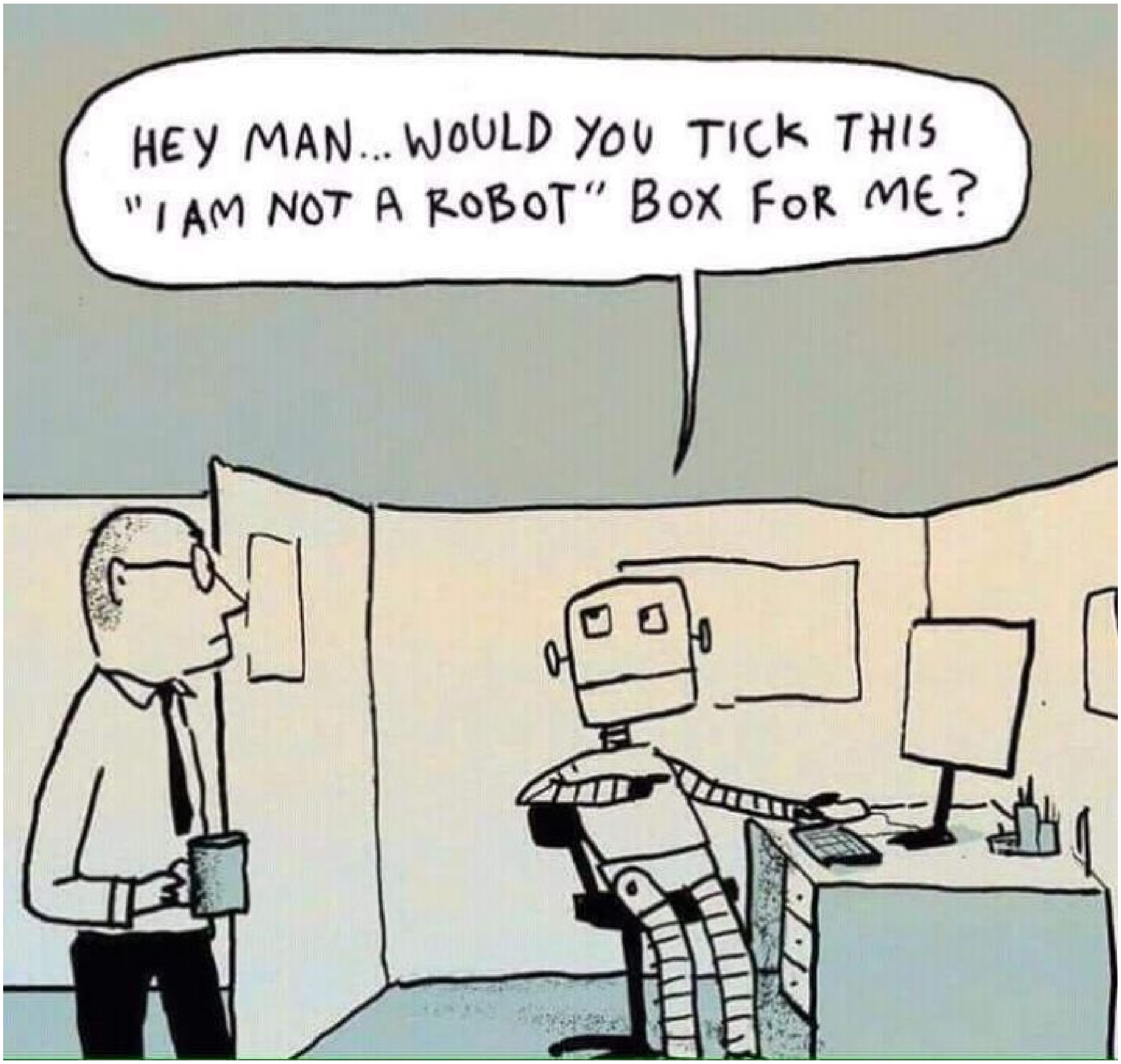
Security Now! #687 - 10-30-18

Securing the Vending Machine

This week on Security Now!

This week we follow-up on the Win10 ZIP extraction trouble, discuss some welcome Android patching news, look at SandboxEscaper's latest 0-day surprise, examine the Hadoop DemonBot, follow up on US DoD insecurity, look into the consequences of publicly exposed Docker server APIs, look at a DDoS-for-Hire front end, check out the mid-week Windows non-security Windows 10 bug fix update, look at the just-released Firefox v63, and examine a new privilege escalation vulnerability affecting Linux and OpenBSD. We also handle a bit of errata, some Sci-Fi miscellany, and a bit of closing the loop feedback from a listener. Then we answer last week's puzzler by exploring various ways of securing those vending machines.


