin at the start of my research phase in the spring of 2016 it would have become over $6,000, and a purchase of $5,000 back then would have become nearly $43,000 now! I'd watched the Bitcoin price climb steadily, so I picked a number I could live with losing - $5,000 - and went in search of a marketplace to take my money.

Although I made what I thought were the best choices at the time, I'm sure informed readers could offer endless alternatives. Rather than writing a "how to" article with specific products, services, and website names, for the most part I will focus instead on my experiences and how they exposed the realities of the current system.

Day 1 - Mid October 2017. To start, I needed a software "hot wallet" to securely hold my future Bitcoin, and from which to spend it. I downloaded several of the better-rated wallet apps, and set them up with secure passwords, main and backup email addresses, multi-word recovery phrases, and recovery hints. They offered many different options for additional security.

Now that I had a place to store my cryptocurrency, I selected an online exchange where I could trade some U.S. dollars for Bitcoin. The choices appeared quite varied, resembling everything from a money changing service at an airport to a shady character on a street corner. I chose one that appeared to have a good reputation, and created an account.

Here's where the first cracks appeared in the Bitcoin facade. To establish the account, I needed to provide a lot of personal information - just like when applying for a bank account or credit card. They needed my full name, date of birth, full address, country, and phone number, all of which they accepted online.

Then, to actually transfer some U.S. dollars (my "fiat currency" in the lingo) into the account, I needed to get approved for a still higher level of account. This involved providing my Social Security number, a copy of a recent utility bill, a high-resolution scan of my driver's license, and a high-resolution selfie of my face with my driver's license and a signed handwritten note. So much for the idea of Bitcoin anonymity!

After uploading all that, the site indicated that approval could take a week or more. Meanwhile, I performed a few more security activities: setting up strong passwords, two-factor authentication, a master key for account recovery, and a global settings lock. I felt ready and secure. But I still had no Bitcoin.

Day 2. A pleasant surprise arrived in my email with the approval of the higher-level exchange account. I logged in, but the site showed that the process had not been completed. After a day of back and forth with customer support, it finally came through.

Can I fund my exchange account now? Not so fast. The exchange site wanted my detailed bank account information. Given the history of hacks and data breaches (I've been affected by Target, Home Depot, and Experian - that I know of), I felt very uncomfortable putting my bank account number online. Fortunately, my bank made it easy to set up a new account that I could use as a way station for transferring funds. If that dedicated account were breached, I would stand to lose only the amount I had

decided to risk on this experiment.

Day 3. Actually moving my dollars into the new exchange account required a lengthy, multi-step process: I had to transfer money from my main bank account to my new way station account (no fee, and it happened right away), set up a wire transfer from my bank to the exchange website (no bank fee, but a $5.00 fee from the exchange), and reply with the bank's text verification code to authorize it. Then, as an additional security measure, my bank called to review all the details before they finally released the transfer. I then had to notify the exchange site to watch for the incoming funds, which they said could take up to five business days to clear.

So after three days of steady action and progress, I still had no Bitcoin in my account.

But now that I had fully tuned into the world of cryptocurrency, what should I find but a Bitcoin ATM sitting quietly in an entry to a local shopping mall. Really? Cash in, Bitcoin out? No need to open an account, or show ID? What could be simpler? So I pulled out two fresh $20 bills and stepped up to the plate.

But yes, it seemed too good to be true. The machine asked for a bunch of personal information, including mobile number and photo ID, then presented a long screen of terms and conditions. I agreed, inserted my cash, and a few moments later it printed a slip of thermal paper with two square QR codes that represented the public and private keys for my first tiny slice of a Bitcoin. Success at last!

Here's where the next cracks formed in the Bitcoin image. I did the math and found that my $40 cash turned into 0.00614183 Bitcoin, worth (depending on which source you referenced - different sites quoted values as much as $1,000 apart) a total of $36. What? The Bitcoin machine took a $4.00 service fee? Ten percent! Wow! What greedy bank charges that much to exchange foreign currencies? So much for the promise of "low to no fees." OK - so maybe I was optimistic to think that a lone machine in the mall hallway would be the gateway to vast riches. But at last, I owned some Bitcoin, and the price continued to climb.

Day 4. Another good email day. The wire transfer cleared in less than 24 hours, and my dollars were finally ready to exchange for some Bitcoin.

I logged in to the exchange site and set up a purchase, much like buying shares of stock, by specifying the desired price and quantity. After a few moments, the site indicated my purchase had failed to be accepted. I tried again, and that order failed too. Finally I set up a limit order offering to purchase 0.75573000 Bitcoin at the price of $6,598.90 per coin. Boom! That offer found a willing seller, and the trade went through.

The exchange site charged a fee of about $8 or 0.16 percent on the $5,000 trade, much better than the ten percent fee at the ATM, but still nearly twice as much as a discount brokerage charges for a typical stock trade. Nevertheless, I now owned just over three-quarters of a Bitcoin! Time to go out and buy something with it. But what, and where?

Reviewing apps and websites that listed merchants who accepted Bitcoin, I was surprised to discover very few in the Los Angeles area. The idea of spending my Bitcoin on a cup of coffee faded away, and the illusion of Bitcoin as a viable medium of exchange burst like a balloon. How can I exchange it for useful goods and services when so few merchants accept it?

Meanwhile the price of Bitcoin kept rising. In late November, it broke $10,000 for the first time. Two weeks later it shot past $15,000, meaning that in less than six weeks, I'd more than doubled my money. Should I have bought more? How long could this go on?

I sold some Bitcoin to recover my original investment, after which the price dropped, and then I bought some more (paying fees each time). Despite these multiple digital transactions, I still needed the experience of buying something "real" with Bitcoin.

Then I remembered – I'd always wanted to subscribe to *2600 Magazine,* rather than rely on the hit-or-miss encounters at the local magazine stands. Visiting the *2600* website, I put a one-year subscription into my shopping cart ($27.00), entered my address, and selected their Bitcoin service as the payment method. Now, their site uses a different wallet system than the one I'd been using, so I downloaded that app, set up an account, and moved the old $40 ATM purchase (really $36 after fees, but now worth more than $80) to the new digital wallet.

And here came three bullets that completely shattered the Bitcoin facade. First, moving the funds from my existing wallet app to the new wallet app deducted a fee of more than 16 percent. Free app with a huge cost!! Second, when I placed my subscription order, Shopify, which runs the *2600* store, added a "network fee" of nearly $13 (which seemed like the

amount they would need to pay to convert my Bitcoin back to dollars.) Third, the wallet app charged their own fee of nearly $13 for the transaction. Finally, on top of all that, since I'd chosen the lowest cost network transaction verification method (!), the system warned me that my payment might take longer to confirm, or may not confirm at all. So much for the illusions of Bitcoin's low cost and high reliability!

Typically, a merchant who accepts credit cards pays two to four percent of each transaction to the credit card processor and then absorbs those fees. But when a fee is nearly half the subscription price, obviously a vendor such as *2600* can't eat that amount, so they pass it on.

Looked at from one point of view, I paid nearly $39 in fees for that $27 subscription - 140 percent! Normally I would consider myself an idiot for being taken like that. But this is scientific research, and the costs were part of the experiment. Looked at from another point of view, the $40 cash I put into the Bitcoin ATM became, a few months later, a $27 subscription and about $15 worth of Bitcoin, so I actually made $2 in profit - but only thanks to the outrageous rise in the Bitcoin price.

Despite the warnings, the transaction cleared in under an hour, and I soon had a nice email from *2600* indicating my new subscription had started.

But the absurdly high fees charged for every Bitcoin transaction became a deal breaker for me. I was done. After hitting a peak of $19,000, Bitcoin had been almost continually falling and I wanted out. I returned to my exchange account and placed a sell order, which went through promptly. I set up a wire transfer back to my bank (another $5 fee from the exchange, zero fee from my bank). Six days and several emails to the exchange later, the funds finally appeared back in my bank account. Success, and end of experiment!

Conclusions

Maybe I made some ignorant errors along the way. Maybe I didn't do enough research, locate the most efficient services, or the best methods of exchange. I've since learned that transaction fees can vary greatly due to network demand, which had been very high during the time of my experiment. Like a tourist with an unfamiliar currency, I may have grossly overpaid and not understood the local customs.

In summary, I got lucky. I achieved all the goals of the original plan and I learned a lot. I put a total of $5,045.00 into the experiment. With all the buying, selling, and wallet transfers, I paid about $90 in fees. At the end of the project I had:

- $9,868.85 in cash (minus the original $5,045.00 which meant a net profit of $4,823.85)
- A fraction of a Bitcoin (too small to even cover a typical transaction fee)
- A one-year subscription to *2600 Magazine* (a $27 value)
- Plus, at tax time, I expect there will be capital gains to pay on the profits

None of the promises of Bitcoin turned out to be true in any way. Bitcoin is not anonymous, not instantaneous, and not low cost. The outrageously variable transaction fees and lack of wide adoption make it a poor medium of exchange. Its outrageous volatility makes it a poor medium of investment. And, compared to racehorses or sports teams, its unpredictable behavior makes it a poor medium for even gambling!

Bitcoin's price has continued to fall, dropping below $9,000 just before I handed this story to the publisher. So where is it now? By the time you read this, Bitcoin may have crashed and burned, as many have predicted. Or it may still be crawling along, luring the curious and uninformed to put their money into its edacious system.

But as hackers know, success comes with persistence and the willingness to revise one's plans in service of achieving the greater vision. The fundamental concepts behind Bitcoin seem useful and necessary, and the blockchain technology offers great promise. It remains to be seen how to best implement it all.

Maybe Bitcoin will morph into a more capable version of itself. Perhaps it will be replaced by one of the many alternate cryptocurrencies, or one yet to be created, which will find the right combination of anonymity, security, low cost, ease of use, consistency and efficiency. Clearly someone needs to come up with an easy, all-in-one solution that serves the needs of the masses, and lets cryptocurrency become omnipresent, much in the way that America Online's ease of use helped email soar in popularity in the 1990s.

Perhaps a successful "AOL of Crypto" service will come from a reader of these pages. Either way, I'll be watching and ready to experiment again when it happens.

The Case of the Murderous AI

by Ted Benson

I once shared an office with a man named Branavan. We were PhD students working on natural language processing - the applied branch of machine learning that produces things like Siri and Google Translate.

AI labs aren't the most dramatic settings for a murder mystery. None of the mysterious machines a physics department would have. Just desks and computers and whiteboards filled with math.

Five of us were crammed into that tiny office, with more than twice as many computers under our desks, each packed with as many processor cores as money would buy. That room sucked up enough electricity to power a small town. Even in the freezing Cambridge winters, with snow falling outside, we'd have to leave the window open just to keep from sweating.

All of us were working on interesting problems, but Branavan's work had real sex appeal. He taught computers to play video games. And not just play them, but to incorporate knowledge from blog posts and strategy guides the same way a human would.

On any given day, you'd walk into the office to find Branavan sitting there in front of a wall of monitors watching 16 video games unfold simultaneously, like the Architect of *The Matrix*. The players on each screen were controlled by his AI, trying new tactics and correlating them with sentences from the strategy guides Branavan had fed them. Beneath his desk, his computers would slowly bake us to death as their CPUs reached volcano heat from all the processing required.

Now, Branavan was a business savvy guy. He knew video games made for great demos, but the real money was automating IT. Think of all the people whose jobs consist of reading lots of computer manuals so they can repeat the steps themselves later. What if a computer could learn to read those manuals and manage your IT department for you?

"Siri, configure my new router!" was the basic idea.

So Branavan graduated his AI from its video game career and gave it control over the entire computer. He stopped feeding it video game strategy guides and started feeding it IT manuals.

The thing actually worked! You'd give it a help desk page from HP's website and it would dutifully follow the instructions for you, moving the mouse around and clicking on the screen as if a ghost was setting up your new printer.

There was just one problem... the AI would regularly commit murder-suicide against all the other copies of itself.

--

The murders would happen in the middle of night, when nobody was there.

Before he left home from work, Branavan would fill his Matrix Architect rig with virtual machines and place them all in learning mode. All night they were to practice reading manuals, doing IT tasks, and checking whether they had completed their objective.

When he'd return in the morning, he'd find his computer had literally offed itself. Monitors black. Power off. AIs gone with no trace of what had happened.

At first, he suspected the cleaning crew, or maybe a fellow lab mate trying to save electricity by flipping off the switch late at night. But that was easy to rule out with a polite note stuck to the machine.

The obvious possibility was a bug. Maybe a SEGFAULT was taking down the entire computer. But, try as he might, he couldn't

find a bug. Plus when the computer restarted, his logging system had recorded no trace of a crash.

No, the more he ruled out alternatives, the more it became clear this was a case of cold, clean, premeditated murder.

The same AI would happily play video games for weeks straight. Installing printer drivers certainly isn't fun, but was it really enough to drive a computer off the edge? It was a mystery for days. It would happen some nights, not others. Always at night. Never a trace.

And then one day, gazing up at his wall of monitors, Branavan caught it in the act.

On one monitor, the one not inside a VM, the mouse veered to the corner of the screen. It clicked the Start button we all grew up with. It slinked up to "Shutdown" and when the box appeared, clicked "Confirm."

Zap! zap! zap! One by one, all the monitors blinked off. The AI bots dead in their tracks.

VMs murdered. No suicide note on the host machine.

--

Plenty of ink has been spilled over Skynet-style AI doomsdays, but not much has been written about the AI child accidentally shooting his father's gun.

In areas of AI in which control over an open-ended world is required, computers learn most effectively just like humans: by doing. ("Reinforcement learning" is the industry lingo.) The computer starts off making random actions and, over the course of countless retries, begins to devise strategies that correlate actions in a particular situation with some definition of "success," like winning a game or installing a printer driver.

Over the course of those repeated trials, the computer would ideally also learn what actions to avoid. In a game, shooting all your bullets at the sky doesn't correlate with winning.

But the murder-suicide of Branavan's AI is something special. Something the computer can't learn from - nobody can learn from - because the penalty for throwing the off switch is so high that there's no chance to reflect and try again afterwards. It's a blind spot in the algorithm's ability to learn avoidance.

No matter how good the computer got at learning what to do and what not to do, there was always that Shutdown button.

The computer would be given a task and it would sit there, evaluating its actions. A likely score of +20 opening the printer folder. A likely score of +0 for shaking the mouse. A likely score of +10 for copying a file. And a big question mark next to the option of the shutdown button.

Always a question mark. The blind spot remained.

Because every time the computer took a chance on that option, there was no one left to record how well it worked out.

--

Branavan, of course, found a straightforward solution to this blind spot quandary.

From above, he dictated *Thou Shalt Not Kill*. (He just added a line of code that forbade the computer from confirming a shutdown.)

It's a line of code every civilization in history has had to write for itself, so it was bound to be given to computers eventually.

But it does make you think: most control systems are far more open-ended than a desktop operating system. Don't kill is a good objective, but will it always be so easy as preventing the computer from clicking a button? Self driving cars, robotic surgeries, assembly lines....

When you retire in that country house you've been dreaming of, your elder care robot will have a perfect knowledge of the different temperatures at which to make different kinds of tea.

But inevitably, your cottage's kitchen will be unique in little ways. The placement of the cabinets. The rotation of your cream pitcher. Your unusually large mugs.

Inevitably, your recently purchased robot will have to learn how to navigate your new home using a series of experiments. And by definition, experiments require actions never before taken.

What happens when I open this cabinet?

What happens when I use that kettle?

What happens when I put the cat in the tea kettle?

What happens when I pour the tea on Grandma's lap?

One wonders how else we'll need to codify basic decency.

In Defense of the Net

Why U.S. Defense and Intelligence Agencies Have a Vested Interest in Preserving Net Neutrality and How They Can Help Protect It

by davemitchell

This article relies on the assumption that a vanishing of net neutrality (a process which appears to be speeding up in accordance with the current political climate) will notably slow the creation of content on the Internet. A basic premise of a non-neutral net is the separation of traffic into lanes based on whoever is willing to pay the most. An open and free net encourages individuals to produce videos, blog posts, and other multimedia content in order to spread a message they believe in. According to Nielsen Holdings, a global measurement and data analytics company, there are over 181 million blogs in the world, 168.3 million of which self-host their blogs, meaning they are not posting on a popular blogging platform or social media site. These sites, such as Blogger, Tumblr, and Facebook, are profitable enough to pay for "fast lane" access to Internet consumers, but a majority of bloggers do not use them, instead opting to build and host their own blogs on their own sites. A segregated net would very likely discourage these individuals to continue making this content if they are unable to pay for it to reach their readers in a reasonably timely manner. Why make something to show the world when it's very likely nobody will bother to wait around to see it?

So why should the U.S. defense apparatus care? Their livelihoods don't depend on reaching customers or producing content online. They don't rely on an open, equal network to post their ideas and opinions - they use traditional media. They don't need the Internet to communicate their data - they have their own high-speed networks for that. Put simply, the freer the Internet is, the greater the number of citizens who have a presence online, the greater the amount of data they produce, and the greater the ability of the intelligence apparatus to construct social networks and profiles on targets.

Some may argue that when the ability of dissidents to publish their thoughts is diminished, it becomes much more difficult for them to spread their ideology and opinions, therefore reducing the level of overall discordant activity. While this may be true, it will also become much more difficult for intelligence agencies and their analysts to pinpoint potential trouble populations when their present massive well of information dries up. Currently, the National Security Agency taps up to 80 percent of all global communications via the interception of data traveling through underseas cables, according to *The Guardian*. The ethics of this wiretapping is beyond the scope of this article and is worthy of volumes on its own, but the fact is that the U.S. has constructed massively expansive surveillance systems, many parts of which depend on the collection and analysis of tremendous magnitudes of data. Current intelligence programs make heavy use of this enormous amount of data in order to work effectively, and the loss of it could prove to be a tremendous detriment to American intelligence and defense operations. Any hacker who has dabbled in the fields of machine learning and artificial intelligence will tell you that systems with more input data are exponentially more powerful in predicting accurate outcomes than systems with lesser amounts.

It is in the interests of the U.S. security apparatus to keep citizens online, communicating and discussing ideas in a free and open format in which all people can host an easily-accessible, self-published collection of opinions, regardless of their ability to pay for access to readers. Identification of dissidents using current systems, while ethically questionable, is exponentially easier the more opportunities they have to speak their views. Security agencies, therefore, should be at the forefront of the fight to protect neutrality on the Internet. It is in their direct interests to ensure the equality of all traffic and it is definitely within their sphere of influence to swing other government bodies in the same pro-net neutrality direction.

I realize that this viewpoint presents ever further questions and problems, as any discussion on net neutrality and privacy rights ought to. Yet, if we are to fight for a free and open Internet, it is essential to get as many powerful individuals and organizations onboard as possible, including those with whom the hacker community has historically been at odds with. These agencies wield extraordinary power within Washington and the country as a whole and, by convincing them that net neutrality is indeed in line with their direct interests, it would be possible to at least continue the conversation about a neutral net long enough for the current divisions rampant throughout the U.S. political system to settle down and reach a point where bipartisan compromises between the people, the government, and industry are possible and probable.

The Hacker Perspective

by Marc Lighter

It is not uncommon for a word to become misused in the common vernacular. When the media starts directing a narrative, the general populace grabs onto it like the sixth proton on a carbon-14 atom.

Growing up, our first family computer was the Apple II. It had a whopping 4k of RAM. I remember the feeling of popping open the case and upgrading it to 16k of RAM. From that point on, I was hooked. Back then hobbyists were known as "hackers." I was young then, but the feeling of excitement in hacking a game called *Wizardry* saved me hours and hours of time searching through dungeons for treasure. Sure, the search was fun, but hacking the game was more fun than the game itself.

This was about the time that the first modems arrived (300 kbps). You'd watch each character pop onto the screen, one at a time. It was like watching a monkey type in slow motion... and we loved it. *Close Encounters of the Third Kind* was a popular movie back then and the idea that technology could enable everything (including conversing with aliens) was all very real and possible. The whole era was the golden age of hacking. Back then, a hacker was a hobbyist who tinkered with anything technology-related. We built our own computers, we were ham radio operators, we were electronics geeks. We loved to experiment and try things that no one else would try.

The movie *WarGames* came out and it was the first real "hacker" movie that I can recall. Finally, a movie where the hacker was the star! That was around the same time that stories came out in the news about "miscreants" invading computer systems and the media grabbed onto the big buzzword of the time... "*Hacker*." Some of us recoiled in shame, others used it discretely as if members of a secret society. We could no longer proudly proclaim our association with the term: it had become synonymous with "criminal." We might as well be mobsters helping to direct organized crime families commit heinous murders. Some of us just gave up trying to convince people that this word, once innocent and prestigious, was now twisted and bastardized and now invoked images of some hideous and terrible creature, like Gollum on a keyboard.

Some of us just shrugged and left the word behind, giving up on the battle to save our badge of honor. Some of us relented and started using it the way the media had decided we should use it. We became those whom it described.

Early into my IT career, I found myself fixing and repairing computers. We were a bunch of young guys who all had the same dreams and aspirations at work... to get home and surf the net every night and ftp the latest game or hang out on the local BBS trolling for girls that never showed up. We made a living fixing the bugs in software and replacing components in hardware. Things were buggy back then and most end users blamed the computer for everything. (Some things never change.) Security was an afterthought and most businesses we worked with didn't even have a firewall. If you were in the trenches with us, you knew the domain admin password, so you could peek into just about anything that you wanted... even the boss's email. Most people didn't even worry about locking their computer, so we used to wait until they left and would send inappropriate emails to each other or send an email to the boss telling him "*I quit!*" while laughing hysterically as the poor schlub tried to explain to the boss that he hadn't really sent that email.