HEY MAN... WOULD YOU TICK THIS "I AM NOT A ROBOT" BOX FOR ME?

Security News

Windows 10 ZIP file contents overwrite:
https://answers.microsoft.com/en-us/windows/forum/all/if-you-copy-files-from-a-zip-file-without/18c78dcb-8ec6-4d82-90c6-31b90a428fc7

At 5:27PM last Tuesday afternoon Microsoft acknowledged the ZIP file problem in Windows 10 in a posting with the long title: "If you copy files from a .ZIP file without extracting them, they might not be copied or moved correctly, even though it looks like they have been."

There is a known issue in the Windows 10 October Update where the consent prompt “Do you want to replace these files” is missing when copying contents from a .ZIP file.

With the Windows 10 October 2018 Update, if you copy or move files from a .ZIP file (without first “extracting” the contents) in to a new destination folder that contains duplicate filenames or is write-protected, you don’t get a “Do you want to replace these files” prompt. It will appear that the files were overwritten, when in fact the copy action for those files is not executed and files have not been overwritten.

This failure can occur in the following scenarios:

- Copying from a compressed (.zip) folder to a regular folder.
- Moving from a compressed folder to a regular folder.
- Copying from a compressed folder to a protected folder.

Note: While the copy action for the duplication file names does not complete and no files are overwritten, the “move” command will also silently fail and might remove/delete the moved file. We recommend you fully extract the zip folder before you copy files to a new destination folder to avoid this issue.

If you deleted items at any point in this process, you can recover files that were not copied to the destination folder or were unintentionally recycled by doing the following:

Restore files from the Recycle Bin:

- Open the Recycle Bin
- Locate item
- Right-click and select Restore.

Restore files from Temporary File Directory:

- Open the Run command box by simultaneously pressing Windows logo + R keys.
- Type %temp% and then click OK to open the Temp folder containing temporary files.
- Locate the file or folder. On the ribbon, select Move to and choose a location or folder to move the file into.

Important: Do not attempt to Cut and Paste items from a compressed (.zip) folder. This may result in unintentionally deleting items that may not be recoverable.

Microsoft is working on a resolution and estimates a solution will be available in early November for this issue.

(Note that since the 2nd Tuesday of November is the 13th, and since this is a potentially destructive bug, it's unclear whether Microsoft will wait another two weeks.)

Yay!!! "Google mandates two years of security updates for popular phones in new Android contract"

The Verge recently obtained copies of some confidential Google vendor contracts.

Paraphrasing from The Verge's coverage back in early MAY:

> Google says it will require Android phone manufacturers to roll out security patches on a "regular" basis — though it isn't clear who that requirement will apply to or how rigorous the mandate will be.
>
> On Wednesday, during a talk at Google's annual developer conference that was caught by 9to5Google via XDA Developers, the company announced that many more users would receive regular security patches thanks to new agreements it's making with partners. David Kleidermacher, Google's head of Android security reportedly said: "When you have billions of users, it's a large target. And so it deserves the strongest possible defense. We've also worked on building security patching into our OEM agreements. Now this will really lead to a massive increase in the number of devices and users receiving regular security patches."
>
> Unfortunately, there are no details beyond that. We reached out to Google to learn how frequent the security updates will be and who they'll apply to, but the company didn't immediately have answers for us. It sounds like the requirement will apply only to new phones launching on Oreo or later that take advantage of Google Play services — so likely nothing in China. Even then, it isn't clear whether it'll apply to all of Google's partners.

Now, today, with the aid of some leaked confidential documents, The Verge has been able to report the following:

> Every month, a security team at Google releases a new set of patches for Android — and every month, carriers and manufacturers struggle to get them installed on actual phones. It's a complex, long-standing problem, but confidential contracts obtained by The Verge show many manufacturers now have explicit obligations about keeping their phones updated written into their contract with Google.
>
> A contract obtained by The Verge requires Android device makers to regularly install updates for any popular phone or tablet for at least two years. Google's contract with Android partners stipulates that they must provide "at least four security updates" within one year of the phone's launch. Security updates are mandated within the second year as well, though without a specified minimum number of releases.
>
> David Kleidermacher, Google's head of Android security, referred to these terms earlier this year during a talk at Google I/O. Kleidermacher said that Google had added a provision into

its agreements with partners to roll out “regular” security updates. But it wasn’t clear which devices those would apply to, how often those updates would come, or for how long.