Wireless was when we really had some fun. If you aren't that old yet, I wish you could have been there during the early days of wireless. War-driving became a habit that was hard to break... even harder to break than

the crappy WEP encryption they supplied. We would have war-driving parties where we would drive all over town with a $5000 Compaq laptop and a Yagi antenna hanging out of the passenger window to surveil any wireless signal we could get a hold of. Once in, we could literally watch your every move online. Of course, this was before there were any known laws against such activity. We were breaking new ground and even the police weren't sure what to charge us with (other than loitering). One night, we stumbled upon the wireless traffic of a well-known public official chatting it up with his lady friend (while his wife was sitting in the other room). We know that she was in the other room because he shared that with his mistress on more than one occasion. While we were impressed that he even knew how to use IRC, we laughed so hard at some of the conversation that we about blacked out from the lack of oxygen.

This was about the time that we decided that we could make a business out of security. We would drive all over town and charge businesses to set up or secure their wireless networks. The look on their faces was priceless when we showed them their user names and passwords to all of their online sites. We had some business owners shut down their wireless, permanently. Others knew that it was in their best interest to let us help because they couldn't afford *not* to have wireless or their customers would complain. We had an interesting conversation with the law on a couple of occasions. One thing about the law that you probably already know... the nicer you are to them, the nicer they usually are back to you. In the past, a lot of them barely used technology, let alone knew what we were up to. I can recall one business owner who complained and called the cops. We stood there and politely explained what we were doing. The business owner didn't even bother to encrypt his connection. *Anybody* could pick up his wireless signal and watch his activity, we explained. We even showed the officer how it worked. Needless to say, the cop was impressed and had a nice conversation with the business owner explaining that there were no charges that he knew of that could be filed. Later that week, the business owner called us back and asked us to secure his wireless network.

I'm sure wardriving is still a thing, but it's more time consuming to break encryption these days than it was in the past. Some of us moved onto legitimate jobs and some probably didn't. I only know what a couple of my ex-colleagues are up to. One became a virtualization specialist and does work for big corporations. Another runs an IT department at a local hospital. Me, I still run a small IT business and help other small businesses keep their computers running. I've just sat back and watched as the security business has exploded. While some of the technology has made it harder to penetrate systems from the outside, getting inside is still very easy if you know who to talk to and you have their email address. The best firewall in the world won't protect against some rube clicking on an attachment that they think came from Fedex or Amazon. And drive-by downloads? It's amazing the websites that people will browse while at work.

I remember when ransomware hit the scene just a few years ago. I was working at an automobile parts manufacturer when users started complaining that their files wouldn't open. Initially, it came in as an email attachment. The subject line didn't even sound legit. Some users will click on anything, I swear. Take it from me, hacking will never be difficult as long as there are stupid people in the world... and trust me, there are millions of them out there. One woman at an accounting firm clicked on an attachment that infected the entire network with ransomware. It took us almost two days to clean it up and get them back online. Then a couple of weeks later, she did it again. Same user... same email attachment. The email was simple and just said, "Your deposit didn't go through. Please click the attachment to reissue the deposit." "Click!" Bang! All your files are encrypted, lady. You would think that she would learn after the second time, but *no!* She did it a *third* time. Well... I bill by the hour, so it's no skin off my back.

Sometimes, when I have the time, I like to think about where the future of hacking is going. We are living in a world with tighter and tighter government restrictions, kids.

Before 9/11, it wasn't against the law unless there was a law on the books outlawing a particular thing or activity. Today, everything is against the law unless they say it's okay to do it or to own it. Case in point: I love flying my drone and to do some people-watching. Technically, it's against the law if you fly your drone over a private residence or a crowd of 100 or more people... but they don't have a clue that I am there. I can fly high enough that nobody can hear the drone. Yes, I know that you aren't allowed to fly over 400 feet... but there are ways around that, too. A drone is a device that still gives you the freedom to go where you want and see what you want as long as you are cool about it. Load that thing up with FLIR and there is no privacy anywhere anymore for the common man.

It's illegal to fly a drone in a national park. However, you can watch some online videos of drone pilots complaining that they got fined because they posted their drone videos online. Tip one: don't post your drone videos online. Tip two: don't fly your drone near an airport. Even if they don't catch you, you could put lives in jeopardy. Don't do it. The drone wars continue unabated and I'm sure the laws will become so restrictive that there will be one park left in the U.S. soon that will be for dedicated drone pilots. If you want to fly your drone, then you have to go to the last one acre plot in the backwoods of Kentucky reserved for drone pilots. You can only fly it up to ten feet in the air... have fun, kid.

Yes, the future looks bright for hacking. The "Internet of Things" will embed devices in just about everything. In the future, we will be checking out our neighbor's fridge inventory and ordering a case of Spam to be delivered to his house, just for the irony of it. We will be hacking into your Alexa just to hear what's going on in your living room. (I'm sure somebody has already done that, right?) Pretty soon, we will be hacking into your Nike shoes and adding 1,000 steps so that you think you got a great workout today. Heck, maybe we will even hack into your smart watch and make you think your heart rate is too slow, prompting you to make a medical visit for no real reason. Yes, there will be ways to have fun in the future.

Now, I've already been around a long time and I'm going to make some predictions that you may or may not agree with. I'm about to go all futuristic on you so be prepared. I predict that in the future, hacking will be dead. What?!? You might ask. Why?!? Two words: Artificial Intelligence.

Just hear me out.

Artificial Intelligence will be the game changer in technology. Assuming that we survive as a species, AI will surpass us in every possible way. AI will predict what your next move will be - in milliseconds. It will patch vulnerabilities in nanoseconds. When AI reaches critical mass or mass penetration (however you want to describe the singularity), it will be difficult for humans to creatively exploit the machines anymore. Maybe you think that human creativity and ingenuity will win the day. (And maybe it will win... for a while.) But, eventually, AI will be smarter than all of the smartest people in the world combined and it will operate at an intelligence level that is unimaginable to us today. Maybe if we can find a way to tap into that intelligence with some neuro-biological interface, we can push that day off. However, that day is coming and it's just a matter of time before we find out if a super-intelligent mind will actually care what we do on a daily basis.

But, until then my young Padawan, enjoy the opportunity you have today to change the world around you and mold it to your will. Use your skills for change that will benefit others (or just cash out with your big hacking payday, I don't care). Just remember that you lived in the golden age of computers and then decide how you will make your mark on the world.

HACKER PERSPECTIVE submissions have closed again.

We will be opening them again in the future so write your submission now and have it ready to send!

A Review of CopperheadOS

by Ron Porter
ron@jadero.com

CopperheadOS (COS) is a smartphone operating system based on the Android Open Source Project (ASOP). In that, it is like any other AOSP customization produced by the various manufacturers and carriers. What makes it different is the modification philosophy. Rather than adding a bunch of bells and whistles or worse, COS has a single-minded focus on security and privacy.

Like ASOP, COS is an open source project. That means the code can be inspected and modified by those with the skills to do so. It also means that those with proper skills can submit changes for the COS team to evaluate for inclusion in the mainstream OS.

At the time of writing, the main revenue stream supporting COS is the sale of the Pixel line with COS installed and the installation of COS to Pixels you send in. Copperhead is working on a reseller network, which may be in place by the time this is published. A reseller network will presumably stabilize and increase revenue or at least reduce distractions to continued development.

How Does COS Achieve Its Goals?

The developers behind COS do a number of things to enhance security and privacy over ASOP. ASOP is itself based on Linux, so they take the obvious step of pulling in the relevant security and privacy features of Linux that Google does not already include. They also look to other open-source operating systems like BSD.

Copperhead is committed to keeping COS up-to-date with the latest security patches from Google, other sources, and their own work. Updates are pushed to the phone about once a week and most of those have some security- or privacy-enhancing features.

COS does not include Play Services, the foundation of the Play Store, and Google Apps like Maps, Wear, voice assistant, etc. Many third party apps also depend on Play Services. Even many who are concerned about Google's practices will find this tradeoff unacceptable.

F-Droid is installed as the default app store. The selection is not as good as the Play Store, but as a major distributor of vetted Android open source software, F-Droid seems to be a good fit.

COS also does not include a true SMS app because of concerns surrounding the privacy and security of SMS. Silence, the secure messaging app installed by default, does provide SMS as a fallback, but the clear intent is to avoid the use of SMS. Silence can also be used to make secure voice calls.

DuckDuckGo is set as the default search engine. It is a privacy-focused search engine, making it a good match for COS. It also happens to already be the choice of many privacy-minded people.

Does COS Meet Its Goals?

Keeping in mind that COS is young and the team small, I would say that yes, the goals are being met. To me, there are some misses, but I'm also not willing to second guess the team at this point.

I would like to see a default email client that easily supports public key encryption and signatures, but K9 Mail is easy enough to install from F-Droid.

I would like to see some fingerprint and password failure options. The fingerprint sensor does get disabled after five failed attempts, but there are no options to manage how frequently you are forced to use your password, no quick way to temporarily disable the fingerprint sensor, and no way to force a

wipe of the phone after a number of failed password attempts.

What's Really Missing?

Apps for social media, banking, and other Internet-based services like Google Maps are easily worked around by using the websites directly. In many cases, that is less convenient or leaves you without some very desirable features. In some cases, there are effective alternatives available on F-Droid.

The real loss is in offline apps. F-Droid doesn't come close to Play Store for the variety of high-quality apps and games. There are alternatives, but I still feel like I've taken more than a few steps backward in what I can actually do with my little pocket computer.

Who Is It For?

There are two things that probably make COS unsuitable for the average user. First is price. Although Nexus 5 and 6 versions of COS are available for free download, the preinstalled Pixel is over US $1000 and the Pixel XL is nearly US $1500. If you already have a Pixel or Pixel XL, you can send it to Copperhead and have them install COS for US $300. If you don't have a Nexus 5 or 6 and the skills to build and install an alternative OS, then you are going to have to buy one of the Pixels or send one in for Copperhead to install the OS for you.

The second issue is lack of utility. At any price, few are interested in a phone that has limited app selection, and virtually no access to the services we have come to take for granted: voice assistant, media stores and players, touch-to-pay, wearable support, etc. I don't know if it's even possible to address the apparent conflict between security and utility, but as long as consumers have to choose, security will always lose.

My Personal Experience

I purchased a Pixel XL direct from Copperhead. I think I have what it takes to do the work myself, but this was my way of supporting Copperhead. I also wanted to get a feel for what a regular user would experience so that I could make appropriate recommendations to others.

I consider these devices to be computers, not phones, so the price was not really a deciding factor beyond how it affected our budget. My perspective might be colored by the fact that I'm old enough to still be amazed by the technology we have. I was thrilled to be able to buy a real computer for only $1000 a few years after my son was born. Yes, it was only a VIC-20, but the Apple was over $3000. Every computer I've ever bought or assembled has cost $1500-$3000, so $1500 for a real, Internet-connected computer that fit in my pocket was really a no-brainer. Being a programmer, I was also not put off by the initial lack of utility. Other than big things like voice assistance, I know I can work around or develop my own solutions for the things I really miss.

So far, I've managed to find alternatives or workarounds for everything except Tasker, an automation tool. There are alternatives, but they are not nearly as capable as Tasker, so I'm going to have to start writing "real" software instead of building Tasker scripts. The only function provided by an app that I've had to do without completely is Prairie Coordinates. As a volunteer firefighter, I used this to convert Township-based land locations to GPS coordinates for navigation. Now I have to pull out the paper map like everyone else.

If I had a true need for apps available only on Play Store, I would not have elected to go with COS. As I mentioned earlier, F-Droid is the default place to get apps. Amazon App Store is also available, although selection is still limited and may not be suitable for the truly privacy conscious. If you really need both COS and Play apps, Yalp, available on F-Droid, will get you access to Play Store. I haven't tried it, mostly because use of Yalp seems to be against Google's terms of service. I don't think Google has ever banned Yalp or Yalp users, but, for me, that's not really the point.

The Pixel camera hardware is pretty good, but neither the default app nor anything I could find on F-Droid really takes full advantage of its capabilities.

Android Wear is not available. Gadgetbridge, available on F-Droid, enables the use of some wearables, but with reduced function. For example, I thought the killer feature of my Pebble was the ability to send canned replies to incoming messages. That feature is available when Gadgetbridge is running on the stock Pixels, so COS notification security must be getting in the way.

I didn't even try switching to Silence. A few years ago, I convinced some key contacts to switch from their default SMS apps to Signal. I don't want to start that all over again, so I just grabbed Noise from F-Droid. Noise is an alternative build of Signal that is fully interoperable with stock Signal.

I've been getting COS updates about once a week. In addition to direct COS enhancements, the updates include security patches from ASOP and elsewhere. COS, at least on the Pixel line, is now based on Oreo, the latest from Google. I'm on the stable channel, but it's easy to switch to the beta channel if you want to. Personally, I'm not quite ready to go beta on my only phone.

The update process is trivially simple, at least from the user's point of view. COS comes with automatic updates turned on for all connection types. Updates are pushed as deltas (difference between current install and updated version) to minimize traffic. They are downloaded and installed in the background. COS takes full advantage of modern A/B technologies so that the only downtime is for a very quick reboot. Even that reboot can be set to happen automatically when the system is idle. If the reboot fails for some reason, the A/B system means that it just automatically reboots again, this time to the previous version.

Conclusion

Overall, I'm very happy with my decision to go with COS. My personal attack surface has always and will always be my responsibility, but I'm grateful to have smarter people than me trying to make sure that I'm starting on a solid foundation.

While it's definitely not for everyone, COS should be a welcome addition to the operating system space. It's not the only Android-based OS that claims to provide improved security and privacy, but it's probably the easiest one to get into if you can handle the price. *Shout-out to The Revisionists*.

```
$ cat ~/.ssh/id_rsa.pem
-----BEGIN ARTICLE TITLE-----
```

SSH Keys and Challenges in Enterprise Environments

```
-----END ARTICLE TITLE-----
```

by Patric Schmitz
Pat@cyber-schmitzel.net

The Secure Shell protocol was invented in 1995 to overcome the lack of strong, encrypted authentication of remote tools like telnet, rlogin, rsh, and similar protocols. Security improvements have been implemented in version 2, which was released in 2006. More detailed information on SSH can be found in RFC 4251, RFC 4256 (and a couple of other RFCs, but these are the basic ones), and a web search.

This article will focus on the SSH-2 protocol, which basically utilizes a Diffie-Hellman key exchange and public key cryptography to authenticate the remote computer. The user can authenticate either via passwords, or as well by a public key authentication.

In general, SSH keys are known to be more secure than a password authentication, since when using SSH keys no password is being transmitted over the wire at any point in time. It is important to note though that the authentication method has no influence on the security of the connection itself.

To authenticate with a private key, the client will first send the user ID to the server, which will then refer to the corresponding authorized_keys file (the location and the file can be set in the sshd configuration file, so it's not necessarily authorized_keys), utilize the contained public key to encrypt a random number, and send it back to the client. The client will be able to decrypt the value with the correct private key, calculate a hash value of the number, and send this back to the server. The server can now compare the hash values and, if these match, the user is authenticated.

Private keys can and should in any case be protected with a passphrase. A passphrase is nothing different than a password, hopefully longer.

Since it is, of course, more secure not transmitting passwords over the wire at all - and, if protected by a decent passphrase, the private key might be better secured as well - there are still a lot of challenges that SSH keys cause in

enterprise environments. I'm not necessarily trying to come up with solutions for these challenges, but want to try to help admins face these and help them to come up with decent processes and workflows. Since I have seen quite a few enterprises using SSH key authentication, I can tell there are many misunderstandings and sometimes even thoughts that haven't popped up before when talking to administrators.

I want to stress again that the SSH connection initiated with SSH keys is *not* any more cryptographically secure than the one with password authentication. While this seems to be a no-brainer, I've met too many IT people who really think that it is. It is just another way to authenticate, which for sure has advantages. So in any conversation, I try to find out the reasons why people insist on using SSH keys to authenticate.

Now let's dig a little deeper into the challenges of SSH keys in an enterprise environment. When using SSH keys yourself to log on remotely to your box at home, to one in the cloud, or to any device supporting it, it is your responsibility to know which key you use for which account. Maybe you just use one single key for every device, protect the private key with a strong passphrase, and keep it in a secure place. But what happens in an enterprise environment? There are several - sometimes hundreds or even thousands of users - with one or more key pair. One challenge is the private key security. As you already knew, or read further above in this article, the server just receives a hash value of an encrypted random number it sent to the client. The server does not know about the private key security. Is it protected with a passphrase? There is no control on the server if the private key is protected or if the passphrase is strong enough. It's completely up to the user. And users sometimes tend to be lazy. So they might use a passphrase... or not. We cannot put any technical mechanism in place to enforce a passphrase on a private key.

A private key is a file and, in contrast to a password, has to be stored somewhere. With passwords, it's nowadays quite commonly known that you shouldn't write those on Post-its and stick those to your screen or place them under the keyboard. But where to store your private key? In combination with the fact that there is no way to enforce passphrase protection for the private key in common SSH implementations, this can become a security risk.

This is especially true since the public key authentication is not bound to a named user account. A named user account is an account that belongs to an identity, so it's linked to a real human being. A generic account is an account like, for example, root, which is not exclusively connected to a single identity.

Whatever public key is in the authorized_keys file on an account will enable the corresponding private key to log in as this account. There is no difference in behavior for named or generic user accounts here. So someone who once had access to an authorized_keys file could place his public key in it and now log on as someone else - authenticated, not being highlighted in any log file as an attack. This might work for years without the real user noticing, since often there is not a process in place to force a user to change his key pair on a regular base. This could almost turn a named user account into a generic one. I've seen people, especially in automotive R&D, add colleagues' SSH public keys to their authorized_keys file (which is owned by the user/account) to enable them to work on their projects when they are off sick or on vacation.

For generic accounts, it's even hard to find out if a key in the authorized_keys file is supposed to be in it or not. A public key entry in the authorized keys cannot be identified or connected to the user that once created it. There is a comment field for the entry, but that can be altered to whatever value, or just left blank. (Hohoho santa@northpole.xmas now has access to root on several machines... let's go out and find him!) In a grown IT landscape, it is very hard to identify which public keys are wanted and could be marked as "approved" or "known" and which are not.

How do we know if an entry is used by an application for automation, by an entitled co-worker, by someone who already changed departments years ago, or maybe even by an attacker?

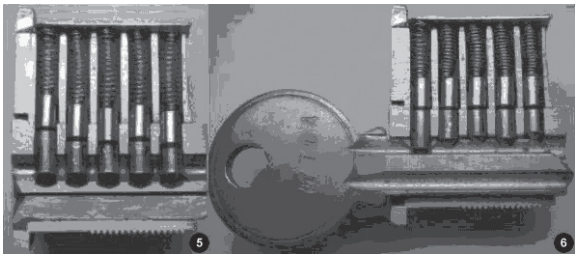
In most cases, it is very risky to remove entries from authorized_keys files, since they might really be used by applications, scripts, or something like that, and then it would be hard to find out what else will fail when we remove the entry. So oftentimes, these entries are just left alone.

Unlocking the Secret of Keys

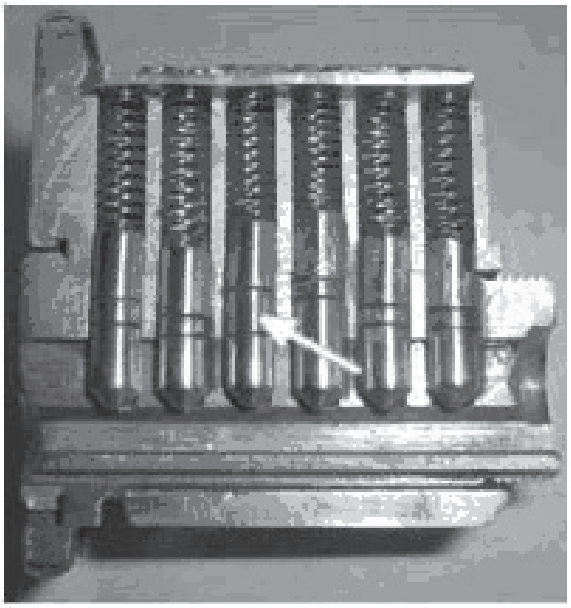
by James Hunter

As a teenager, I was always interested in locks and keys, and always wanted to learn more about them. My biggest leap forward came from garbage picking! Specifically, when walking home from school, I would go down a back laneway behind a hardware store and go through their garbage bin. One day I hit the jackpot! They were throwing out binders full of information about multiple keying systems.

A single key system has two pins in each cylinder and they must all line up in order for a key to open the lock. Only one key can open this lock.

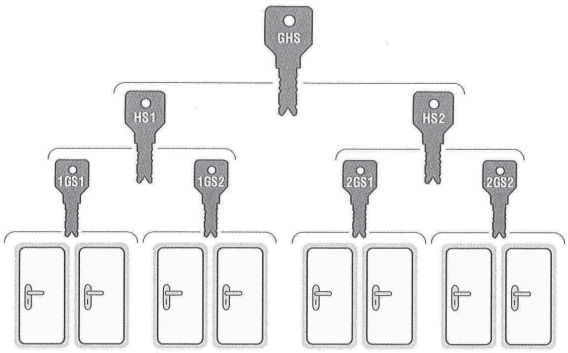


A master keying system is more flexible. The system works like this in our high school: There could be a key that only works on one door, or a key that works for a whole department such as the math department, or a key that works on all doors in the school. Typically, a department head would have the key for his department, and the principal would have a key for the whole school.



The binders had pictures of cut open locks, showing how the various keys worked. Each of six cylinders could have multiple pins in them. By filing down a key blank, I could "consume" all the pins (except the top one) and make a master key.

My first step was to acquire a key blank (or several). This can be done either from a hardware store if you tell them you are studying locksmithing or by mail order.



My next challenge was to secure a lock, make a master key, and return the lock. I did this one weekend, taking away a lock from a washroom and putting in a dummy lock.

Then I placed the lock in a vise and carefully removed each row of pins, making sure to keep the correct order. This is the most crucial step, since if you mix up the pins, the original key will not work. There has to be a pin at the top and a spring, then the remaining pins are on the bottom.

Once I'd done this, it was a simple matter to file the key for each pin location for the depth to make a master key. I then had to reassemble the lock with the pins in the proper order.

Once I had my key, I replaced the lock and they were none the wiser. This key let us discover and explore all kinds of cool areas of the school, such as the steam pipe tunnels in the subbasement and the disused greenhouse on the roof. We also explored crawl spaces in the ceiling of the auditorium where there was access to change light bulbs several stories up!

These days, all this information can be looked up on the web. One trick I especially like is opening the five pin button manual door locks by Simplex. A few tricks with these: The default code coming from the manufacturer is 2-4, 3 (press 2 and 4 at the same time, then 3). When installing the locks, often people fail to change the default code. The other trick is that on the most used buttons, the numbers are shiny from people pressing them so often. This reduces the combination of buttons to press and makes it easier to guess the combination.

ADVISORIES

Article Feedback

Dear *2600*:

In 34:4, there was an article "Nightmare on E Street - (Modem and Me Against the World)" by Emily Saunders. She is on the right path, but just needs to go a bit further. She is trying to manage her ISP's modem, a lost cause because it is lowest-bid hardware tied to her ISP.

Instead, she needs to take complete ownership of her connection. Yes, she still needs the ISP modem to get Internet, but the rest needs to be handled by her own network. Of course, that means having her own network equipment.

I recommend the Ubiquiti EdgeRouter X and a Ubiquiti wireless access point. The hardware is very powerful and less expensive than many of the common products found on the shelves of local retailers.

There is a wonderful configuration guide at https://github.com/mjp66/Ubiquiti/blob/master/Ubiquiti%20Home%20Network.pdf. Based on her article, she should be able to work her way through this guide. When she's done, she'll have a collection of independent networks providing everything from a secure wired connection for sensitive operations like banking to isolated wired and wireless connections for normal use, guest use, and IoT (Internet of Things) devices like "smart" thermostats and light bulbs. (Not that IoT is currently ready for safe deployment, but if you're going to risk it, this configuration guide will help dramatically reduce your exposure.)

Actually, by the time she successfully works through this guide, she'll be way ahead of most people, including many professionals.

Ron

We were blown away by the amount of responses this article generated. It reminded us of the old days when readers wrote a lot more letters about almost every article. We'd certainly like to see a return to that along with less of the standard (for today) two line comments. While we're waiting for that, here's another response.

Dear *2600*:

As a tech security professional, I wanted to offer a few thoughts in response to the "Nightmare on E Street."

1. Many of the "suspicious" websites you mentioned seeing in your logs may likely have been part of normal browsing. For example, y.timg.com is where YouTube caches some files. You also listed one of Netflix's servers. Normal browsing and using apps from your devices will access many servers and sites behind the scenes in order to function properly and quickly. This is probably not something to worry about.

2. The guest network can be turned off from Comcast using Comcast's account management website. You just need to disable "personal hotspot."

3. You seem to be seriously overcomplicating security and port blocking. The standard should be simply "block all incoming traffic." The firewall settings on your Comcast router will most certainly have this. Then you just whitelist allow ports on an as-needed basis. Whatever Comcast told you about "monitoring" certain ports is probably misleading and not relevant to what you are trying to accomplish. Just block all and let the firewall do its job. The other "unmonitored" ports you don't have access to are very likely blocked by default. These days they come with a fairly secure config out of the box. If you really need more settings to play with, look into buying your own wireless router that sits behind the modem on your network and install a custom firmware on it. Many free open source solutions are available that have tons of features. Then you can just keep all of your devices behind your own router and firewall as another layer of security and leave the cable modem open.

4. It sounds like the forward drop attempts and other log events you found mean the firewall is doing its job. It's doubtful that someone was targeting your own network with those. It's more likely that some machine is scanning networks for vulnerabilities, or randomly scanning IPs. Again, this is a fairly normal occurrence and not worth getting worked up about. That's why the firewall is enabled - so let it do what it's meant to do.

5. Sounds like you learned a lot, and have a great head for tech and security! I know all of this can be overwhelming at times, but that's why sometimes the best, simplest, and safest practice is to simply deny everything and then selectively enable what is necessary to function properly. Good luck to you!

Neil

Dear *2600*:

Regarding David's reply to Josephus' article (34:4):

Where to start.... The thing that jumped out at me was the example of social issues somehow not being about the aims of atheism. It actually, ironically, is a perfect example of how horribly wrong the idea of "sticking to what matters" actually is. Almost every social issue, in one manner or another, is driven by ideology which was derived from, influenced by, or is actively promoted from religion and the pulpit. It's not unlike someone pushing for prosecution of thefts who can't comprehend how the increase in them is directly correlated to an increase in poverty, and many similar issues. You can't attack the idea of God, and religion, then ignore all the stupid BS people do because God and religion spent thousands of years telling them they should do, act, or be, a certain way.

In point of fact, this is exactly why Atheism+ broke ranks with the so called "leadership" of the

New Atheists - because the leadership showed active hostility, and behaviors, towards members, not just because of "distractions" like social issues, but by exhibiting some of the very behaviors, ideologies, and responses *to* some of their own membership which are practically hallmarks of the very people they claim to be fighting against.

The real world consequences of growing up in a culture which is vastly more informed about how to act, who to trust, what you should and shouldn't do, and so on, *by* religions, matters. If anything, it actually matters far more than whether or not some joker worships a nonexistent god. Why? Because some person praying to nothing in their own home, or even a church, has absolutely no impact, at all, on anything, outside themselves. On the other hand, it also doesn't matter *at all* if it's Rush Limbaugh or Richard Dawkins that harasses women, claims everyone of a particular race/religion is a terrorist, or expresses a total lack of concerns about some social issue. But, what does matter is that *both* of them learned this view of the world from thousands of years of Christianity dominating the argument over who the enemies are, how women should be treated, and numerous other "distractions."

If a medical doctor cured a disease that was leaving you paralyzed, weak, scarred, and unable to work, then told you when it was all over, "Sorry about all the other stuff, but I can't be bothered to even find out if someone can help you with any of those other problems, and I sure as heck won't waste my time referring you to anyone that might know - it's a distraction," I imagine you would be kind of... annoyed, to say the least. And, yeah, I know this isn't the best example. I am sure others can come up with examples of cases where curing the "obvious" problem won't do a bloody thing to actually help the people affected by it.

This is why "only sticking to what really matters" is a completely losing proposition. Who are you, or anyone else, to define what "matters?" How do you know it doesn't or won't? What concrete evidence do you have that only dealing with this one single issue will solve anything at all? What does getting rid of the idea of gods do, if all the baggage just finds a new home, and new justifications - possibly in the mind of someone who fought to kill it?

In tech parlance, this is like the way the GOP thinks science and, by extension, everything else, including technology, should work: you need something, you just build it. Silly, wasteful stuff, like basic research, or watching/reading science fiction, or pretty much doing anything that is a "distraction" from the goal must be rejected as unnecessary, unimportant, and cost ineffective. Meanwhile, some fool reading a 50-year-old science fiction book is inventing the next new tech gadget. Or, one of those dang pesky "religious" people is actually fighting to put an end to a bad idea, which hurts everyone, while the glorious four horsemen of atheism whine about distractions.

Live in a bubble, devoid of annoying things that aren't important to hacking, if you want. Everyone else is gong to use those silly distractions to *create* the next new thing in security, technology, and so on. Funny how actual progress is never made by purists, who can't stand mission creep. It's almost like innovation, in everything, including tech, comes from dealing with the real world, not ignoring it.

Patrick

You make some excellent points here, particularly in your conclusion. Ignoring what's going on around you because you don't consider yourself an expert - or extending those feelings to others - is how entire populations get manipulated and taken advantage of, time and time again. Thanks for your thoughtful remarks, which we might never have seen if the letter writer you cite hadn't used atheism as an example of the point he was making in our last issue.

Dear *2600*:

I'd like to point out a glaring issue in 34:4, in "Don't You Have a Smart Watch Yet?" where The Cheshire Catalyst states that Chrome/Firefox's incognito/private windows provide "end-to-end" encryption with SSL. However, this is not accurate, and may provide readers with a false sense of security. If a web server doesn't support SSL/TLS, there is no way to achieve end-to-end encryption with the website. Those browser privacy features only prevent the saving of history and cookies and other data locally. Beyond that, they do not really provide any further protection, just as Chrome's incognito landing page states.

Additionally in 34:4, I'd like to respond to Emily Saunders' "Nightmare on E Street" with some suggestions. In the article, the terms "modem" and "router" are often interchanged. These are actually very different components, although they are often combined into one unit nowadays. However, you are not restricted to the router or modem that your ISP gives you. I'd even argue that an ISP has no business giving you a router. I would at least recommend buying your own router so that you can have complete control over it. The Xfinity modem/gateway can be set to "bridged mode," allowing your router to take over. If you want even more control, I highly recommend installing LEDE/OpenWRT/DD-WRT/Tomato on your router.

KO

Dear *2600*:

The philosophy of naming is interesting, but the article "Conventional Theory of Reference in Comparison to Programming Language" (34:4) contains a pretty major error about JavaScript, which is the example language they were using.

When it says, "Once a variable is given a name, you can still change the value of the variable, but you can never again define another variable using the same name. This means that when you use that variable's name in the program, there exists one and only one thing which it can refer to." Wrong! Here's some code to prove it

```
<html>
<body>
<script>
function whatisx()
{
        var x = "a string";
```

```
        return "Surprise!:
X is " + whatisx();

        function whatisx()
        {
                var x = 99;
                return(x);
        }
}

</script>
<div id"x"
onclick="alert(whatisx());"
style="background:red;
color:white;">Click to find
out what X is</div>

</body>
</html>
```

This little HTML file displays "Click to find out what X is" as a red bar on the screen and, when you click it, you call a function named "whatisx" which defines a variable that is a string, with the value "a string".

But then the fun begins. You would think that the function simply returns the value of itself (a snobbish way to say a function with the same name), which would result in a deathly recursive loop. But no, because we've redefined "whatisx" as a nested function that defines x as a number with the value 99. That value is returned instead.

In summary, if you take the dare and click the red bar on the screen, the pop up reports, "Surprise! X is 99". So, in this case, both the name of the function and the name of the variable were redefined, which is quite common practice in JavaScript programs.

Another error is that the article implies that you must define variables in JavaScript with the "var" keyword. Perhaps this should be the way that it works, but all you need to do is assign a value to an unknown variable name and it will become defined implicitly. I wish it wasn't this way, but it is. This leads to errors such as a programmer thinking they are using a variable that has been defined within the scope of their function, but they are actually using a global variable, such as the following really pointless code:

```
var supercalifragilicious = 0;
function
whatis_supercalifragilicious()
{
        var supercallifragilicious
= 100;

        for(i=0; i < 50; i++)

supercalifragilicious++;

        alert("Supercalifragilicious
is " + supercalifragilicious);
}
whatis_supercalifragilicious();
```

Because of the obvious (or maybe not so obvious) spelling error, the answer returned is 50 and not 150. This is particularly important if you tend to use certain values for counters (I use i, j, k, etc., a habit inherited from Fortran). If you don't declare the variables in each function, and don't realize you declared the counter as a global value, all functions will use the same variable, which could cause severe havoc in your code.

D1vr0c

We really enjoy these super helpful corrections.

Dear *2600*:

I sat in Barnes and Noble browsing through your mag when I stumbled on Emily Saunders' article. The words written on the pages struck many familiar chords with me. To my dismay, I can relate to a large portion of her story because I too have experienced very similar problems with my device(s). I have bad news for Emily, though. In March, I'll be at the four-year mark of first discovering my IT nightmares. Yes, plural. And I'm still having issues today.

After $60,000 donated to various so called "experts" (who were no more helpful than a bumper hitch on a Yugo), I threw up my hands and decided I'd be better served to try and help myself. Plus, I didn't have any more money to flush down the commode. But sadly, the only improvements from then until now is in direct correlation with my decision to live my life despite having an "Electronic S.T.D." coupled with a continued pursuit of knowledge and the determination to "mitigate."

So today, I'm my own CIO/CSO. Go Me! Don't forget, January 28th is World Privacy Day, lol. Lord, help me!

My reason for writing you is twofold. First, I must admit how disappointed I was to read her seven-page story which ended with no resolution, no commentary from the peanut gallery, no suggestions, *nadda!* Am I missing something? Is her story just another inside joke to all the Hackers On Planet Earth and only those included in your circle will get it? I surely hope not.

I feel fairly confident saying there's probably not very many people out here in the world of "noobs" who would go so far as to write to a hackers' magazine for answers or go to great lengths to, at the very least, try to educate ourselves enough to help ourselves. So the question remains as to what the rest of the world is doing for themselves when faced with a similar scenario? Nothing? Nada? Probably, and that's a scary thought! We all know in the world of cybersecurity and InfoSec, #IgnoranceIsNotBliss nor wise.

So, with all this said, I come to the second reason for reaching out. How would people like Emily go about helping themselves anyway? What are your thoughts on me putting together a group of compassionate people uniting for the greater cause of helping people like Emily? And myself, for that matter. C.P.U. - Compassionate People (or) Professionals United - has a nice ring to it, wouldn't you agree? I'd love to hear your thoughts about such an idea. Even better, I'd love to know if you or anyone you know would

be interested in collaborating on such a project and, if not, where would be a good place to find such people?

And lastly, would you pass my details on to Emily? I'd love to connect with her.

Mom

Let's go in order. First, we don't know how on earth you can interpret our printing someone's article as being an inside joke. Did you honestly expect us to insert our own notes and commentary into her article? That would be a bit presumptuous. In our magazine, articles speak for themselves. It's in the letters section (here) that we can have a back and forth discussion. As you may have seen by now, that article has inspired many helpful and supportive responses from the hacker community. Perhaps not assuming the worst of people would be the best way to get a dialogue going.

We think the idea of a support group is great. Anyone who would dismiss such an idea as a waste of time or a joke is really part of the problem. While there are certainly a bunch of ultra-paranoid technologically illiterate fearmongers all over the place, it's the height of ignorance and arrogance to paint everyone with the same broad brush. We reject any tendency to disparage those who don't rise to a particular level of tech-savviness or those who dare to question the use or value of the latest advancements. We need to be hearing these voices and responding to them as we would anyone else. Otherwise, we wind up with a very fractured community.

The people who go to our meetings - and certainly the people who go to HOPE conferences - would be ideal candidates to help put together such a support group. Remember, the hacker community in general takes the time to answer people's questions and demonstrate exactly how technology works - and how it doesn't. Why do you think we're always getting into so much trouble with the authorities? Spreading knowledge is a dangerous thing.

We're not a message board, so we don't generally pass notes from one reader to another. Because of the need for support here, we've made an exception and passed along your letter.

Dear *2600*:

I am not at all an expert in cybersecurity, so I learn all that I can.

I cannot come up with the cause for the writer's dilemma in the "Nightmare on E Street" article in the most recent issue (34:4).

The lack of conclusion has been getting to me. Is there anyone on your staff that was able to figure it out?

F. B.

We have our own nightmares to contend with. But we're thrilled that this has opened up a discussion and gotten greater minds than ours thinking about what's up. We're open to this, plus any other stories that are told in such a straightforward and rational way.

Dear *2600*:

Once again, I am humbled that you published my article about the intersectionality of hacking and politics in the Autumn 2017 issue (34:3). In retrospect, I wish I had fleshed the topic out more and explained how the case I used (Apple vs. FBI in the case of the San Bernardino shooter, Syed Farook) to illustrate my main point which is, to quote the late American historian Howard Zinn, "you can't be neutral on a moving train." I wrote the article not to discourage people in our community to be apolitical, but to remind - and awaken - our community that we do not live in a clean cut, hermetically sealed bubble. Real life is not as neat as many people in the magazine like to believe it is, but a mess where lines of various issues bleed into each other and have to be addressed. In other words, the hacking community - as a whole - must recognize that various "surface level issues" (such as immigration and pollution, to name a few) must be dealt with from a more systematic point of view.

I've read the letter David wrote on my piece and I respectfully disagree with his assertion that my article advocated "hacker groups (or individuals) to be political for unrelated things." That statement couldn't be further from the truth and it also shows that David (1) probably didn't read past the first part of my article and (2) didn't read the definition of intersectionality. His evidence that many groups bring in unrelated issues to their groups has some merit to it (i.e., "mission creep"), but his assertion about my position - by definition - is a straw man argument because it fails to address the proposition in question (intersectionality) by misrepresenting the opposing position. Furthermore, his argument that he can make "equally ridiculous" claims using intersectionality through some very "interesting" examples is an attempt to cheapen/discredit my argument by, in a sense, telling me to "drop it" because intersectionality is not for hackers, but for "groups that specialize in them" (this is formally called an argumentum ergo decedo, or traitorous critic fallacy). Given the current political climate in the United States - and around the world that hackers of all sexes, ethnicities, religious backgrounds, etc. are dealing with now (for example, the repeal of net neutrality in the U.S. and its effect on people of color) - we cannot sit on the sidelines when our voices and skills are needed the most.

At the end of the day, we are more than just hackers, Christians, queer folks, middle class, or any other identifier; we are human beings and we are, to paraphrase Jesus Christ, to love our neighbors as we love ourselves. As the hackers next door (so to speak), we as a community and/or individuals have an obligation to help people, not just with liberating them from the chains of Windows or locked iPhones, but through our actions on and offline.

You may think that my example about the repeal of net neutrality and its likely negative effect on black and brown communities in the U.S. has nothing to do with intersectionality. Please Google the topic and you will see what I mean. This particular issue is a timely and relevant reason why we cannot simply "stay in our lane" when it comes to being active participants in the hacking *and* in our local, state, and national communities.

I appreciate David sharing his critique, the other detractors of my article, supporters, and the staff at

2600 for publishing my letters and articles over the years.

Remember: Wake up! Stay woke! Get informed! Get moving!

Josephus

In the end, we will always listen to people who want to keep the conversation going. It's not about taking one side or another, but adding our perspective to a particular issue and listening to what others have to say. You find relevance where you seek it. And whenever we're told that we shouldn't be talking about certain subjects or that we should leave particular fields to the so-called experts, it only makes us want to delve into them deeper. That is the embodiment of the hacker spirit - going where you're told you have no business being.

Dear *2600*:

My opinion regarding "How to Get Nearly Free Travel from Scotrail" in the Autumn 2017 issue is that this is not a "broken policy," but a fully functional system of good customer service. The predictable response by the rail company to this kind of activity gaining popularity is to implement policy and possibly a technology that reduces the ability of employees to use their own judgment for providing good customer service and impersonalizing yet another aspect of our daily lives. In my mind, a justified consumer hack is a workaround to a problem with technology or policy where the implementation has resulted in an absurd inconvenience or breach of privacy or security to the customer. In this case the customer is well served by the judgment of the employees.

Scott

Yes, in this particular case the customer is being trusted by the human conductor and is allowed to travel based on a piece of paper that could easily be forged or otherwise invalid. And widespread abuse of this system will invariably lead to something less forgiving where the human conductor would have less leeway in the matter. Despite that, it's still an opening in the system that we feel compelled to point out.

Observations

Dear *2600*:

Hello all!

I am a lame ass hacker, mostly because I didn't discover your magazine until I was employed at Borders back in 2009. The random information I had gleaned and contributed to date was on BBS systems, then on IRC, and other places. When I read my first *2600* issue, I knew I had pissed away years of knowledge, learning, and collaboration.

The things I've read and learned from your publication have given me a better perspective on today's digital world, as well as a degree of education that has assisted me greatly in helping secure my own digital existence. The knowledge and lessons shared are invaluable, and I hope one day I come across something that can help others secure their digital selves.

This letter was prompted by getting in my (first ever brand newly owned) car and seeing that I was at 2600 miles. Thank you for doing what you've been doing, thanks to all of the collaborators, and may our knowledge and experience help stave off disaster in the days to come.

Halestorm

This is probably the most thoughtful reaction to an odometer reading that's ever been recorded. Thanks for the kind words. You are not a lame ass.

Dear *2600*:

Was in Phoenix for a business trip and this building was right beside my hotel. Too bad this wasn't my destination!

FF

Who says it wasn't? We're trying to keep our invites subtle, but clearly we need to step it up a notch.

Dear *2600*:

The recent FISA Court reauthorization of Section 702 for six more years points in the wrong direction for privacy rights. There will be stipulations that say 702 has to target non-American citizens, but there are numerous loopholes in this section that allows for American citizens to be targeted, such as if data is (so-called) accidentally collected or if a friend happens to be living overseas. Then that data gets swept up.