The terms cover any device launched after January 31st, 2018 that’s been activated by more than 100,000 users. Starting July 31st, the patching requirements were applied to 75 percent of a manufacturer’s “security mandatory models.” Starting on January 31st, 2019, Google will require that all security mandatory devices receive these updates.

Manufacturers have to patch flaws identified by Google within a specific timeframe. By the end of each month, covered devices must be protected against all vulnerabilities identified more than 90 days ago. That means that, even without an annual update minimum, this rolling window mandates that devices are regularly patched. Additionally, devices must launch with this same level of bug fix coverage. If manufacturers fail to keep their devices updated, Google says it could withhold approval of future phones, which could prevent them from being released.

SandboxEscaper drops another 0-Day, leaving all Windows users exposed.

First, to recap: Back in late August, when she was apparently having a bad day, the researcher known as "SandboxEscaper" posted to Twitter to expose and detail a PoC exploit for a local privilege escalation vulnerability in Microsoft Windows Task Scheduler due to its handling of the Advanced Local Procedure Call (ALPC) service. And, as we know, almost immediately after the PoC was released, the exploitation of this vulnerability was found being exploited in the wild... before Microsoft had time to address the issue, as they did in the following September 2018 Security Patch Tuesday Updates.

And at least the first portion of that history is now repeating: SandboxEscaper has again tweeted news and details of a previously unknown vulnerability, providing a working PoC that leaves all Windows 10 and server 2016 and 2019 machines vulnerable to attackers until it can be fixed, hopefully as part of next month's security Patch Tuesday, scheduled for November 13, 2018.

Her tweet leads with a {Link to the exploit} … then continues: "Here's a low quality bug that is a pain to exploit.. still unpatched. I'm done with all this anyway. Probably going to get into problems because of being broke now.. but whatever."

https://github.com/SandboxEscaper/randomrepo/blob/master/DeleteBug1.rar

She describes what she found as a privilege escalation flaw residing in the DSSVC.DLL. This is a new Microsoft Data Sharing service introduced in Windows 10 and recent versions of Windows server editions, the vulnerability does not affect older versions of Windows operating systems including 7 or 8.1. This service is a local service that provides data brokering between applications. It runs under the LocalSystem account so it has full system privilges and its compromise can potentially give the attacker those same rights.

The PoC exploit code (deletebug.exe) she released (and which my Firefox browser immediately flagged as dangerous and cautioned against) allows a low privileged user to delete critical system files—that otherwise would only be possible via admin level privileges. So it proves the point.

She commented: *"Not the same bug I posted a while back, this doesn't write garbage to files but actually deletes them.. meaning you can delete application dll's and hope they go look for them in user write-able locations. Or delete stuff used by system services c:\windows\temp and hijack them."*

The service contains an undocumented function which checks whether the requesting user has permissions to create a file in a chosen location. To do so, it 'impersonates' the requesting user, tries to create an empty file, remembers whether this file creation succeeded and then deletes the file. The problem is that it drops the impersonating the user too soon, causing said file deletion to be performed as the system user instead of the requesting low-privileged user." Whoopsie!!

SandboxEscaper's PoC deletes the critical system 'pci.sys' file, rendering the system unbootable:

> Bug description:
>
> RpcDSSMoveFromSharedFile(handle,L"token",L"c:\\blah1\\pci.sys");
>
> This function exposed over alpc, has a arbitrary delete vuln.
>
> Hitting the timing was pretty annoying. But my PoC will keep rerunning until c:\windows\system32\drivers\pci.sys is deleted.
>
> I believe it's impossible to hit the timing on a single core VM. I was able to trigger it using 4 cores on my VM. (Sadly I wasn't able to use OPLOCKS with this particular bug)
>
> Root cause is basically just a delete without impersonation because of an early revert to self. Should be straight forward to fix it…
>
> Exploitation wise.. you either try to trigger dll hijacking issues in 3rd party software.. or delete temp files used by a system service in c:\windows\temp and hijack them and hopefully do some evil stuff.