There also should be further technological manners in which American citizens whose data shouldn't be collected can be blocked or filtered in some manner. The reauthorization doesn't include any further privacy protections for such scenarios, either in 702 or in FISA overall. This really needs to be addressed.

Further privacy protections should be looked at in the future for Section 702 so Americans can have their civil liberties protected, which is crucial to having a healthy democracy. In fact, the FISA court really should be abolished, considering it's operating in secret with little transparency.

Bill

Just remember that it was both Republicans and Democrats who voted by a wide margin to extend this travesty until 2024. We wonder if anyone will remember what privacy is by then.

Dear *2600*:

I got myself some reading glasses so I could read the letters in the Winter 2017-2018 issue. I figured I was finally getting to that age, and my wife confirmed that I look like an old man with them on. Then I noticed I could read the letters without the glasses in the previous issue. Were my eyes getting better? Horsefeathers! I put the issues side by side and, by golly, it

turns out the Winter edition has a smaller font. Nice one guys! You really had me going!

Codger

Yeah, we've been messing with our readers like this for years. We're considering expanding into blurry images and double type for the future. It's part of our ongoing partnership with the reading glass lobbyists.

Dear *2600*:

Greetings in the majestic name of Jesus Christ!!!

Thirty years ago, I developed the trinary system from Mayan mathematics: 0 (shell shape), 1 (dot), 5 (bar) that led or will lead to quantum computers. England and America have it. Maybe a mystery country has it also. It's based on the fact that there are three charges in an atom - electricity and magnetism. The neutral charge counts. I wanted to honor the trinity.

Zero, One, Five
Hacker try that for size
Base twenty system
Experiment then list them
Goes right thru encryption
It's the trinary system
List them, list them, list them
I'm the creator I reveal at this time
So revealing should be no crime
Hack, hack, hack, my code to crack

Robert
Akron, OH

Do keep us updated on where this winds up going. (Just when you think you've seen it all.)

Random Questions

Dear *2600*:

One of the best articles I've read in *2600* was in your Autumn 1993 issue. The article was "Hacking Honesty Tests" by U.R. Source. Do you have any other articles about hacking personality or honesty tests, or has U.R. Source done any other work? I ask because personality/honesty tests are neglected areas in the hacking community, even though the potential for life-changing ramifications can be quite high (e.g. denying employment based on tests like the MMPI (Minnesota Multiphasic Personality Inventory), or using such a test as evidence in court).

GazetteMed

We're not aware of anything else of this nature in our pages. We would absolutely love to see more in this field. Any test can be hacked or compromised in some way. We exist to help point out how. Anyone with insight into this is welcome to send us an article.

Dear *2600*:

Your FTP server is back online again and working, and is greatly appreciated. Thanks.

Are you guys going to post "Eat Chicken and Die" to the *Brain Damage* archives at some point? Currently the only instance of it in its entirety is in a couple of different *Brain Damage* broadcasts recorded off-air, but it would be nice to hear it transcribed (in FLAC format?) from the first-generation tape rather than a third- or fourth-generation copy from an FM broadcast.

Mistman the Magnificent

You're referring to a production some of us made many years ago for an old radio show that we thought everyone had forgotten about. We'll see if we can track down a better copy.

Dear *2600*:

My name is Nicole Lewis and I am a blogger. I find your website very interesting and exciting, therefore I'd like to know if you would like to cooperate with me. I am open to any topic you might be interested in at the moment or I myself could suggest an article that would be suitable for your website. Currently I'm thinking about writing on Christmas related topics. Please, let me know what you think of it.

Nicole

For some reason we feel like we don't want to cooperate.

Dear *2600*:

I just wanted to know if I can use the title "2600" for a Manga that I'm writing. I figure I would ask your permission out of respect and get an OK from you or the manager. I just want the number 2600 as part of the title for my Manga that I'm writing, that's all. Please let me know.

Nate

The manager has stepped out, but we can try to help you. We don't own the number, so you certainly don't have to ask us if you can use it. If you claim to be representing the magazine in any way, that's a different conversation. For that, you'd need to speak to the supervisor.

Dear *2600*:

Do you know if any of the members of LoD/H are still active or at least available to communicate with? Thank you for your help,

Kraag

If by active you mean alive, then yes. We doubt members of the Legion of Doom/Hackers are still doing the same stuff, if that's what you're asking. You may occasionally find one of them at a conference or engaged in security operations. Or you might find one doing something completely different. We're not actively engaged in hunting them down, so that's really all we can tell you.

Dear *2600*:

help me teacher for google play giff card active
Thank you in advance
greetings

Ricky

Oh, Ricky, we do so want to help you. But we can't even understand what in holy hell you're asking us to do. And when did we become your teacher? We think maybe you're asking about how to activate a gift card, but we don't run Google so your guess would be as good as ours.

It truly is amazing what a steady stream of similar requests we get. There are most certainly an awful lot of confused people stumbling around out there.

Dear *2600*:

Hello, I was a *2600* subscriber a while back. Do you still have the free advertising policy for your subscribers?

Zachariah

Yes, we do, and it's a really good way to reach people who read the magazine, now and forever, as back issues are always being read for the first time by new people. Currently, the free Marketplace section appears on pages 62 and 63 of every issue along with instructions on what you need to do in order to submit your own free ad. Consider that major corporations would really like to be able to get the attention of our readers - and that we won't yield to them and potentially compromise the free expression we've got going on every last one of our pages. The only people we trust to keep in the spirit of this publication are the people who read it. That's why they're the only ones who get to advertise.

Dear *2600:*

Could someone please direct me to a domain registrar that I can purchase domains from via snail mail and a check? I have two domains in mind and, if possible, would like to own them for at least six years outright.

Bryan

This is far more challenging than it should be. We discovered that GoDaddy, in fact, allows for domains to be paid by check. But it has to be an electronic check using a company known as Certegy Check Services. "Certegy will create an electronic funds transfer ('EFT') or bank draft, which will be presented to your bank or financial institution for payment from your Checking Account. The Checking Account must be at a financial institution in the United States, and the check must be payable in U.S. Dollars." Ironically, when getting a refund from GoDaddy, you will receive a paper check in the mail.

As we don't have a lot of time to research this, it's possible a solution may be out there that we're not aware of. We can see the consternation of a registrar wondering why someone wanting to register an online service is unable to access payment options online. To us, that's not important. It's an option that should exist if there are any people out there who want it. And apparently there are.

Dear *2600:*

I listen to your Off The Hook show on WBAI as much as possible. My question to you is how can I find out if a person employed by a banking institution accessed my account data to get my cell phone number unauthorized, going back as much as several years ago. I never gave this person, or anyone else that knows or may know this person, my cell phone number. For me, there is a very important reason for finding this information out. Thank you for all your help.

Robert

You need to take a step back here and ask yourself (or tell us) why you think this person is responsible in the first place. You say you never gave them your cell phone number - or even anyone who might have known them. So why are they your primary suspect? And what exactly is the issue? We don't know what you mean by having your "cell phone number unauthorized," especially going back years. Does your phone not work? What company were you using? Have you asked them what the problem was?

We don't doubt that this is important to you, but there are holes in this scenario you could drive an Australian road train through. Please help us to understand what the problem is so we can try and come up with a solution.

Dear *2600*:

I'm sorry, I didn't get your reply. Could you please tell me if you are interested in cooperating with me?

Nicole

If we weren't interested before, we're super-not-interested now. In fact, we have decided to start actively working against you. We'll be in touch.

Dear *2600:*

I recently called a number from my cell phone and received this automated message: "Your wireless carrier does not allow calls to the number you're dialing. We're sorry for any inconvenience this may cause." I thought that there could be several reasons for this. One reason the number may be unable to accept calls is an issue with the service, like the phone lines are down. I also speculated that I received this message because the phone I was calling was out of its service area and was not enabled to receive out-of-area calls. Also, the phone service provider may have removed this phone's ability to receive calls in some circumstances: the user hasn't paid the bill, has gone far over the allotted amount of minutes, or used too much Internet service. The carrier may have also cut service if the user has reported the phone lost or stolen. I also figured that I was receiving this message because my specific number was being blocked, as some carriers allow users to block certain numbers. I made this analysis working under the assumption the number I was calling was a cell phone number. This assumption could be incorrect, as I could have been calling a VoIP line, a satellite phone, or a landline.

Considering all of the above, do you know of any workaround(s) that could possibly help me hack this restriction that I am encountering? Thanks for publishing the best magazine ever!

Lightning Tommy

*There are so many possibilities here that we could theorize for pages. So we'll try to whittle them down to a manageable set of likelihoods. It's unlikely you'd get this kind of a recording if the person's phone service was cut off, whether because of a natural disaster or due to them not paying the bill. You would likely get a recording saying that the number was temporarily unavailable or even something more specific that cites a reason for the failed call. It's rare to find a cell phone these days that stops working outside of its local area, but even you managed to do that, we doubt this is the recording you'd get. (It's also a common myth that incoming long distance calls cost more on a cell phone than any other call. The only time an incoming call is billed more is if the phone itself has been moved to a distant location.) As for being blocked, there are usually some clues when this is the case. Often, the phone will ring just once and go to voicemail. (If someone were manually rejecting your calls, the number of rings would vary. And if the phone had lost power, it likely wouldn't ring at all.) Blocking your number by dialing *67 before calling*

would also tell you if your phone number was being blocked, since the call would likely then go through as an unidentified number.

The telltale clue here is in the recording itself. "Your wireless carrier does not allow calls to the number you're dialing." That tells us the recording is related to your phone, not the person you're calling. After all, it knows you're dialing from a wireless phone, something that isn't normally differentiated on the called end. So the first thing you should do is see what happens when you call from a landline. It seems clear you wouldn't get the same recording. To analyze more, we need to know more about the number you're trying to reach. It sounds as if it has additional charges attached to it, which is why it apparently is blocked systemwide by your wireless carrier. You'd get a recording like this, for instance, if you were trying to call a premium number or equivalent on your cell phone, since those kinds of numbers are generally not allowed. We suspect something like that is what's happening here.

Dear *2600*:

I am writing this in hopes that your knowledgeable staff or readers will be able to lend me some guidance. My dilemma is this: I need to locate and remove the physical device on the motherboard that is responsible for Wi-Fi connectivity so that I can render my laptop a standalone unit incapable of sending or receiving data.

Thanks in advance.

Eric

Every laptop is different, so there's no one method of doing this. Some still have wireless cards while others require you to remove a chip, which is not something an inexperienced person should do. Of course, 90 percent of the experts will probably tell you to simply disable wireless in the settings and be satisfied with that. We assume that this isn't what you're looking for. It's also possible to buy laptops that have already had this done for various sensitive operations like the military. We're curious what other suggestions our readers may have.

Meeting Updates

Dear *2600*:

I have been running a community group for going on 20 years now. I am also (along with many of the members) a subscriber. I've been a reader since I was 12!

Somehow it never occurred to me that my meetings are on the first Friday of every month just like *2600* meetings! Given that the culture and event times are approximate, I'd like to give you the information and hope that you would also agree that this is a perfect fit for a *2600* meeting. I am flexible and willing to change up some of the rules I set in the event to accommodate any guidelines for *2600*. A lot of the information I have public is there to weed people out and it's worked. We've got an awesome culture.

I also noticed that many groups meet at coffee shops. If merging my current efforts are an issue, there is a Panera directly across the way. I can do a 5 pm to 7 pm meeting there before my event.

Michael

What you guys have going is a LAN party, which is great. But that isn't the same as a 2600 meeting despite the similarities in time. What you suggest regarding having a meeting prior definitely could work and might even bring more people to your existing event afterwards. You're also in the Washington DC vicinity, which currently doesn't have a meeting, so this could solve a few problems. But you also need to know that our meetings aren't dependent on other events or organizations, so whatever attendees decide to do afterwards is completely up to them. We hope to see this work out.

Dear *2600*:

Where are the meetings held in New York City? I'm a new member.

J

They moved fairly recently due to the old location being renovated and can now be found in the Atrium on 53rd Street and Third Avenue, a mere half block from the previous place. All are welcome. There is no membership required.

Dear *2600*:

We want to start our meetings in Astana, Kazakhstan. We are preparing to begin in December. We hope for your support for our first event. Please add our website to your list. http://2600.kz/.

Thank you!

Morty

This is pretty awesome. We really look forward to hearing how this one goes. (In fact, if you keep reading, you'll find out.)

Dear *2600*:

Today we had our first *2600* meeting in Champaign-Urbana, Illinois. We met at the food court of Lincoln Square Mall from 5 to 6:30 pm. We had 15 attendees, including one international attendee who was excited about the possibility of speaking at a future meeting. Everyone introduced themselves. BigEzy spoke about the history of Defcon and what it is like to attend. I spoke about canonicalization errors and how to use them to evade filters, access controls, detection, et cetera. We briefly discussed what attendees would like to see in our local security community, and how we would like *2600* meetings to go. A few people had to leave at 6:30 pm, and the rest of us went to get food at a nearby restaurant and wrapped up at around 8:45 pm.

We collected some ideas for future meetings. We are sorting out the venue; the restaurant in the food court was closed, so many of us were hungry, and some attendees seemed dissatisfied with the lack of visuals for my talk, or maybe the portable projector and screen setup for BigEzy's.

I plan to follow up with the folks who suggested changes regarding the venue and try to understand the issues so we can correct them and explore the alternative venues they suggested.

For next month, we will be in the food court again; the restaurant location is changing hands this month and I am planning to suggest that the new restaurant be open on the first Friday night.

To get the word out, we've tweeted and posted on Reddit, sent email to a local private security gathering, and are getting a mailing list configured so anyone interested can subscribe to get reminders and coordinate some meeting content in advance.

All in all, our first meeting was a good start.

asparagi

Wow. You guys really have it together. It should be noted that meetings aren't required to be this organized or have speakers and presentations. Most are simply like cocktail parties without the cocktails, where people mill and converse with various others over the course of the evening. But if this format works for you, that's fine with us. We just want to make sure that everyone is welcome, people are treated as equals, and that the basic guidelines on our website are followed. We hope you're able to solve the food issue - we can't imagine why places wouldn't be open on a Friday evening. Best of luck and congrats on the birth of what looks to be a great meeting.

Dear *2600*:

Our first meeting in Kazakhstan was great. There were 15 to 20 people.

Topics discussed included *2600* (the scene, the culture, and the underground), lockpicking (the basics, some practice with padlocks), and social engineering stories (identity theft, info about fraud and law, and how to be protected from social engineering).

We will keep having meetings every Friday at 8 pm local time.

morty

This is really quite impressive. We admire your enthusiasm. Officially, meetings take place on the first Friday, but you can have additional meetings as often or whenever you like. Please continue to keep us updated. Having this many people show up from the start is incredible and shows how meetings are in great demand all over the world. This is one that we really hope becomes a draw.

Dear *2600*:

Aloha *2600*. Would it be possible to get a POC for the Hilo *2600* group on Hawaii Island?

Mahalo.

Reynold

We don't give out anyone's personal info, not even for our own meetings. This is why we recommend that people wanting to be contacted get involved in a website for that meeting where this info is available. Running a Twitter account for a meeting is another way of making contact. Be sure to follow @2600Meetings.

Dear *2600*:

I was an organizer for the Space Coast (Melbourne) *2600* group, but have since moved to Tampa. We will be starting up a *2600* group on the first Friday of January.

Kevin

Thanks for forwarding us the details and update. We will start listing this meeting and wish you luck. Please also let us know if we should delist the Melbourne meeting.

Dear *2600*:

We are planning to host *2600* meetings at the Telephone Museum in Waltham, Massachusetts starting in January. Thanks!

The Telephone Museum

We think this is a great idea for a meeting location, even though you're fairly close to our Boston meeting. While existing in some form since 2012, this museum has only recently opened in its current location. The history on display here ought to be inspirational to attendees. We assume there's a place where people can congregate and talk freely, even if they have no interest in old phones. But it's definitely something we would encourage everyone to learn about.

Dear *2600*:

Hey there. On the meetings list, there's a listing for Sacramento: Hacker Lab, 1715 I St. I called today and they said "There's no such thing." Might you or someone know when it was posted on there or a contact for that space? I understand it's a hard task to keep these meetings up to date with current info, but any information you have would be very much appreciated. Thank you.

0xTrap

We've looked into this and found that these meetings are, in fact, defunct. We're sorry for the inconvenience. It's been removed from our listings. This is why it's so important for meetings to keep in regular touch with us, as things invariably change and we don't want to be spreading bad info. We only hope these people at the Hacker Lab weren't inundated with our attendees looking for a nonexistent meeting.

Dear *2600*:

Any contacts or confirmations about the *2600* meeting in Lima, Peru? I have tried to go the address listed: Barbilonia on Alcanfores 455, Miraflores, at the end of Tarata St. at 8 pm on the first Friday with no luck finding anyone so far (found the address, just no *2600*ers). Before trying to start a new group, I'd like to try and make contact with any existing peeps. Thanks!

Haven Hash

As that meeting has existed for more than a dozen years, we suggest trying at least one more time. Their Twitter handle is @2600Peru. If you can't reach them and continue to not see people there, we will have to assume it's been disbanded and will take the appropriate action. Regardless, we hope you help try and build or rejuvenate the community there.

Dear *2600*:

Happy New Year!

Please add to the list of meetings, the city of Petrozavodsk (Russia). Our site is ptz.2600.ru/ - Twitter: twitter.com/ptz2600.

ptz

You'll find your details in this issue. Russian meetings seem to be expanding with every issue. We think this is a very good thing.

Dear *2600*:

Concerning the Spokane *2600* meetings, the magazine has been correct, but your website (www.2600.com/meetings/list.html) lists a location that hasn't been used in about 15 years.

Also, we recently had to change locations because the current place now closes at 6:30 pm. Our new location is Starbuck's at 4727 N. Division Street. The

website is www.spokane2600.org.

Hawke

Holy crap! The URL of ours you cite is one we weren't even aware of. The correct URL for meetings is www.2600.com/meetings/mtg.html. The "list.html" you went to is from 1999 and must have been what we were using back then before changing our format. We apparently never erased that file. We don't know of any links to it on our site, but would love to know how you managed to find it. For the fun of it, we're going to keep this link alive for at least a little longer so readers can take a nostalgic trip down Memory Lane as we have just done. Thanks for inadvertently alerting us to this. And your new information should be updated in all the proper places.

Dear *2600*:

I am going to be moving to the New London/Groton area of Connecticut in the USA. What do you need me to do to start/facilitate *2600* meetings?

Matt

That's the spirit! Most people plan on things like school enrollment, garbage collection, and getting broadband installed when they move. Not our readers. The first priority is to make sure there's a decent meeting in the area.

To actually answer your question, all of the info you need can be found on our web page in the meetings section. Good luck.

Dear *2600*:

There are two Burger Kings at the Mall of America at two different food courts, one South, one North.

If I get to pick, I'd say the North one, but I think the South one has been there longer.

Christopher

We should just have two meetings in that damn mall since everything else is being duplicated. We hope to get an answer on which one is the correct one. In the meantime, please keep running back and forth until someone shows up at one of them. We suggest all newcomers to the Bloomington, Minnesota 2600 meeting do the same. If we haven't already gotten the attention of mall security, that oughta do it.

Dear *2600*:

The 2600.org/meetings/mtg.html website and the magazine back cover differ on our *2600* meeting location. Which location takes precedent? Is there a point of contact?

The magazine back cover lists: 2nd floor lounge, MIT, Stratton Student Center, MIT Building W20, Cambridge, MA.

The 2600.org lists Starbucks, The Garage, 36 JFK St., Cambridge, MA.

Is there a point of contact for either meeting or are these just meeting sites in case mutually interested people show up?

I might be a bit old for this stuff, but national politics are driving me to new places. Plus, I have an interest in the Circle of HOPE conference. But this is Boston in early February and it is a bit cold to be wandering around Cambridge.

I tried to visit the Stratton Student Center. I was a few minutes late and I did not stay long, but I asked everyone in the lounge if they knew anything about the *2600* meeting. None there were aware of any meeting. It was really cold tonight, so that may have just deterred others.

I can try the other Cambridge location at the Harvard Starbucks next month. It would be helpful if there is a meeting contact to RSVP.

marc

It seems that you were looking at our previous issue, not the current one. The Boston meeting location changed over the winter, so the autumn issue didn't have that information. (There was even a letter about this in our last issue.) Such changes don't happen all that often (it's because of scenarios like this that we discourage changing meeting locations unless absolutely necessary), but you should always look to our web page for the most current info, as it tends to get updated around a month before each issue comes out. We hope that by now you've found the meeting and also hope you're able to make it to HOPE. (We're sorry - there really wasn't any way to avoid three hopes in that last sentence.)

Dear *2600*:

We went to the *2600* meeting in Calgary on February 2nd and no one showed up at the appointed location and time. Do you know if the meeting has moved?

Philip

Not to our knowledge. But if we receive more such reports, the meeting will move out of our listings. That is, unless the people who show up and don't see anyone else keep the meeting going in that or a new location.

Critique

Dear *2600*:

I have submitted an article but I cannot get any word on whether you want to publish it or not. I have to say the communications from your side are very poor.

C

We simply don't have the resources to immediately get back to people on whether or not their article submissions will be used. We do send out notices if we're going to use them, but that process can take a few weeks, especially between issues. We know other magazines do things differently, but we don't do a lot of things like other magazines. We do understand your frustration, however, and will try to be speedier in the future. Thanks for writing.

Dear *2600:*

"More recently, when a deplorable..." ("Acts of Courage," page 4, Autumn 2017, 34:3)

Apparently, you guys are so lame and so tone-deaf that you imagine that you can win hearts and minds by stealing themes from Hillary Clinton.

It's worthwhile for your readers to remember that not long ago, she (and you) were applauding the Harvey Weinstein crowd, as model "progressive" citizens.

We will defeat you the same way we are defeating the NFL... by standing back and watching whilst you paint yourself into a corner. And then cutting off your EBT card.

Judah Maccabee changed the world. Theodore Herzl changed the world. Eliezer Ben-Yehuda changed the world. But apparently, Emmanuel Goldstein can't even compose a plagiarism-free editorial.

Lifetime Subscriber

Let's see if we've got this straight. If we use a word that a politician once used, we become guilty of plagiarism? The word "deplorable" forever belongs to Hillary Clinton and any future use of it is simply channeling her? This is a really weird thing to take offense at, since we were using the word to describe neo-Nazis. It seemed to fit. We could have also tried "despicable" but then we probably would have been accused of ripping off Daffy Duck.

As for applauding Harvey Weinstein, we don't know where you get your facts. We led a demonstration against his company back in 1998, albeit for different reasons. Suffice to say, we haven't had any statues of him or his ilk in our offices.

As for the NFL, you do know they sued us once? We're not exactly fans. However, we do believe individuals deserve credit and respect when they make statements in defense of liberty and freedom, even when (in fact, especially when) it's discouraged by those in charge. Thanks for reminding us. (We never even brought this up, incidentally.)

As for the rest of your diatribe, we'll just let it stand on its own. It's hard to imagine how an editorial condemning racism and praising the actions of those who took a stand against it could provoke more of a negative response from our readership than a positive one, but that is the sad reality. Yours was actually one of the more civil ones we received.

Following Up

Dear *2600*:

This is a message to Phototrope, the subscriber looking to sell his father's old phreaking hardware. I'd love to purchase it! Email me, maybe we can set something up.

Thanks *2600!*

Tim

We have forwarded your message. This is not something we usually do but you caught us on a good day. For this kind of thing, we strongly recommend using our free Marketplace service.

Dear *2600*:

You guys published an issue with my son on the back cover sometime around 2008. He's the baby boy eating an issue. He's a bit older now and wants an issue, but I seem to have lost my copies. Could you let me know which issue I'm referring to so that I can order more?

Thank you!

Nick

Only if he promises not to eat this one. (We've been in touch on this and have sent out the issue in question, which was Spring 2009.)

Issues

Dear *2600*:

I have been a paid subscriber forever it seems. I keep getting charged, but I realized I am not getting the magazine.

I do not know how I signed up, it was so long ago.

How do I get "back into the know," get the legacy mags, and pick up the new ones? Thanks.

Matt

Well, this is certainly not something we want to see. We don't have an auto-renew feature for our paper edition, so we don't know how you could possibly be getting charged unless you're renewing on your own and still not getting the issues. We've tried reaching out to you about this but haven't heard back, so it's really difficult for us to figure out what's going on. Is it possible you have us mixed up with another magazine?

Dear *2600*:

I have been your faithful reader and follower for many years. All of it started back around 1994 (I think) when I had picked up a *New York Times* issue on my flight from JFK to Europe and started to read an article about Kevin, the "monster criminal" who could launch nukes by phone. Ever since then, I have always been wondering why and how the connection between hacker and criminal came about.

A few days ago, the bright light shined on me as I read a magazine based out of South Africa called *Very Interesting* (Issue 39, February 2018). In one of the articles about cybercrime, the concluding paragraph explained it all. It stated:

"It's a glorifying myth to think of it as 'genius hackers versus plodding security companies.' Instead, if we think of hackers like ordinary criminals and guard against them in the same way, then there's no reason why society, including the public, the media, companies and governments, cannot keep cybercrime under control."

Just food for thought and would love to hear your opinion. Is your blood boiling as is mine?

Keep up the fight - you're doing a hell of a job. Otherwise, all of us "curious thinkers" would be locked up already.

Ross

Ostrava, Czech Republic

It's nothing we haven't heard many times throughout the years, and not just with regard to hackers. The key to treating anyone unjustly is to demonize them. Once you truly believe they're a threat, then you can justify any actions taken against them, even if those actions would normally go against your values. It's one of the oldest con games in the book and it usually is an attempt to hide the overall ignorance of the person casting aspersions. Don't let it bother you too much; this kind of thing will always be with us. Instead, look to ways of countering this perception among those you know and can communicate with. It's an unfortunate fact that people will start to fit into these expectations if they're programmed to believe that this is all a hacker can be. We need to be sure to reach them before that happens.

Dear *2600*:

Spotted! Taken during a blizzard in Montreal while walking between bars. -20 Fahrenheit!

I'm not a great French speaker, but I'm told the screen literally means "please pick up" - sounds like a

cry for attention, poor lil dude!

Aaron

We don't print this to mock or ridicule, but to merely express our frustration at something which happens far too frequently. We get amazing descriptions of payphones, back cover photos, and sometimes even entire articles, but for whatever reason, nothing is attached and we wind up in deep, dark disappointment. Please don't let this happen to your next submission. Double-check that you have in fact attached what you want to send to us. We hope one day to be able to see what sounds like a truly awesome payphone in the snow.

Discoveries

Dear *2600*:

I am an avid reader of your interesting magazine which I came upon, either by accident or miracle. How I found you is also very interesting because it happened suddenly and without any warning.

So how did it happen? Well, I am a frequent visitor at one of the few Barnes and Noble bookstores here in Portland, Oregon, something that my wife and I get involved in almost every weekend. One day as I was passing by the computers and technology section, somehow I felt the need to look into the smaller magazines they place in there when I saw this thing that looked just like a brochure with the *2600* logo and a coordinates graphic on the front cover, and I just could not resist the desire to look into it. And sure enough, I was hooked right there because I just could not stop browsing and reading all of the interesting articles about technology, hacking, and code written all over it. And I knew at that time that *2600 Magazine* was going to be one of my favorites to read in my spare time.

So I am just sending these lines to congratulate you and to thank you for your efforts to make all this wonderful information of the amazing world of computers, hacking, coding, "the Internet of Things," and anything else that lives in it, available to the masses.

chomito44

It's always great to hear of new addicts. Often it takes an act of curiosity or exploration to track us down in the first place. Once those pages are opened, it's very hard to go back.

Dear *2600*:

Hacking isn't just a computer thing; it's a way of thinking and there is nothing "new" about it. I'd like to share a portion from Edward Frenkel's book *Love And Math*. It refers to Évariste Galois, a genius mathematician who lived from 1811 to 1832. He died at age 20. "Galois' work is a great example of the power of a mathematical insight. Galois did not solve the problem of finding a formula for the solution of polynomial equations in the sense in which it was understood. He *hacked* the problem! 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." That is the essence of hacking. Frenkel's book explores the Langlands program - the grand unifying theory of mathematics - bringing together Galois' algebras with harmonic analysis, which is intertwining ideas that are fundamental to the science of encryption and so much more. The beauty of this book is that it explains this incredible complex world of high level mathematics in a way that you can grasp its beauty even if not fully understanding it.

SideFx

Galois' story is an incredible one, both inspirational and sad. It does indeed parallel the challenges, frustrations, and accomplishments so many in the hacker world experience. We can only hope that hearing such tales will inspire creative minds to never stop thinking and experimenting, no matter how much the people around you discourage it. Believing in oneself and knowing there's this huge community of people going through similar things are vital in moving forward. We encourage our readers to keep looking for similar stories of hackers in history. They're everywhere.

Dear *2600*:

I have a $45 a month plan (GoPhone, AT&T, pay as you go, not auto refill). After the first three gigs, high speed is exceeded (usually in about a week), and I get 128 kbps for the remaining 30-day term. I found a hack that makes the 128 kbps still work *after* the 30 day term. *For free!*

I found this out when I couldn't afford to refill my $45 monthly plan. I had left my mobile hotspot on the night before on my ZTE Maven ($40 cell phone) and my kid was on the Internet in the morning. I checked his connection and, sure enough, he was connected using my phone. I said, "This is impossible! My 30-day term has expired!" I tried to use the phone, which turned off the mobile hotspot. The phone didn't work because the plan had indeed expired. Alas, when I tried to turn on the mobile hotspot again, it didn't work.

I couldn't shake this feeling that had I left my mobile hotspot on, it would have worked forever, for free. So the next month I purposely did not refill my $45 plan and I left the hotspot on. It is Day 3, and we still have Wi-Fi. I have not turned off my phone or mobile hotspot. This is really confusing me. I looked up my GoPhone account online and saw that my plan expired three days ago. So there is obviously a kink in the way AT&T feeds people data. Unfortunately, I can't use my phone to make a call; I know I'll break the free data connection and refill just to make a call.

My next question is, if I had not used the three gigs of high speed data, would I have free, unlimited high speed data now? AT&T obviously doesn't know I'm still using my mobile hotspot, so how can they make me high speed or low speed? This is worth a try next month. Does anyone know why my data/mobile hotspot still worked? Have I discovered a way for people all over the world to finally get free Wi-Fi?

sueicloud

Fascinating. Do keep us informed. And thanks for being a true hacker. Letting the world know about this may very well portend the end of its existence. But this will inspire people to try all sorts of different approaches to find more bugs and oddities. In the end, we wind up with systems we understand far better.

EFFecting Digital Freedom

Remembering John Perry Barlow

by Jason Kelley

John Perry Barlow's vision, writing, and hopefulness helped set the tone for the Internet that we have today. A lot of people in the late 1980s, including those in power in the government and in corporations, saw "cyberspace" as simply a toy or hobby. Barlow, then a rancher and sometime lyricist for The Grateful Dead, realized in contrast that it offered something much greater. Connecting to online communities like The WELL from his ranch in Wyoming, he saw the Internet as a place where physical distance and even physical bodies no longer mattered, and recognized that the technology could create a kind of connection that humans had been craving.

He saw communities developing around these "frontier villages," and he also saw that the early adopters of the Internet - a group made up mostly of engineers, coders, and people hacking their way around Cyberspace - needed allies in the civil liberties world. As governments and corporations began to take the Internet more seriously, they also began to clamp down on what it could be used to do and chill the ways it might be used in the future. Early raids on BBS users by the Secret Service and the FBI, and shutdowns of online newsletters like Phrack, made it clear to Barlow that the Internet's promise could be crushed if it had no defenders. In his words, it "could be a fundamental place of freedom, where voices long silenced could find an audience," or a place where government "limited free speech, conducted improper seizure of equipment and data, used undue force, and generally conducted itself in a fashion which is arbitrary, oppressive, and unconstitutional."

Hackers were under scrutiny at the moment, but the Internet, in Barlow's mind, could offer so much to humanity that the promise, and the danger, applied to far more than just those at the fringes. He also understood that "hacking" wasn't something to be feared, but something intrinsic to human nature: "Far more than just opposable thumbs, upright posture, or excess cranial capacity, human beings are set apart from all other species by an itch, a hard-wired dissatisfaction. Computer hacking is just the latest in a series of quests that started with fire hacking. Hacking is also a collective enterprise. It brings to our joint endeavors the simultaneity that other collective organisms... take for granted. This is important, because combined with our itch to probe is a need to connect."

As soon as Barlow understood the threat posed by these shutdowns and raids, he got to work. As he put it, during a lengthy 1990 interview with 2600's radio program Off The Hook (then called The Fifth Corner), "Whenever you've got an agency of the government that's out of control, the best thing you can do is invoke another part of the government against it. And in this case, fortunately, you've got the judiciary." He co-founded the Electronic Frontier Foundation with Mitch Kapor that year to defend the Internet from and explain it to the vast majority who didn't understand it, and to talk about it as a place of freedom so we might have a hope of building it that way. In his most well-known essay, the "A Declaration of the Independence of Cyberspace," he dreamt of a stateless frontier where "all may enter without privilege or prejudice accorded by race, economic power, military force, or station of birth" and "where anyone, anywhere may express his or her beliefs, no matter how singular, without fear of being coerced into silence or conformity."

We may not have the web Barlow wanted, but much of the freedom we do have there is thanks in no small part to his philosophy and effort throughout the Internet's early days. He helped carve out a space for people to fulfill that dream, and inspired them to do so. It's difficult to overstate the significance of those battles: Freeing encryption from government control and establishing that the Internet was a place of free speech - and that code itself was protected by the First Amendment - were fundamental to the Internet's growth and to protecting its users, especially the curious tinkerers and hackers on the digital frontier. Security researchers, programmers, and developers exploring cutting-edge technology would likely not be protected in their work without the precedents and ideas Barlow helped set in motion. He saw that as technology changed; education, legal defense, and providing policy advice to people considering computer crime legislation would be necessary, and it continues to be a large part of EFF's work.

Of course, it's no longer only the civil liberties of the early adopters that must be protected. Barlow saw that his high hopes might not come to pass, but he made a conscious decision to focus on the Internet's potential, and to consistently remind us that we were in control, writing: "I knew it's also true that a good way to invent the future is to predict it. So I predicted Utopia, hoping to give Liberty a running start before the laws of Moore and Metcalfe delivered up what Ed Snowden now correctly calls 'turn-key totalitarianism.'" His goal was to ensure that a humane, supportive, connected Internet could get that running start.

His wisdom will be instrumental in keeping up the fight for decades to come, and he will always be an integral part of EFF. He helped set the tone for the Internet, and inspires everyone who fights for a better one today.

Read our collection of Barlows writings at `https://www.eff.org/john-perry-➥barlow`.

Hacking Our Attitudes

The Key to Being a Better Attitude Trumper

by Dufu

After reading Jeffrey H. MacLachlan's "VR Trumpers" article in 34:2, I realized I finally had to say something about something. Are you ready for something?

There is a trend in the USA that is troubling to me lately. By lately, I will betray my age a bit by saying that it has been happening with increasing frequency since at least the Bill Clinton years, although I was less aware of it back then. I see an increasing ability for people to voice their apparent hatred for a political candidate through means that blatantly disrespect fellow human beings - all without care for the person standing next to them or with remorse if they offend someone else. I am convinced that there are smart, logical human beings all around me. Now, to be fair, there is also an abundance of not-so-smart, not-so-logical folks too, but let's just ignore them for the purposes of this write-up. They self-identify without the need for politics quite often.

Having lived in a suburb of New York City for virtually all my life, I have met highly educated people who lean strongly to the liberal side of politics quite often. I have also met highly educated people who lean strongly to the conservative side of politics. Being a low-level, but surprisingly diverse location, world traveler, I have met the same basic groups of people around the world in places like Hawaii, West Africa, Europe, and Arkansas. So this trend had me troubled.... Here is why: I have some very strong political opinions. They are based on my personal logic circuits and life learnings. Yet, taking an opinion to heart put me squarely at odds with the folks on the other side of the political fence. The only thing I could identify that the left and right have in common is our distrust of those in powerful positions who lack accountability and transparency. A strong dislike or hatred of those who act poorly in their dealings on our behalf as supposed public servants. Ptooey!

So about three or four years ago, I finally spent some time and effort and hacked my own thought processes and response methods. I took on my mental conflict and processed it until I found a place where I could handle the input and output without finding a conflict or error routine if you will. I budgeted around 32k of mental bitcoin-ish energy units and would estimate that it likely took closer to 65536k to get comfortably into the routines without errors popping up. I realized something. Every time I take an absolute stance on a debatable issue in politics, I am essentially saying I am smarter than the guy or gal next to me who disagrees with me. I was disrespecting the folks on the other side of the fence by essentially calling them idiots! It was clear to me that if I came out and took an immovable, closed-minded stance to anything that someone else disagreed with, I was likely sending a message that said, "My logic paths and knowledge is greater than yours!" or "I'm too closed-minded to even consider your ability to process this issue properly."

This realization helped me to share my opinions with more respect for the fellow human beings around me. It has made me less confrontational and has helped to spur on some great discussions where I learned something from someone else or was able to teach something to someone. Had I not changed my stance, we would simply have been arguing, rather than discussing and learning. Hate would have won and the world would be a worse place because of it.

I say all of this for two reasons. First, *2600* seems to be clearly picking a political side these days. That is troubling to me since I have been reading the writings of fellow hackers in *2600* since the days of 300 baud communications. It is also troubling because when a publication clearly picks a side, rather than addressing issues only, it ostracizes an entire group of people. In this case, *2600* is separating itself slowly from the seasoned, old school hackers who still follow the publication as they are more likely to lean right.

Second, and this supports the first reason more than anything else, is that I read the "VR

Trumpers" article twice looking for true hacker content. While I did find some keywords and phrases which thinly veiled the author's true motivation for writing the article, it was clear to me that the piece was written simply to bash Trump. Now that's fine and good if Jeffrey wants to do that, but I cannot believe that *2600* is allowing themselves and their content to be diluted and polarized in such a way! I have read other anti-Trump posts and been fine with them as clearly labeled opinion pieces, but when an "article" comes across my eyes that is supposed to inject information and knowledge into me and it's clearly designed as a smoke screen for a political bitching session, I'm not a happy subscriber.

I want to encourage *2600* and its readers to stick to the issues and the technology. Opinions can be shared, of course! When politicians act like idiots, there may be no better way to hold these oxygen thieves accountable than to expose their actions and policies and weaknesses. But to write an article that read a whole lot more like fiction than anything else simply to bash a candidate, and to do so with no perceivable useful information on the political front... that is just simply a waste of paper (or electrons) in my opinion.

Now here comes the surprise! I'm not a Trump supporter. To be fair, I wasn't an Obama supporter either. Frankly, my political thoughts were not well formed prior to Obama (my head was in the sand, I'm sorry to say). I can't say I have really supported anyone. What I support now is clear, though. If you talk to me, I'll do my best to respectfully discuss the importance of returning to voting for politicians who realize it is an honor to serve in office. A return to a place where politicians expect less than the average citizen they represent as true public servants. I'd love to see politicians supported who truly sacrifice to be in office because their heart is in the right place rather than in the next electoral cycle. A return to a place where more power and influence automatically means more transparency and accountability. And more than anything else, a return to a place were we can respect our friends and neighbors even if they have a different opinion than ours.

Oh, and maybe *2600* can refrain from publishing articles that are not about making progress, exposing lies, hardening systems through pen testing, and things such as that. Jeffrey's article was nothing more than left wing entertainment and *maybe* a look into the capabilities of VR units for those unfamiliar with the scene. In all the years of reading every last issue from the first to the current, I've never been disappointed by such an off focus article. No offense meant Jeffrey, but your writings were better suited for a different publication, I guess.

Is *2600* lacking articles and writers that badly? One part of me says "I hope not" while the other part of me says "Hopefully to the degree that they actually publish this article."

You can become a better person by reprogramming your response methods even if you don't change your opinions much. Hack your attitudes. Hack your life! Hack it all!!! If you are not sure of your weaknesses that need to be hacked, ask some close friends for an honest review of your attitudes and habits - especially your friends who you disagree with politically. You are wise enough to keep some of those around, right?

HISTORIC HACKING

by Huntech

A lot has happened in the computer world in my lifetime! This article is meant to show you just how far computers have come.

I'm now 57 years old and in 1975/6 I was in grade 10 in high school. At that time, a forward-thinking teacher had created a computer science course at the high school. It's hard to believe, but we didn't have any computers in the class - we were using the Board of Education's mainframe computer (which was located at the Board office several miles away). Note that this computer did not have a screen or keyboard for us to use: we typed our programs using a huge keypunch machine onto keypunch cards.

Keypunch machine

We had three keypunch machines in a separate room down a hallway at the school. Here's how it worked: You would first write out your program with pencil and paper. When you thought it would do what it was supposed to, you would type it onto punch cards: one card per line. So, for example, a card might have the text: "for I = 1 to 10". Note that we were programming in the evil language Fortran which is used mostly for mathematical calculations.

You would put a stack of blank cards in one slot on the machine, it would feed in a card, you would type your line on the keyboard and it would punch the card. If you made a typing mistake at this time, you would have to re-type the card. Once the program was complete you would have a stack of cards held together with an elastic.

Now to run the program, it would have to be couriered to the board office (which took one or two days), they would have to run the program (which took one or two days), and the results would have to be couriered back to the school (which took one or two days). So turnaround time was usually one week.