On the topic of these 0-day releases...
Mitja Kolsek / @mkolsek / Oct 24 / Replying to @SandboxEscaper

Mitja Kolsek Retweeted Security Response
Hey girl, in case you missed this:
https://twitter.com/msftsecresponse/status/1055156542280884224?s=19 …

Please continue with your work in this area and get paid by MS. What you're finding aren't easy-to-find bugs and your PoCs and writeups are of high quality. There, you should know this. Cheers.

MsftSecResponse: Yes, this vulnerability is in scope for the Windows Insider Preview bounty program. -KP

Peter Stevesant / @binaryfraud2017 / Oct 26
Replying to @msftsecresponse @mkolsek

well , when you see how much MS pay for exploit, you should sell them to http://zerodium.com ...they will consider your work much better ..

The new DemonBot network courtesy of Hadoop...
https://blog.radware.com/security/2018/10/new-demonbot-discovered/

Wikipedia: Apache Hadoop is a collection of open-source software utilities that facilitate using a network of many computers to solve problems involving massive amounts of data and computation. It provides a software framework for distributed storage and processing of big data using the MapReduce programming model. Originally designed for computer clusters built from commodity hardware—still the common use—it has also found use on clusters of higher-end hardware. All the modules in Hadoop are designed with a fundamental assumption that hardware failures are common occurrences and should be automatically handled by the framework.

As a consequence of Hadoop cluster nature, Hadoop clusters are often very powerful, well connected, and placed into the Internet cloud... where they are accessible either deliberately or inadvertently.

As it turns out, Hadoop's "Yet Another Resource Negotiator" (YARN), which provides cluster resource management for enterprise Hadoop deployments, has had a known flaw in the handling of its REST API, for which a Metasploit proof-of-concept was published after a 21-line Python PoC was posted onto Github seven months ago, last March:
https://github.com/vulhub/vulhub/blob/master/hadoop/unauthorized-yarn/exploit.py

The description for the Metasploit module reads: "This module exploits an unauthenticated command execution vulnerability in Apache Hadoop through ResourceManager REST API."
https://www.exploit-db.com/exploits/45025/

Radware reports that DemonBot does not self propagate in worm style -- where existing infection instances search for new victims -- but that they have identified more than 70 servers using the "spray and pray" tactic to blindly find and infect millions of exposed, publicly available and, of course, unpatched, Hadoop instances.

The DemonBot is being used to source extremely powerful and debilitating DDoS attacks, which those Hadoop instances can do because they are typically located on large and well-connected pipes.

As with the millions of compromised routers and other insecure publicly exposed and vulnerable services, attacks such as these will be part of the Internet terrain until we figure out how to build secure systems.

Hack the DoD?

Two weeks after that rather horrifying Government Accountability Office (GAO) report detailing glaring cybersecurity issues in weapons systems at the US Department of Defense, the DoD has announced that is expanding its existing "Hack the Pentagon" bug-bounty program to include hardware assets. They will be tapping the Synack, HackerOne and Bugcrowd platforms to attract more white hats to the effort.

Since the "Hack the Pentagon" program was kicked off in 2016, bug-hunters have found more than 5,000 code vulnerabilities, and six public-facing bounty challenges have been run, including the most recent, Hack the Marine Corps in August. Other sessions have focused on the Air Force, Army and the Defense Travel Service.

A three-year, $34 million “indefinite delivery, indefinite quantity” contract package covering the three bug-bounty managing companies will crowdsource vetted hackers to probe the DoD’s websites, hardware and physical systems.

According to the contract’s performance work statement, the government expects its military contractors to run at least eight limited-time challenges and five continuous challenges during the first year of the 3-year contract, and more if an option is exercised. Each program will last between three months to a full year, and they could overlap.

It seems to me that that virtually all of the interesting -- and critical -- systems are classified. How does some public white hat hack an F-35? I wonder whether what the U.S. needs is an internal, vetted, freely roaming security-cleared internal hacking squad that is not tied to, and which has no loyalty to, any single specific branch or group. If the various branches of the military are competing with each other, as I believe they tend to -- or perhaps if various contractors are competing with each other -- turn each opposing group's hackers loose on the opposition's systems. Everything we have learned about bug hunting and hacking informs us that an inherently adversarial motivation is part of the success formula.

Believe it or not... Publicly exposed Docker Engine APIs. <sigh>
What is Docker? "Docker creates simple tooling and a universal packaging approach that bundles up all application dependencies inside a container. Docker Engine enables containerized applications to run anywhere consistently on any infrastructure, solving “dependency hell” for developers and operations teams, and eliminating the “it works on my laptop!” problem.

Docker Engine is available for Linux (CentOS, Debian, Fedora, Oracle Linux, RHEL, SUSE, and Ubuntu) or Windows Server operating systems and is based on containerd - the open source container runtime project that Docker donated to the Cloud Native Computing Foundation (CNCF) in 2017. It is available as both a free community-supported engine and as a commercially-supported enterprise engine (Docker Engine-Enterprise) that also forms the foundation for an enterprise container platform."

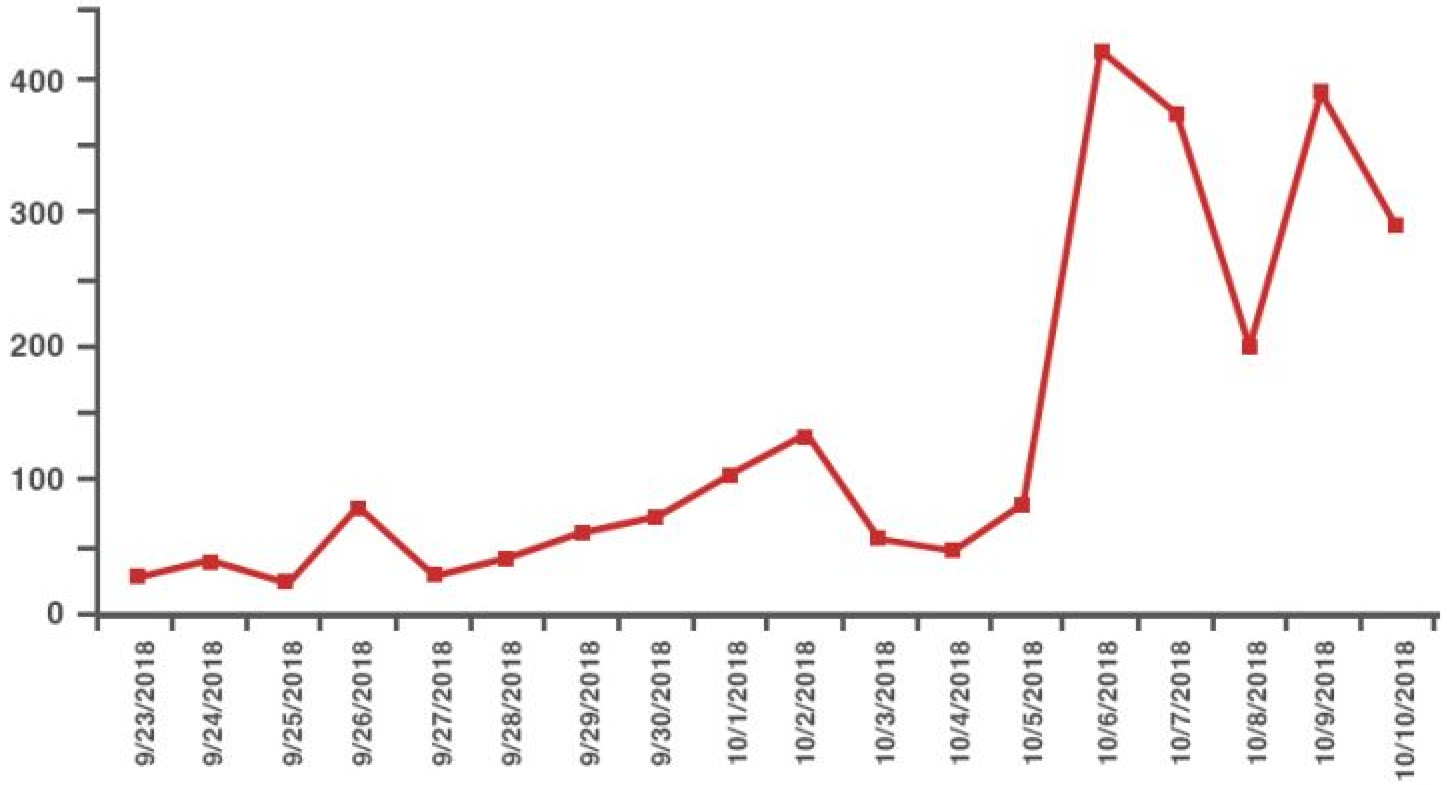
Believe it or not ... (regular Security Now! listeners will not be surprised) ... this Docker Engine is a server which interacts with its local environment via an API... and that API exists, exposed, on the public Internet. Or as Trend Micro put it in their security advisory: "Misconfigured Container Abused to Deliver Cryptocurrency-mining Malware."
https://blog.trendmicro.com/trendlabs-security-intelligence/misconfigured-container-abused-to-deliver-cryptocurrency-mining-malware/