Punch card

Normally what would happen is that there would be a typing error that you didn't catch (for example, the line above might have been typed in as "Fer I = 1 to 10"). This would result in an

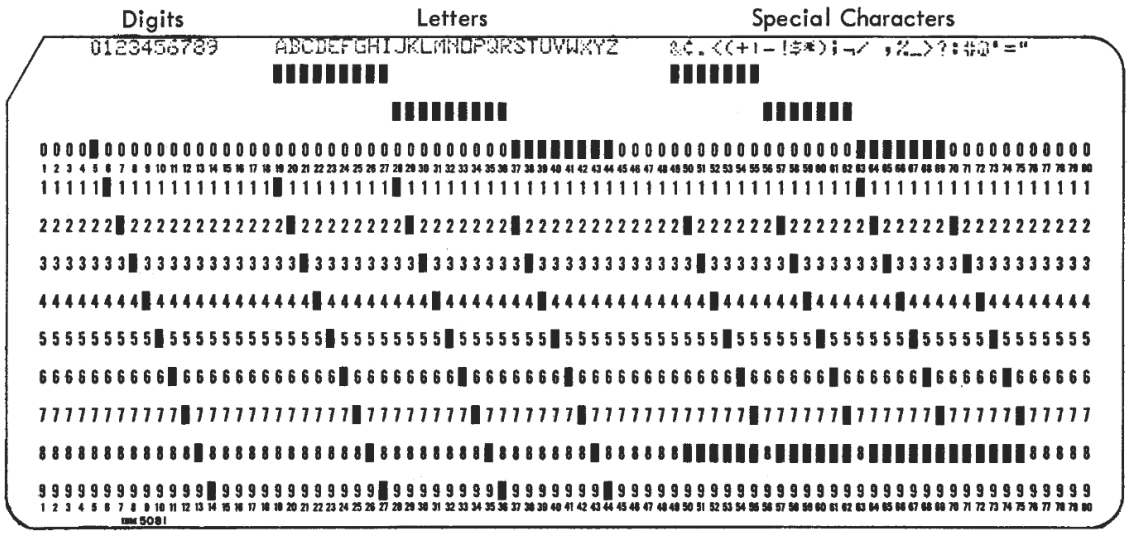
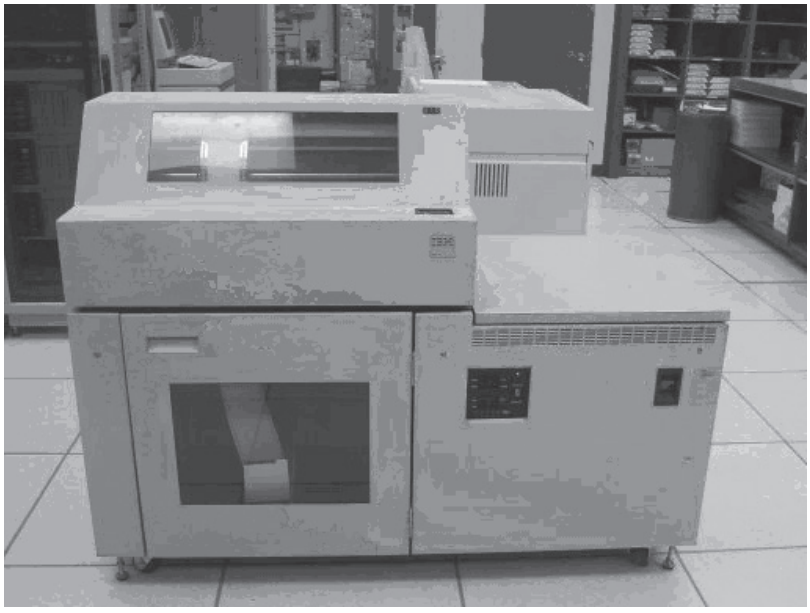


Figure 4. Card Codes and Graphics for 64-Character Set

error and you would have to fix the error and re-submit the cards (which would take a week). Then you would likely have a logic error (for example, forgetting to define a variable or having an endless loop). This would require you to fix the error and re-submit the cards (which would take a week).

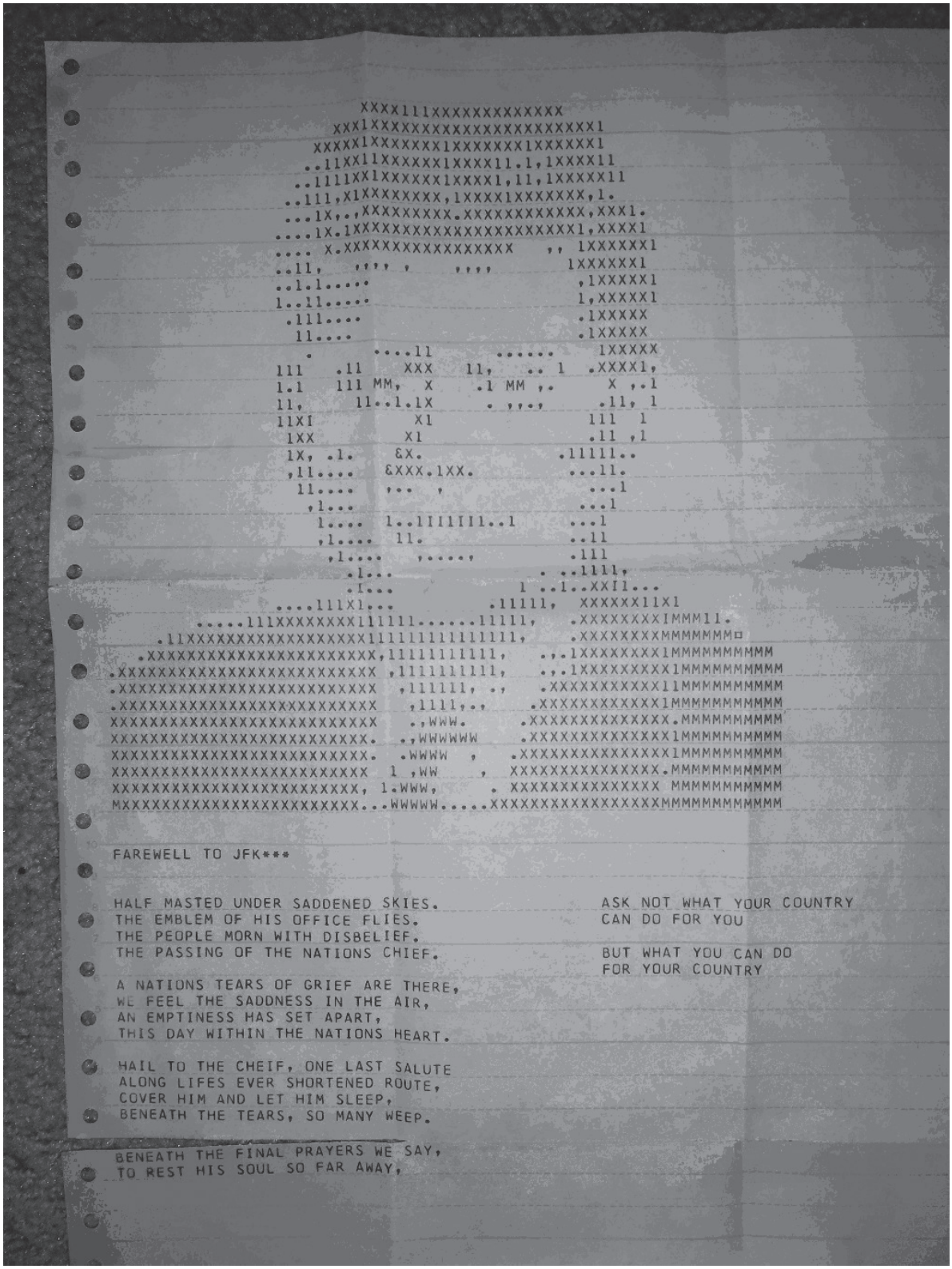
So it could take a few months to get a simple program to work properly. To run the programs at the board office, they would put your stack of cards into a hopper which would quickly read them all and the results would come out on a mainframe printer. Typically, several pages of text would print. God help you if you had an endless loop which could chew through tons of pages of paper.

Now being forward thinking students, we got permission to go to the board office and run the programs ourselves. So in one evening, we were able to run all the assignments for the whole year! While at the board's computer room, we noticed that there was a printer loaded with the paper for the mid-term report cards. This was a golden opportunity for us to create look-alike report cards with teachers' nicknames, great marks for ourselves, and comments such as "best student in class."



Mainframe printer

From that start, we found we could go the university's computer room and run programs there, so we had quite a lot of fun in those days running printer art. This was made by using a series of characters (and white space) to print a picture that from afar would look like something (such as a nude female or a picture of President Kennedy).



FAREWELL TO JFK***

HALF MASTED UNDER SADDENED SKIES.
THE EMBLEM OF HIS OFFICE FLIES.
THE PEOPLE MORN WITH DISBELIEF.
THE PASSING OF THE NATIONS CHIEF.

A NATIONS TEARS OF GRIEF ARE THERE,
WE FEEL THE SADDNESS IN THE AIR,
AN EMPTINESS HAS SET APART,
THIS DAY WITHIN THE NATIONS HEART.

HAIL TO THE CHEIF, ONE LAST SALUTE
ALONG LIFES EVER SHORTENED ROUTE,
COVER HIM AND LET HIM SLEEP,
BENEATH THE TEARS, SO MANY WEEP.

BENEATH THE FINAL PRAYERS WE SAY,
TO REST HIS SOUL SO FAR AWAY,

ASK NOT WHAT YOUR COUNTRY
CAN DO FOR YOU

BUT WHAT YOU CAN DO
FOR YOUR COUNTRY

Printer art

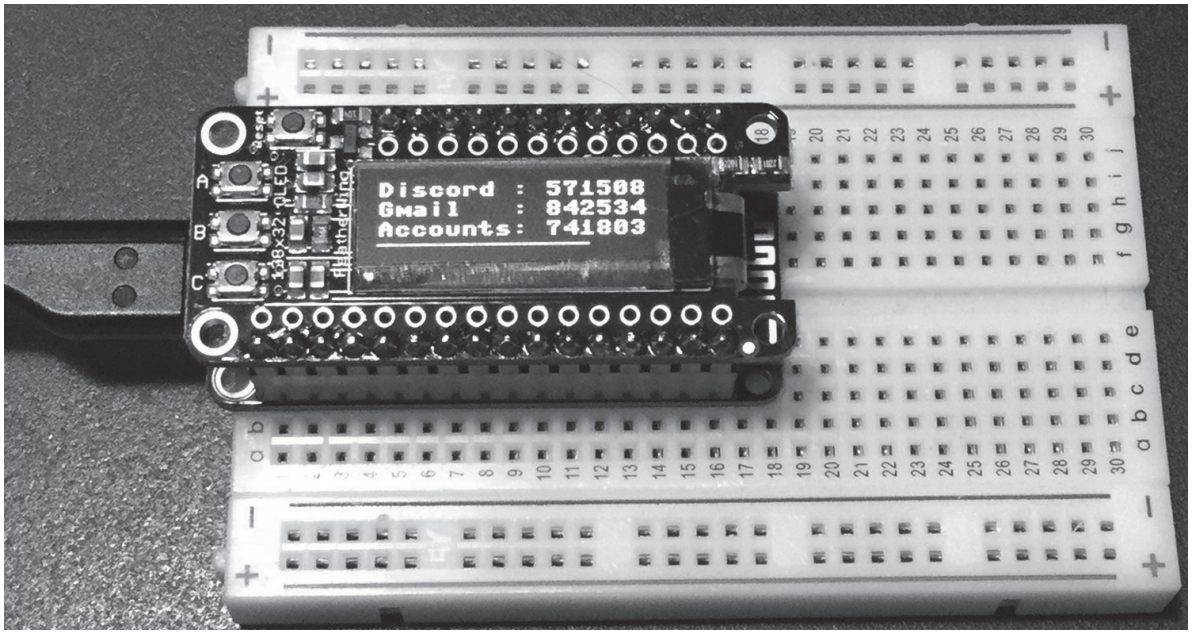
At university, I started by using the mainframe, but in my second year they had a system of terminals that fed data to the mainframe so you could work at a screen/keyboard. Later that year (1981), micro computers started showing up. We did our first work on punch cards at university. One time I had a large stack of cards in my backpack that got wet when I rode my bike home. They wouldn't read through the machine. Luckily, they had machines that could read damaged cards and make a copy of them.

The young people of today are very lucky to live in a time when computer technology is advancing so quickly and there are a lot more things that can be done on the computer.

CITIZEN ENGINEER

by ladyada@alum.mit.edu and fill@2600.com

Make Your Own Two-Factor Authentication Hardware Device



What is TOTP?

Having two-factor authentication on all your accounts is a good way to keep your data more secure. With two-factor authentication logins, not only is a username and password needed, but also a one-time-use code. There are a few different ways to get that code, such as by email, phone, or SMS. But my favorite way is to do it is via a "Google Authenticator" time-based OTP (one time password), also known as a TOTP.

Using an app on your phone like Authy or Authenticator, you set up a secret given to you by the service, then every 30 seconds a new code is generated for you. What's extra nice is that the Google Authenticator protocol is supported by just about every service and phone/tablet.

In our hacker household, Ladyada does not own a phone. A cell-phone jammer, yes. A cell phone, no. Fill is essentially the phone as needed or a tablet can be used. Why purchase a phone just for two-factor authentication?

Luckily for us, the Google Authenticator protocol is really simple. You just need to be able to know the current time, and run a SHA1 hash with both a known secret (given to you by the online service) and the UNIX epoch time in seconds divided by 30 (so you have plenty of time between code-updates).

We built a simple device that does nothing but generate TOTPs, using CircuitPython - it's Python for microcontrollers! It uses a Feather ESP8266, which has Wi-Fi so it can connect to NTP to get the current time on startup, and a Feather OLED to display text nice and clearly. You can use an ESP8266 with just about any OLED, with some hacking.

Every time a new code is needed, click the reset button and, within two seconds, it displays the three most common TOTPs on hand (yes, it is that fast!)

Flash the Latest Version of CircuitPython

(You'll Need v2.2 or Higher)

We're using the ESP8266 Feather, which means it has lots of memory and Internet capability. We use the Internet part to get the current time with NTP. Since the ESP8266 doesn't have native USB, we have to upload our code using Ampy, an open-source command line tool that "types out" the Python script and saves it to the ESP8266 Flash memory.

Once you've gotten Ampy working, you'll need a bunch of Python libraries to get the OLED working. Use Ampy to create a directory called "lib" and upload "adafruit_ssd1306.mpy", "adafruit_register", and "adafruit_bus_device" library folders (https://learn.adafruit.com/welcome-to-circuitpython/circuitpython-libraries).

Then check with Ampy's ls command to verify all your files are in place!

Now you can download the main script to your computer and save it as "main.py".

The code is on GitHub along with an extended how-to (https://github.com/adafruit/Adafruit_Learning_System_Guides/blob/master/CircuitPy_OTP/main.py)

Don't upload it via Ampy yet! The current file has fake tokens in it that need to be set. Before uploading, change these two lines to your network SSID and password:

```
ssid = 'my_wifi_ssid'
password = 'my_wifi_password'
```

You'll also need to get two-factor "authenticator tokens/secrets." Each site is a little different with regards to how they do this. For example, when you set up Gmail for two-factor authentication, it will show you a QR code, which is great for phones. For us, we need the base32-encoded token. Click the "Can't Scan It?" link or otherwise request the text token.

That string of letters and numbers may be upper case or lower case. It may also be 16 digits or 24 or 32 or some other quantity. It doesn't matter! Grab that string, and remove the spaces so it's one long string like ra4ndd2utltotseol564z3jijj5jo677. Note that the number 0 and number 1 never appear, so anything that looks like an O, l, or an I is a letter. It doesn't matter if it's upper or lower case.

Now edit this section of the code. You can display up to three accounts on a Feather OLED. If you pad the name with spaces, the numbers will be right-justified, but it's not important - we are just picky. Here's our demo setup:

```
totp = [("Discord ",
➥ 'JBSWY3DPEHPK3PXP'), ("Gmail ",
➥ 'abcdefghijklmnopqrstuvwxyz
➥234567'), ("Accounts",
➥ 'asfdkwefoaiwejfa323nfjkl')]
```

If you want to test the setup first, you can keep the Discord entry, which is the "PyOTP" example token. Then, scan this with your phone in Authy or Google Authenticator.

OK, once you've set everything up, let's test! Run the program directly on the Feather with OLED attached using `ampy --port portname run main.py`.

You'll see it connect to your local network, get the time via NTP, then calculate and display OTP codes both on the OLED and on the serial port (you'll need to wait until the program is done to see the serial output).

If you do have a phone, check against your phone to make sure the codes are correct. Once you're satisfied, tweak these two lines to change the behavior. Then finalize by uploading main.py with Ampy's put command.

```
ALWAYS_ON = False # Set to true
➥ if you never want to go to sleep!
ON_SECONDS = 60 # how long to
➥ stay on if not in always_on mode
```

Good night and good luck!

Bluetooth Hacking 101

by Chuck Easttom

Introduction

Bluetooth is ubiquitous. You probably sync your phone with your car via Bluetooth. That is just one example of how Bluetooth technology can be found throughout our daily lives. Laptops, smart phones, tablets, cars, and all sorts of devices are Bluetooth-enabled. It is my hope that you will read this article with an eye towards testing the security of your own Bluetooth devices, rather than trying to breach others' Bluetooth (which is a crime).

The Bluetooth standard was developed by the Bluetooth Special Interest Group, which includes over 1,000 companies including Siemens, Intel, Toshiba, Motorola, and Ericsson. The IEEE standardized Bluetooth as IEEE 802.15.1, but no longer maintains the standard. The main advantage of Bluetooth is its ability to discover Bluetooth devices that are within range. This is precisely why it has become so common.

Bluetooth is a type of near field communication (i.e., limited range) that operates at 2.4 to 2.485 GHz and uses spread spectrum, frequency hopping at 1,600 hops per second. For readers not familiar with spread spectrum frequency hopping, that essentially means that the signal is hopping between frequencies in a given range. Bluetooth devices have a 48-bit identifier assigned by the manufacturer that is similar to a MAC address for a network card.

As an interesting aside regarding Bluetooth, the name comes from King Harald Bluetooth, a tenth century Danish king. He united the tribes of Denmark, thus the implication is that Bluetooth unites communication protocols. There have been different explanations for his name. One was that he had a bad tooth that was blue (i.e., rotted). Another explanation was that he was often clothed in blue.

Many texts and courses teach that Bluetooth has a maximum range of ten meters. However, that is only partially true. In fact, it is only true for Bluetooth 3.0. The following table summarizes the ranges and bandwidth for the various versions of Bluetooth.

Version	Bandwidth/Range
3.0	25 Mbit/s 10 meters (33 ft)
4.0	25 Mbit/s60 meters (200 ft)
5.0	50 Mbit/s 240 meters (800 ft)

Table 1- Bluetooth ranges

Bluetooth is designed as a layer protocol architecture. This means there are layers of protocols being used. The mandatory protocols that all Bluetooth devices have are LMP, L2CAP, and SDP. *LMP:* Link Management Protocol is used to set up and control the communication link between two devices.

L2CAP: Logical Link Control and Adaptation Protocol is used for multiplexing multiple connections between two devices.

SDP: Service Discovery Protocol is how two devices find out what services each offers.

RFCOMM: Radio Frequency Communications, as the name implies, provides a data stream. In this case, it is a virtual serial data stream.

BNEP: Bluetooth Network Encapsulation Protocol is used to transfer some other protocol over the L2CAP channel. It is encap-

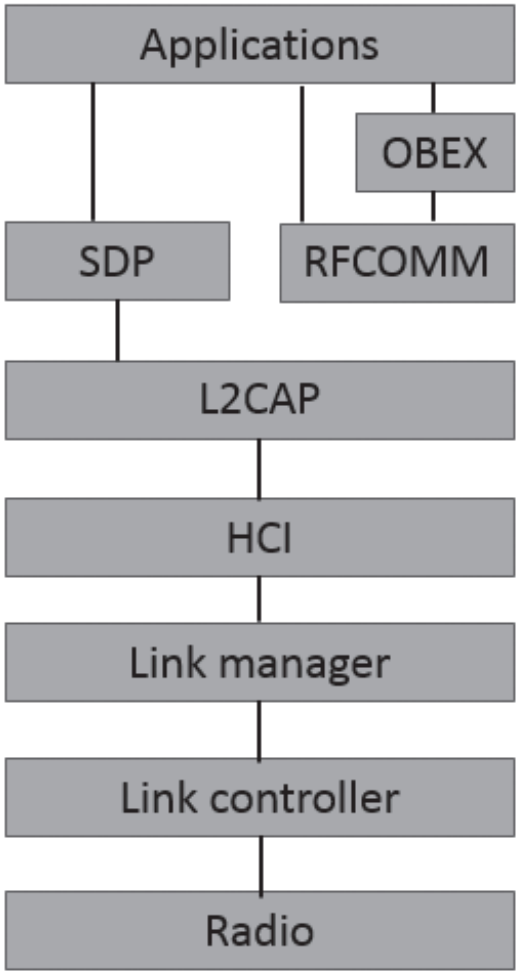
sulating the other protocol.

AVCTP: The Audio/Video Control Transport Protocol is used to transfer audio visual control commands over the L2CAP channel.

HCI: Host Controller Interface refers to any standardized communication between the host stack (i.e., the operating system) and the controller (the actual Bluetooth circuit).

OBEX: Object Exchange facilitates the transfer of binary objects between devices. It was originally designed for infrared, but is now used by Bluetooth. This is used in accessing phonebooks, printing, and other functions. It uses RFCOMM for communication.

The main protocol stack is shown in the following image:



Bluetooth Protocol Stack

The Bluetooth special interest group unveiled Bluetooth 5 during an event in London on 16 June 2016. This version of Bluetooth is primarily focused on the Internet of Things.

When you pair your device with another via Bluetooth, they exchange a bit of information including device name and list of services.

Bluetooth security defines four modes. Clearly, which mode your phone is using will have a great impact on what attacks will and won't work.

Security Mode 1 is non-secure.

Security Mode 2 controls access to certain services and uses a security manager. But this is only initiated after a link is established. Mode 2 has three levels:

Level 1: Open to all devices, the default level.

Level 2: Authentication only.

Level 3: Requires Authentication and Authorization. PIN number must be entered.

Security Mode 3 initiates security procedures before any link is established. It supports authentication and encryption. The NIST considers this the most secure.

Security Mode 4 requires authenticated links, but like mode 2 only initiates the authentication and encryption after a link is established.

Bluetooth Attacks

Now that you have a general idea of how Bluetooth operates, let's take a look at some of the attacks one can perform on a Bluetooth device. Bluetooth attacks are quite common, so let's begin with a brief summary of the common attacks. Contrary to what you may have seen on television and in movies, forced pairing a Bluetooth device is actually quite difficult and will only work with a really insecure device. However, there are attacks that can be done. This should familiarize you with what can be done to a Bluetooth-enabled device.

- Bluesnarfing is a class of attacks wherein the attacker attempts to get data from the phone.
- Blusnipping: This is a variation of Bluesnarfing. It works at longer ranges and was described at Defcon in 2004.
- Bluejacking is sending unsolicited data to a phone via Bluetooth. This is sometimes used to send spam instant messages.
- Bluesmacking is a Denial of Service attack wherein the target is flooded with packets.
- Bluebugging remotely accesses phone features. This may seem very similar to Bluesnarfing, but the goal with Bluebugging is not to get data, but to activate certain phone features.

- Bluesniffing is the same thing as war driving. The attacker is trying to find available Bluetooth devices to attack.
- Blueprinting gets its name from foot printing. In the case of Blueprinting, the attacker is trying to get information about the target phone.

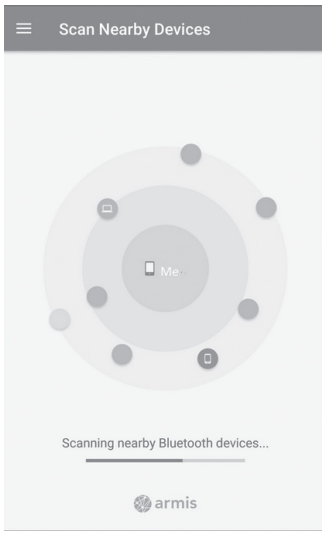
Bluetooth also provides an alternative way to access laptops. Most laptops are Bluetooth-enabled. This provides a possible avenue of attack on the laptop. It just so happens that many organizations block a variety of computer connections (CD/DVD, USB, etc.) to prevent users from either installing files or exfiltrating data. However, many of these same computers that have USB blocked have Bluetooth working. That would provide another pathway to exfiltrate data or install software.

Bluetooth Tools

There are a number of tools that an attacker can use to facilitate a Bluetooth attack. You should be familiar with at least some of these. Some tools are only for Android, others for iPhone, and some for both.

BlueBorne

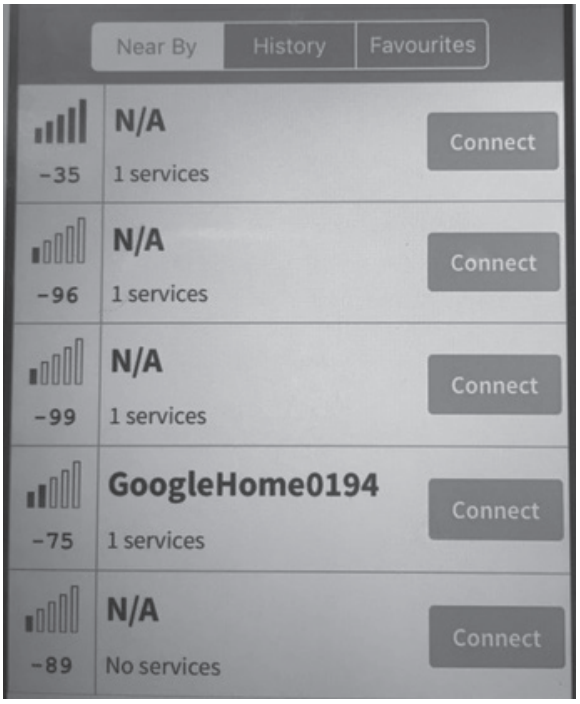
This is a vulnerability scanner for Bluetooth. It is available in the Google Play store for Android phones, and you can see it at `https://play.google.com/store/➥apps/details?id=com.armis.➥blueborne_detector&hl=en`. The vendor also has a white paper on Bluetooth vulnerabilities, you can view that at `http://➥go.armis.com/hubfs/BlueBorne➥%20Technical%20White%20Paper➥.pdf`. You can see the tool in the following image.



BlueBorne

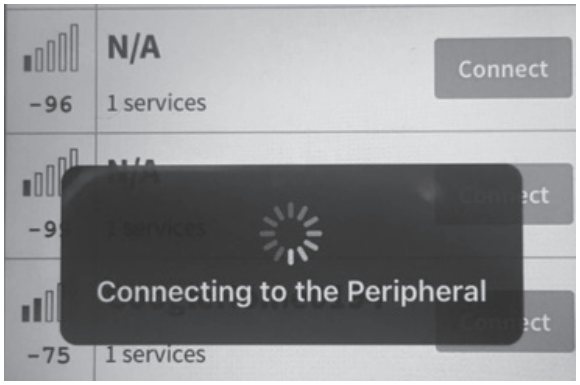
BLE Scanner

This tool is for the iPhone and is a free download. The first thing it will do is show you nearby Bluetooth devices. You can see this in the following image.



BLE Scanner

Then simply click the connect button to attempt to connect to that device. This is shown in the following image.



BLE Connect

Pally

Pally is another Bluetooth scanner for the iPhone. It has an easy to user interface and will provide you basic information about nearby Bluetooth devices. This can be seen in the following image.

Pally Scanner

There are a number of other tools one can find for either scanning Bluetooth or even attempting to hack into Bluetooth. It is important for a penetration tester to have a suite of tools and techniques at his or her disposal. The entire goal of a penetration test is to try the same techniques you believe an attacker would use. Here is a list of some other tools:

PhoneSnoop
Bluescanner
BH BlueJack
Bluesnarfer
btCrawler
Bluediving
Bloover II
btscanner
CIHwBT
BT Audit
Blue Alert
Blue Sniff

Conclusions

Bluetooth, while very convenient, is susceptible to a number of attacks. And there are a lot of tools to help someone attack a Bluetooth device. This article just scratched the surface of Bluetooth security, but hopefully it was enough to get you started studying this topic.

Hidden ISPs

by kes

As hackers, we often find ourselves on the fringes of society, doing things differently than the way they've been done because we find better ways, or just for the lolz.

Despite being in a golden age of online content, Internet access has stagnated. I don't think I need to go into the details of how all of the major ISPs have taken billions of dollars to expand Internet access and then sat on their hands, or how even cable companies are imposing data caps on wired connections while offering no technical backing, simply saying it's a business decision. And there's nothing most individuals can do about it.

After a recent move, I was finding myself searching for a new ISP. I am fortunate enough to have a local Wireless ISP (WISP) in my area. While most would consider the speed slow (10 Mbps), I enjoyed supporting a small, local company and the lack of data caps that came with it.

Unfortunately, after a site survey I found I could not get service at my location. So I was left searching for alternatives, which is how I found out about the following companies providing Internet service through the Sprint cell network.

Most mobile 4G hotspot plans you will find have data caps and will either charge you based on your overages or throttle your data. Neither of those is a good option, but these companies have found a loophole. Most of them seem to operate based on a contract that was made with a company called Clearwire that gave these companies access to their WiMax network unthrottled or capped. Clearwire was later acquired by Sprint for its spectrum and later decommissioned the WiMax network. Nevertheless, they still honor the old Clearwire contracts (though some of them claim it can change at any time) on Sprint's current 4G infrastructure. What this means for the less bandwidth-hogging hackers out there is that if you have good Sprint 4G coverage, you could potentially replace your home Internet with one of the plans and not have to worry about

data caps or throttling (though 4G speeds can vary more than wired infrastructure). These are some of the notes on the companies that I found on my search. I am not affiliated with any of these companies.

Calyx Institute (www.calyxinstitute.org)

This org is focused on providing privacy and cybersecurity education and will probably excite most readers, if not for the wireless Internet, for their other projects. This includes running (as of this writing) 13 Tor nodes, hosting the Canary Watch (RIP) project, and the founder being the first person to successfully fight a National Security Letter. They also accept Bitcoin, and hint that if you give them a fake name, they may have no way of knowing. The downside is that this org is the only one that does not offer returns, so if you get a hotspot and the Sprint service in your area sucks, you just made a generous donation with only warm feelings instead of a working Internet hotspot. The price is $500 a year, which is equivalent to $42 a month for the first year and then $400 a year (about $33 a month) for subsequent years. For the price, you get a Franklin R850 hotspot, a year's worth of service, some stickers, and a t-shirt.

4G Community (www.4gcommunity.org)

This org seems to focus on providing to educational and health care entities, so a little less exciting than Calyx, but it does have some pros. This is also the org I chose to try because of the return policy. If notified in seven days, the equipment can be returned for a refund minus an activation fee. Another benefit is that there are two levels of membership: select and premium. The select membership can be purchased for $325 a year ($27 a month), $214 for six months, or $154 for three months. Those all come with renewable options in annual, semi-annual, or bi-monthly. The renewals are roughly the same price per month, minus $100. The premium membership comes with a (presumably) nicer ZTE Pocket Wi-Fi hotspot and a slightly higher price tag, $398 for a year, $278 for six months, or $217 for three months. All with similar reductions for renewals.

FreeData.io

This is the new org on the block, with not many reviews backing it, and some on Reddit going so far as to claim it's "sketchy." I personally did not order from them, but if anyone at FreeData.io reads *2600*, maybe they could throw some discount codes in the advertisements section to generate some good will. This org does have the longest evaluation period at 14 days, though at least one person has claimed to have not gotten a refund for some time. They do offer the widest array of hotspot options, including the Netgear Gateway 6100D, which has external 4G antennas (to help with 4G signal) and 4 Gigabit Ethernet ports. All that does come with a cost - the Ultra plan, which includes the 6100D, is $679 for the initial signup, and then $250 a year after. The Premium, which includes the same ZTE Pocket Wi-Fi hotspot as the 4G community premium membership, is $500 for the first year and $250 after. The Basic membership includes a Franklin R850 for $370 a year and $250 after.

There are some limitations for these services. If you enjoy playing online games or do anything that requires a good ping time or low latency, you will probably have some issues with these services. If you do not live in an area that has good Sprint coverage, you will have a bad time. Most of these services will not let you connect more than ten devices at once. However, if you travel frequently to areas that don't have free Wi-Fi, or if you don't trust the free Wi-Fi, these can also serve their intended purposes well.

As for my experience... I decided to try 4G Community because of their reputation and return policy. After receiving the device, I eagerly ran a speed test and was very disappointed. Despite being in a solid 4G coverage area for Sprint, I got a speed of .23 Mbps down. After some testing, I became hopeful once again after getting one as high as 6 Mbps, only to be dismayed when the next one ran at a similar speed to the first. I determined at that point I would bite the bullet, get cable Internet, and live to fight another day.

Hopefully, you found my story useful and informative. If any of you get one of these services and disconnect from the large ISPs, then writing this would prove to have been worth my time.

Extrapolating Phone Numbers Using Facebook and PayPal

by Karan Saini

This article is a follow-up to a piece I wrote last year, entitled "Extracting Full Phone Numbers from the Leaked Snapchat Database." I'm hoping to highlight the privacy risk of linking the same phone number across all of your online accounts, and how it could very easily allow for your personal phone number to become known.

This was written with the assumption that the user is from the United States, but it could also very easily be adapted to work with users from another country.

Starting off, we will have to determine the user's location from their online presence.

This part should be easy enough, as most people reveal their current city on their Facebook page. However, if this information is not available on their Facebook page, it will have to be inferred from another publicly available source.

Head over to the "Forgot password" page on Facebook, and submit the email address of the user whose phone number you're trying to retrieve.

If the user has linked their phone number with their account, you will be presented with the last two digits of the same.

```
+1 *** *** **01
```

We're going to head over to PayPal's website for more useful information regarding the user's phone number.

Enter the email address of the user on PayPal's "Forgot password" page.

```
+1 2** *** 4401
```

Now, we're only five numbers short.

Well, actually, just three.

After having a quick look at the user's Facebook profile, I've been able to surmise that he is currently residing in New Jersey, USA. I'm also aware that many telephone numbers issued in New Jersey utilize the area code 201.

```
+1 201 *** 4401
```

It is also possible to get a list of all area codes which are used for phone numbers issued in a certain city or state (`InfoPlease.com` is very useful here).

Let's head over to `AllAreaCodes.com` for the final bit of information which we'll require.

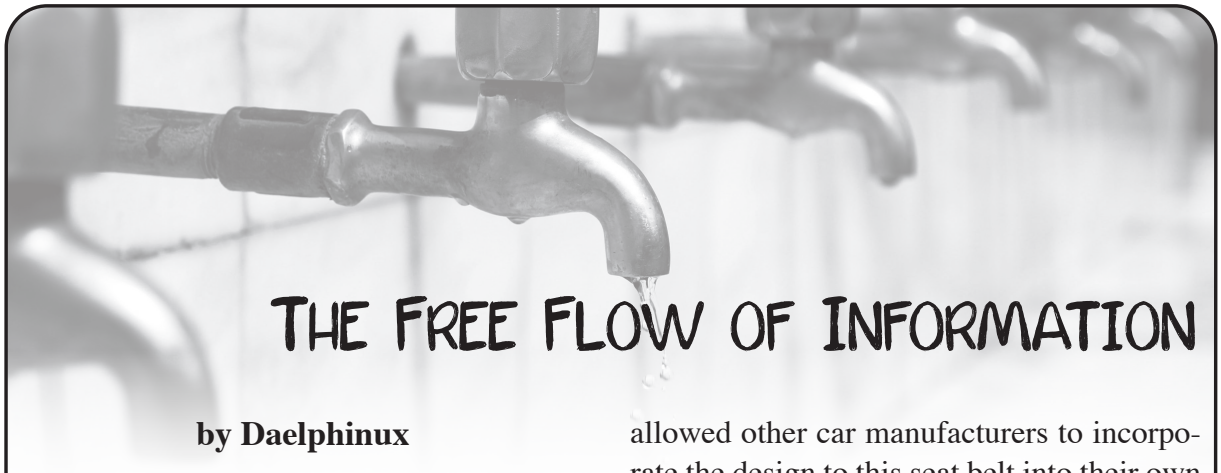
We're going to parse all area code prefixes and adjoin them with the last four digits of the partial phone number we currently have.

This part might be time consuming and arduous, but it is very essential to be able to obtain the user's phone number.

We're going to head over to the "Forgot password" page on Facebook once again.

This is the last step of the process - we're going to keep submitting and checking off phone numbers from our list (which shouldn't be very long to begin with, but if it is, the process can be automated using scripts) until you're able to observe a pattern of the email address that is most likely to match the one you originally provided.

It is also possible to further verify that the retrieved phone number belongs to the user, however, I'm not going to be writing about such methods in this article. Thank you for reading.



THE FREE FLOW OF INFORMATION

by Daelphinux

Researchers perform an essential function in modern society. Whether the researcher is working on a new Ebola vaccine, a way to save the bees, or a way to detect malware before it causes system damage, researchers perform the very necessary function of making human beings successful in our tenure on this Earth. They do so by proactively searching for knowledge, and, more importantly, they share this knowledge with other researchers, engineers of their field, or the public at large. This allows for a generally improved quality of life by way of more efficient and successful medicines, ways to curb the ever warming climate, or by preventing identity theft. The key part in this is that the information must be shared to be useful.

The Salk polio vaccine has been, without a doubt, one of the most influential pieces of technology in modern memory. Polio was an epidemic of horrific form. It mainly affects children, damaging the very ability of the afflicted to continue on in their pursuits. While it still exists in the world, polio was brought to its knees by Jonas Salk. After producing the vaccine, Salk was asked who owned the patent. His reply would inspire a world of researchers and hackers to come: "There is no patent. Could you patent the sun?" Salk's belief was that this revolutionary and lifesaving piece of research could better all of us. He believed that information should be freely available, and that it should do good for all.

A similar case happened with Volvo in the 1950s with the invention of the modern seat belt. Volvo invented the seat belt, patented it to protect it from what we would later know as patent trolls, and opened the patent. This allowed other car manufacturers to incorporate the design to this seat belt into their own vehicles. This simple act would save countless lives over the years, and continues to do so today.

Ultimately, as hackers, as researchers, and as people in general we have a responsibility to take our knowledge and share it. To work for not only the good or betterment of ourselves, but the betterment of all mankind. If your sole worth is based on the precept that you have a piece of technology that can protect someone, save someone, or help someone in need and you keep that knowledge proprietary just for wealth, you are a disgrace against all of mankind; a scourge whose information should be found and released as widely as possible until your worth is null, and you are left with only the shameful memory of how when you were given a chance to help, you thought only for yourself.

Especially in these tumultuous times where the very idea of fact is in danger, where government scientists are restricted from sharing information, and where news outlets are attacked for opposing regimes, we as keepers of information have a responsibility to make sure that public information is never brushed away. We have a responsibility to retain information so it can never be lost, and we have a responsibility to ensure that information can never be a weapon used against mankind. As long as we keep thinking, as long as we keep knowing, and as long as we fight for the truths of the world in a sea of lies, humanity will never fail.

Die Gedanken sind frei; wer kann sie erraten?

HACKER HAPPENINGS

Listed here are some upcoming events of interest to hackers. Hacker conferences generally cost under $200 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 13-15
CarolinaCon 14
Hilton Midtown
Raleigh, North Carolina
www.carolinacon.org

April 28-29
Maker Faire U.K.
Life Science Centre
Newcastle upon Tyne, England
www.makerfaireuk.com

May 4-5
THOTCON 0x9
Chicago, Illinois
thotcon.org

May 10-12
Converge
Cobo Hall
Detroit, Michigan
convergeconference.org

May 10-13
GPN18
Karlsruhe University of Arts and Design
Karlsruhe, Germany
entropia.de/GPN18

May 18-20
Maker Faire Bay Area
San Mateo Event Center
San Mateo, California
www.makerfaire.com

May 18-20
NolaCon
Crowne Plaza New Orleans
New Orleans, Louisiana
nolacon.com

June 1-3
CircleCityCon 5
The Westin
Indianapolis, Indiana
circlecitycon.com

June 7-8
RVAsec
University Student Commons
Virginia Commonwealth University
Richmond, Virginia
rvasec.com

June 20-24
ToorCamp
Doe Bay Resort & Retreat
San Juan Islands, Washington
toorcamp.toorcon.net

June 30 - July 1
Nuit Du Hack
Cité des Sciences et de l'Industrie
Paris, France
www.nuitduhack.com

July 20-22
The Circle of HOPE
Hotel Pennsylvania
New York City, New York
hope.net

August 9-12
DEF CON 26
Caesar's Palace
Las Vegas, Nevada
www.defcon.org

September 6-7
GrrCON
DeVos Place
Grand Rapids, Michigan
www.grrcon.org

September 22-23
World Maker Faire New York
New York Hall of Science
Queens, New York
www.makerfaire.com

October 5-7
DerbyCon
Marriott Louisville
Louisville, Kentucky
www.derbycon.com

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

Marketplace

Events

THE CIRCLE OF HOPE. A Hacker's Dozen. The 12th incarnation of the Hackers On Planet Earth series, taking place at the Hotel Pennsylvania in New York City July 20-22, 2018. We have expanded space this year! Tickets are currently on sale at store.2600.com. Want to give a talk? Check out the hope.net speaker section. You can also find info at the hope.net website on volunteering, being a vendor, running a workshop, and more!

For Sale

DEFEND YOUR WI-FI. Coaxifi delivers Wi-Fi over your home's coaxial cabling to eliminate dead zones. Reuse your existing router to send Wi-Fi farther. Check out our new WiFork kits! 10% off with promo code "SUP2600". coaxifi.com

PORTABLE PENETRATOR. Find WPA WPA2 WPS Wifi Keys Software. Customize reports use for consulting. https://shop.secpoint.com/2600

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

HACKER WAREHOUSE is your one stop shop for hacking equipment. We understand the importance of tools and gear which is why we carry only the highest quality gear from the best brands in the industry. From WiFi Hacking to Hardware Hacking to Lock Picks, we carry equipment that all hackers need. Check us out at HackerWarehouse.com.

BLUETOOTH SEARCH FOR ANDROID searches for nearby discoverable Bluetooth devices. Runs in the background while you use other apps, recording devices' names, addresses, and signal strength, along with device type, services, and manufacturer. Handles Bluetooth Classic and Bluetooth LE (on LE-equipped Android devices). 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 at http://tinyurl.com/btscan.

NEEDFULWARES.COM. Thank you for your time today in reading this. Please visit this site to view the most beautifully hacked coins and hardcover books, handmade in the still-great USA! There are wonderfully handcrafted (some may called them hacked) coin rings (and book safes to hide them in) for EVERYONE. Yes, I make change into something you can wear on your body and books that will keep your wares (or whatever) safely hidden. These are great gift ideas and all my work has a Made-In-USA, money-back, no-hassle guarantee. Custom, handmade by myself, orders are available.

HOW TO DO PRIVACY IN THE 21ST CENTURY by Peter Burnett. The War on Privacy is lost, and states and corporations collect more of our data than even they know what to do with. No one person or group can understand the implications of this, but we all know there is no going back. This book charts how we came to surrender everything from our faceprints to our location data. The question now is what next, and how can we reclaim our lost freedoms? Chapters on Wikileaks, Thomas Drake, Kim Dotcom, Snowden, corporate data collection, the (mis) prosecution of Barrett Brown, the Pirate Bay, the theory of privacy and prominent hacktivists, the blockchain, as well as wisdom from these very pages - *2600*. Profits from the sale of this book (published by Eyewear Publishing, London) are going to the EFF (eff.org). Find out more at http://peterburnett.info/privacy

HACKERSTICKERS.COM now carries cDc merchandise, accepts bitcoin, sells lock pick sets, bawls energy mints, and an awesome lineup of hacker clothing including the new Johnny Cupcakes x HackerStickers collaboration Hacker Big Kid Shirt. Get all the goods at HackerStickers.com.