"We recently observed cases of abuse of the systems running misconfigured Docker Engine-Community with Docker application program interface (API) ports exposed. We also noticed that the malicious activities were focused on scanning for open ports 2375/TCP and 2376/TCP, which are used by the Docker engine daemon (dockerd). The intrusion attempts to deploy a cryptocurrency-mining malware (detected by Trend Micro as Coinminer.SH.MALXMR.ATNE) on the misconfigured systems."

"The Docker APIs allow remote users to control Docker images like a local Docker client does. Opening the API port for external access is not recommended, as it can allow hackers to abuse this misconfiguration for malicious activities.

The Docker engine itself isn’t compromised or abused, and Docker’s enterprise platform is not affected. We found these rare instances of abuse on Docker Community [as opposed to Enterprise] versions. In fact, Docker’s technology has security features that its users can enable and configure to protect containers and workloads. Docker also has tools, documentations, and guidelines that can help with securing Docker community and enterprise platforms. Of course, in either case, security best practices would dictate that these ports should never be left open. For example, enterprises running business applications are recommended to use a commercial Docker Enterprise solution that has a precise, role-based access control settings that only allow authenticated use of the API.

In our research, the exposure of the Docker API ports was a result of misconfiguration on the user’s part, as we found that the misconfigurations were manually set up at the administrator level. Indeed, exposure to threats via misconfigurations isn’t new, but it can be a perennial challenge for organizations. In fact, our Shodan search revealed that many still have their Docker hosts misconfigured, especially in China. The search also revealed exposed misconfigured Docker hosts in the U.S., France, Germany, Singapore, Netherlands, United Kingdom, Japan, India, and Ireland. The majority of the exposed systems run Linux OS.



However, note that the numbers are in the hundreds, rather than in the thousands. This strongly suggests what is apparent from Docker's deliberate security focus: Unlike many of the "open by policy" mistakes we keep seeing routers making, in this case it appears that the Docker system is an example of a developer group who have done everything right. But that, nevertheless, it IS possible to somehow semi-deliberately misconfigure a Docker server instance such that it's API ports are publicly exposed.

What happens when an open Docker server instance is found?...

When the container is deployed and activated, it will launch an auto.sh script that will download a Monero miner and configure it to launch automatically. The script will also download port scanning software, which will scan for other vulnerable Docker Engine instances on port 2375 and 2376 and attempt to further spread to them.

Scan all networks seen from the host, with a scan rate of 50,000 packets per second, for open port 2375 and 2376; the result is saved in local.txt (anonymized/defanged):

```
masscan “$@” -p2375,2376 –rate=50000 -oG local.txt;
```