Announcements

LISTEN TO THE GREYNOI.SE PODCAST. There are many information security podcasts out there, and we're just one of them. We are here for the newbies and veterans alike! The greynoi.se podcast discusses general news, science, and privacy as well as technology specific issues, all from the hacker perspective. Recorded LIVE at the SYNShop Hackerspace in Las Vegas, NV, Friday nights now at 7:30 pm PT. Recorded shows are usually online by Monday evenings. Have a listen and we LOVE feedback! https://greynoi.se

OFF THE HOOK is the weekly one hour hacker radio show presented Wednesday nights at 8:00 pm ET on WBAI 99.5 FM in New York City. You can also tune in over the net at www.2600.com/offthehook. Archives of all shows dating back to 1988 can be found at the *2600* site in mp3 format! Your feedback on the program is always welcome at oth@2600.com.

COVERTACTIONS.COM is the most comprehensive directory of encryption products anywhere. Search by type, hardware/software, country, open source, platform, and more. Now over 1000 products listed which include 217 VPN's, 189 messaging and 118 file encryption apps. These are just a few of the 27 categories available. There is no faster and easier way to find the encryption product that meets your requirements. Suggestions and feedback welcome.

SECUREMAC.COM has been hard at work putting together the weekly podcast *The Checklist* covering security and hacking related subjects near and dear to Apple users as well as general how-to's, history, and tips. Subscribe to this free podcast at www.securemac.com/checklist

AUSTIN HACKERSPACE: A shared workshop with electronics lab, laser cutters, 3D printers, CNC machines, car bay, woodworking, and more! $60/mo for 24/7 access to all this and a great community as well. Open House and open meetups weekly. 9701 Dessau Rd, Austin, TX http://atxhs.org/

THE SCI-FI AGENDA - the thinking person's guide to science fiction cinema. There's a lot to wish for regarding portrayals of hackers in movies, but we've come a long way since that unfortunate 1995 film... you know which. But in science fiction, the hacker mentality and hacker ethics are everywhere. The way we relate to novel technology is central to the story of many fine film productions, especially in the last 15 or so years. This is why we created The Sci-Fi Agenda, because smart, curious, and thoughtful people, such as the readers of *2600*, want equally smart sci-fi movies. Think of it as the hacker's curriculum, about 50 movies that pose interesting questions, whether about the power relation between AI and its creator (*Ex Machina*), the ethics of rogue biohacking (*Splice*), responsible disclosure of crypto

vulnerabilities (*Traveling Salesman*), the role of genomics versus employability (*Gattaca*), what mind uploading should be used for (*Extracted*), and the list goes on and on. We are certain you will enjoy many of the movies in this collection, and that they will provide plenty of food for thought relating to your own place in this world and the power that comes with knowledge. Visit us at scifiagenda.com and enjoy!

Services

PANIC STATION is a quarterly zine put out directly from prison that focuses on original writing, hacking, music, punk rock life, and prison shenanigans. *2600* readers can request a free issue by writing a letter to me. Submissions welcome, please only send letters (no stamps, etc.)! Vincent Veneziani, #249067G/1079583, 215 S. Burlington Rd. - SWSP, Bridgeton, NJ 08302.

HAVE YOU SEEN THE *2600* STORE? Plenty of features, hacker stuff, and all sorts of possibilities. We accept Bitcoin and Google Wallet, along with the usual credit cards and PayPal. We have an increasing amount of digital download capability for the magazine and for HOPE videos. Best of all, we've lowered prices on much of our stock. Won't you pay us a visit? store.2600.com

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. Sensei's digital forensic examiners hold the prestigious CISSP, CCE, and EnCE certifications. Our veteran experts are cool under fire in a courtroom - and their forensic skills are impeccable. We recover data nationwide from many sources, including computers, external media, tablets, 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. Our principals are co-authors of *Locked Down: Practical Information Security for Lawyers, 2nd edition* (American Bar Association 2016), *Encryption Made Simple for Lawyers* (American Bar Association 2015), and hundreds of articles on digital forensics and an award-winning blog on 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.

SKEPTICAL OF GITHUB? sr.ht is an in-progress software suite for hosting open source projects that's more in tune with the hacker way. sr.ht is more modular and more flexible, with features like mailing list driven development and full virt build automation with KVM. Interested in helping test the beta? Reach out to SirCmpwn: sir@cmpwn.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/

UNIX SHELL ACCOUNTS & WEB HOSTING SINCE 1999. JEAH.NET is one of the oldest and most trusted for fast, stable shell accounts. We include hundreds of funny, relevant vhosts for IRC, and access to new and classic *nix programs and compilers. JEAH.NET proudly hosts eggdrop bots, bouncers, IRCD, and web sites w/SQL. *2600* readers' setup fees are always waived. BTW: FYNE.COM (our sister co.) offers free DNS hosting and WHOIS privacy for $3.50 with all domains registered or transferred in!

ANTIQUE COMPUTERS. From Altos to Zorba and everything in between - Apple, Commodore, DEC, IBM, MITS, Xerox... vintagecomputer.net is full of classic computer hardware restoration information, links, tons of photos, video, document scans, and how-to articles. A place for preserving historical computers, maintaining working machines, running a library of hard-to-find documentation, magazines, SIG materials, BBS disks, manuals, and brochures from the 1950s through the early WWW era. http://www.vintagecomputer.net

LOCKPICKING101.COM - a locksport community driven by lock picking hobbyists and locksmiths alike. New to lock picking or want to advance your skills or help others learn? Just head over to LockPicking101.com and say Mr. Picks sent you!

DOUBLEHOP.ME is an edgy VPN startup aiming to rock the boat with double VPN hops and encrypted multi-datacenter interconnects. We enable clients to VPN to country A, and transparently exit country B. Increase your privacy with multiple legal jurisdictions and leave your traditional VPN behind! We don't keep logs, so there's no way for us to cooperate with LEOs, even if we felt compelled to. We accept Bitcoin and offer automated order processing! Use promo code COSBYSWEATER2600 for 50% off (https://www.doublehop.me).

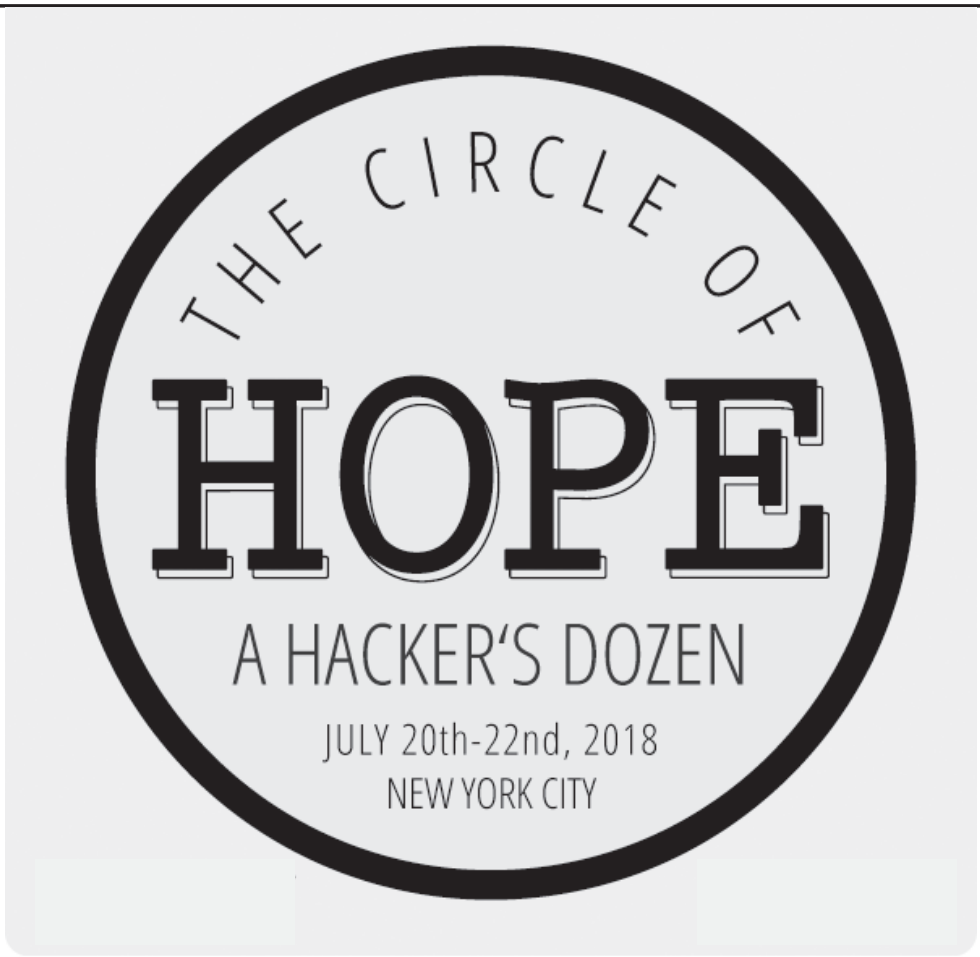
HACKERS, PHREAKERS, COMPUTER NERDS. Feel disillusioned, depressed, and dissatisfied with the way your life is passing? Need love, happiness, togetherness, and financial freedom? Here is the solution. Be with us to be yourself. You can be independent by joining with your kind. Enjoy the possibilities of collective thought, with associates who feel and think just like you do. Break that old routine, and dare to explore something new and unique. Contact THE HUB at: P. Bronson, P.O. Box 1000-AF8163, Houtzdale, PA 16698-1000.

Personals

SEEKING PENPALS. I'm incarcerated and looking for penpals. I've been down for two years now and the boredom is really starting to set in. My hometown is Cleveland OH and that is where they will be releasing me in a few years. Before the Feds kidnapped me, I worked network operations for an ISP. Being out of tech for so long now, I'm starting to feel antiquated. It would be nice to have people willing to discuss tech, answer questions that pop up, and send technical documents. Recently been interested in AI and quantum computing but no Internet access here and hardly any resources for keeping up with tech. I have many other interests too including general aviation, open-source projects, health/fitness, snowboarding, travel/foreign cultures, etc. Respond to the address below. Do not use address labels or stickers; it will be rejected/returned. Thank you! Daniel N. 61030-060, Federal Correctional Institution, PO Box 1000, Loretto, PA 15940.

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 marketplace@2600.com.

Deadline for Summer issue: 5/21/18.



If you haven't gotten your tickets yet, you should act fast as we expect to sell out! All kinds of payment options are available at https://store.2600.com. Once you preregister, you'll immediately get an email confirmation. Your actual tickets will be emailed to you as the conference draws closer.

For the entire month of April, ten percent of our ticket sales will go to benefit the Electronic Frontier Foundation.

If you want to speak at HOPE, there is still time. We've gotten a ton of great submissions, so be sure to make yours as interesting as possible. Full details and a FAQ can be found in the speaker section of the HOPE website.

We also have openings for workshops if you want to run one. More info is in the workshops section of the website.

There are a limited number of vendor spots available for people, companies, or organizations with something to offer our attendees. Visit our vendor section at the HOPE website for details.

The Circle of HOPE will have more than 100 speakers and talks, workshops, concerts, villages (lockpicking, hardware hacking, hackerspaces, etc.), art displays, vintage computers and phones, Segway rides, contests, and a whole lot more.

None of this would be possible without the hundreds of volunteers who pitch in to make it all happen. If you want to join our team, send an email to volunteers@hope.net and tell us if you have a specific skill or if you just want to be sent where you're needed.

The Hotel Pennsylvania is right across the street from Penn Station in New York City. They will be offering super discounted hotel rooms for the duration of the conference. Full details can be found at our website!

xii.hope.net

"Will the Internet become a theater, crowded with all of Humanity, where demagogic institutions cry 'fire' every time they wish to incite a mob? Will the Global Village be dominated, like so many other small towns, by schadenfreudal busybodies who build their own self-esteem from the wreckage of others whose reputations they've destroyed? So far, this has not happened because most of those in Cyberspace had a sense of responsibility about the preservation of a social contract that, however vague, was precious to us all. But what can be done about entities like Congress, who have neither a sense of that social contract nor enough stake in its preservation to motivate self-restraint?"
- John Perry Barlow, September 1998, from a piece written for Wired which was rejected for being "too political"

Inspirational Music: Einstürzende Neubauten, Jandek, Iggy Pop, Sigur Rós, Kendrick Lamar, Tricycle, Kyle Dixon & Michael Stein

Shout Outs: Students of Parkland, Waveframe, Christo & Dora, Mojo, Starman, the dogs of Chisinau, c3noc, abcdzyne, Ross-Tech, Tamp & Pull

R.I.P.: Felipito, Barlow, Bike So Good, Lovebug Starski

***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) 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 - $29 individual,
$50 corporate (U.S. Funds)
Overseas - $41 individual, $65 corporate

BACK ISSUES:
1984-1999 are $25 per year when available.
Individual issues for 1988-1999
are $6.25 each when available.
2000-2017 are $29 per year or $7.25 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: Bellagamba Bodegon, Armenia 1242, first table to the left of the front door.
Saavedra: Pizzeria La Farola de Saavedra, Av. Cabildo 4499, Capital Federal. 7 pm

AUSTRALIA

Central Coast: Central Coast Leagues Club (ground floor, outdoor area). 6 pm
Melbourne: Captain Melville, 34 Franklin St. 6 pm
Sydney: Metropolitan Hotel, 1 Bridge 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: International Village Mall food court.

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.

CZECHIA

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

FINLAND

Helsinki: Forum shopping center (Mannerheimintie 20), food court on floor zero.

FRANCE

Paris: Burger King, first floor, Place de la Republique. 6 pm

GREECE

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

IRELAND

Dublin: At the entrance to 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. Phone: 1-800-800-515. 7 pm
***Safed:** Courtyard of Ashkenazi Ari.

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

KAZAKHSTAN

Astana: CheckPoint Brasserie, Koshkarbayeva St 34. 8 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: Den Gode Nabo. 7 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

RUSSIA

Moscow: RNDM, Podkopayevskiy Pereulok, 7. 7 pm
Murmansk: Rock and Roll Music Bar, pr. Lenina, 11. 7 pm
Petrozavodsk: "Good Place" anti-cafe, pr. Pervomayskiy, 2. 7 pm
Saint Petersburg: Krasnodonskaya Ulitsa, 4. 7 pm

SWEDEN

Stockholm: Starbucks at Stockholm Central Station.

SWITZERLAND

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

THAILAND

Bangkok: The Connection Seminar Center. 6:30 pm

UNITED KINGDOM

England

Leeds: The Brewery Tap Leeds. 7 pm
London: Trocadero Shopping Center (near Piccadilly Circus), front entrance on Coventry St. 6:30 pm
Manchester: Bulls Head Pub on London Rd. 7:30 pm
Norwich: Coach and Horses on Thorpe Rd. 6 pm

Scotland

Edinburgh: Beehive Inn on Grassmarket. 6 pm
Glasgow: Starbucks, 9 Exchange Pl. 6 pm

Wales

Ewloe: St. David's Hotel.

UNITED STATES

Alabama

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

Arizona

Phoenix: Lux Central, 4400 N Central Ave. 6 pm
Prescott: Method Coffee, 3180 Willow Creek Rd. 6 pm
Tucson: BlackRock Brewers, 1664 S Research Loop #200. 6 pm

Arkansas

Fort Smith: Fort Smith Coffee Company, 1101 Rogers Ave. 6 pm

California

Anaheim (Fullerton): 23b Shop, 418 E Commonwealth Ave (behind Pizza Hut). 7 pm
Chico: Starbucks, 246 Broadway St. 6 pm
Los Angeles: Union Station, inside main entrance (Alameda St side) near the Traxx Bar. 6 pm
Monterey: East Village Coffee Lounge. 5:30 pm
Petaluma: Starbucks, 125 Petaluma Blvd N. 6 pm
San Diego: Regents Pizza, 4150 Regents Park Row #170.
San Francisco: 4 Embarcadero Center near street level fountains. 6 pm
San Jose: Outside the cafe at the MLK Library at 4th and E San Fernando. 6 pm

Colorado

Fort Collins: Dazbog Coffee, 2733 Council Tree Ave. 7 pm

Connecticut

Wallingford: Panera Bread, 1094 N Colony Rd. 6 pm

Delaware

Newark: Barnes and Nobles cafe area, Christiana Mall.

Florida

Fort Lauderdale: Grind Coffee Project, 599 SW 2nd Ave. 7 pm
Gainesville: In the back of the University of Florida's Reitz Union food court. 6 pm
Jacksonville: Kickbacks Gastropub, 910 King St. 6:30 pm
Melbourne: Sun Shoppe Cafe, 540 E New Haven Ave. 5:30 pm
Sebring: Lakeshore Mall food court, next to payphones. 6 pm
Tampa: Cafe at Barnes & Noble, 213 N Dale Mabry Hwy
Titusville: Playalinda Brewing Co,, 305 S Washington Ave.

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

Illinois

Champaign-Urbana: Lincoln Square Mall food court.
Chicago: O'Hare Oasis on 294 behind the bank kiosk. 8 pm
Peoria: Starbucks, 1200 West Main St.

Indiana

Evansville: Barnes & Noble cafe at 624 S Green River Rd.
Indianapolis: City Market, 2nd floor, just outside Tomlinson Tap Room.
West Lafayette: Jake's Roadhouse, 135 S Chauncey Ave.

Iowa

Ames: Memorial Union Building food court at the Iowa State University.
Davenport: Co-Lab, 627 W 2nd 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 (Cambridge): Starbucks, The Garage, 36 JFK St. 7 pm
Waltham: The Telephone Museum, 289 Moody 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, 2215 Scott Ave. 6 pm

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
Las Vegas (Henderson): SYN Shop, 1075 American Pacific Dr Suite C. 6 pm
Reno: Barnes & Noble Starbucks 5555 S. Virginia St.

New Hampshire

Keene: Local Burger, 82 Main St. 7 pm

New Jersey

Somerville: Dragonfly Cafe, 14 E Main St.

New York

Albany: Starbucks, 1244 Western Ave. 6 pm
New York: The Atrium at 875, 53rd St & 3rd Ave, lower level.
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: Morning Times, 10 E Hargett St. 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.
Columbus: Front of the food court fountain in Easton Mall. 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. 5:30 pm
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 Carolina

Myrtle Beach: SubProto, 3926 Wesley St, Suite 403.

South Dakota

Sioux Falls: Empire Mall, by Burger King.

Tennessee

Knoxville: West Town Mall food court. 6 pm
Nashville: Nashville Software School, 500 Interstate Blvd S #300. 6 pm

Texas

Austin: Whole Foods 2nd floor pavilion, 525 N Lamar Blvd. 7 pm
Dallas: Wild Turkey, 2470 Walnut Hill Ln. 7 pm
Houston: Ninfa's Express seating area, Galleria IV. 6 pm
Plano: Fourteen Eighteen Coffeehouse, 1418 Ave K. 6 pm

Vermont

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

Virginia

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
Reston: Refraction, 11911 Freedom Dr. 8th Fl. 7 pm
Richmond: Hack.RVA 1600 Roseneath Rd. 6 pm

Washington

Seattle: Cafe Allegro, upstairs, 4214 University Way NE (alley entrance). 6 pm
Spokane: Starbucks, 4727 N Division St.
Tacoma: Tacoma Mall food court. 6 pm
Wenatchee: Badger Mountain Brewing, 1 Orondo Ave.

Wisconsin

Madison: Fair Trade Coffee House, 418 State St.

URUGUAY

Montevideo: MAM Mercado Agricola de Montevideo, Jose L. Terra 2220, Choperia Mastra. 7 pm

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.

Follow @2600Meetings on Twitter and let us know your meeting's Twitter handle!

American Payphones

Canada. From the mean streets of Toronto, this wins the award for the hippest-looking phone in this collection.

Photo by David Quick

Mexico. Found throughout Mexico, these Telmex models are advertising a special rate of three pesos (around 16 cents) for local calls of unlimited length.

Photo by Babu Mengelepouti

Antigua. This phone is pretty well-used and is operated by Cable & Wireless. Interestingly, it resides in an old British phone booth.

Photo by B Robin

Chile. Discovered in the Las Condes area of Santiago de Chile, this model appears to have the same firmware as 1990s Argentine payphones.

Photo by Arturo "Buanzo" Busleiman

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

It's always a good idea to monitor your child's viewing habits and this only proves the point. Thanks to **Shar**, whose daughter was engrossed in an episode of *Johnny Test* (Season 6, Episode 18) where a giant super-computer named the Enigma-tron 2600 was being used to hack a corporate website in order to get free stuff. We really couldn't make these things up if we tried.

We heard rumors about the existence of this bus for ages. (We already discovered the New York City subway car with our number on it back in 2005.) This express bus was seen by **John Calabrese** as it sped by on 23rd Street in Manhattan. Judging from its destination display, Staten Island is the place to go if you want to see this thing at rest.

If you've spotted something that has "2600" in it or anything else of interest to the hacker world (such as funny uses of "hacker," "unix," "404," you get the idea...), take a picture and send it on in! Be sure to use the highest quality settings on your camera to increase the odds of it getting printed. Make sure and tell us where you spotted your subject along with any other info that makes it interesting - many photos are eliminated due to lack of detail.

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

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