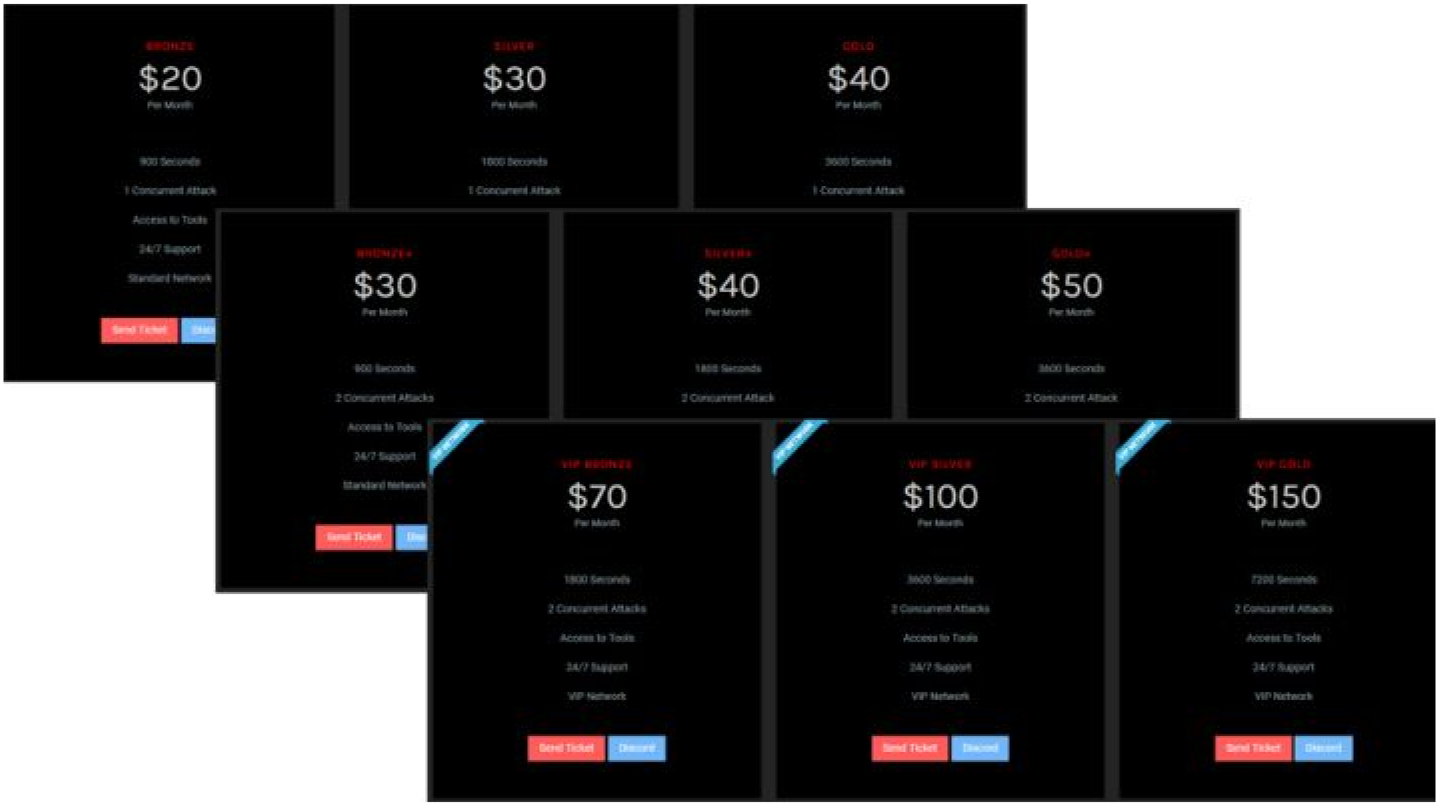
Conduct lateral movement by infecting or abusing more hosts found in previous reconnaissance:

```
sudo sed -i ‘s/^Host: \([0-9.]*\).*Ports: \([0-9]*\).*$/\1:\2/g’ local.txt;
sudo sh test3.sh local.txt;
```

With this method, a large number (all of them) Docker Engine containers can be amassed that mine coins for the attacker.

What's the going rate for DDoS for hire?
https://www.fortinet.com/blog/threat-research/ddos-for-hire-service-powered-by-bushido-botnet-.html

Fortinet's Threat Research article is titled: "DDoS-for-Hire Service Powered by Bushido Botnet"



BRONZE
$20
Per Month
900 Seconds
1 Concurrent Attack
Access to Tools
24/7 Support
Standard Network
Send Ticket
Disc

SILVER
$30
Per Month
1800 Seconds
1 Concurrent Attack

GOLD
$40
Per Month
3600 Seconds
1 Concurrent Attack

BRONZE+
$30
Per Month
900 Seconds
2 Concurrent Attacks
Access to Tools
24/7 Support
Standard Network
Send Ticket
Disc

SILVER+
$40
Per Month
1800 Seconds
2 Concurrent Attack

GOLD+
$50
Per Month
3600 Seconds
2 Concurrent Attack

VIP BRONZE
$70
Per Month
1800 Seconds
2 Concurrent Attacks
Access to Tools
24/7 Support
VIP Network
Send Ticket
Discord

VIP SILVER
$100
Per Month
3600 Seconds
2 Concurrent Attacks
Access to Tools
24/7 Support
VIP Network
Send Ticket
Discord

VIP GOLD
$150
Per Month
7200 Seconds
2 Concurrent Attacks
Access to Tools
24/7 Support
VIP Network
Send Ticket
Discord

- $20 for the Bronze plan: 900 seconds / 15 minutes, offering 1 concurrent attack using the Standard network.
- $150 for the VIP Gold plan: 7200 seconds / 2 hours, offering 2 concurrent attacks using the VIP network.

The world we live in today...

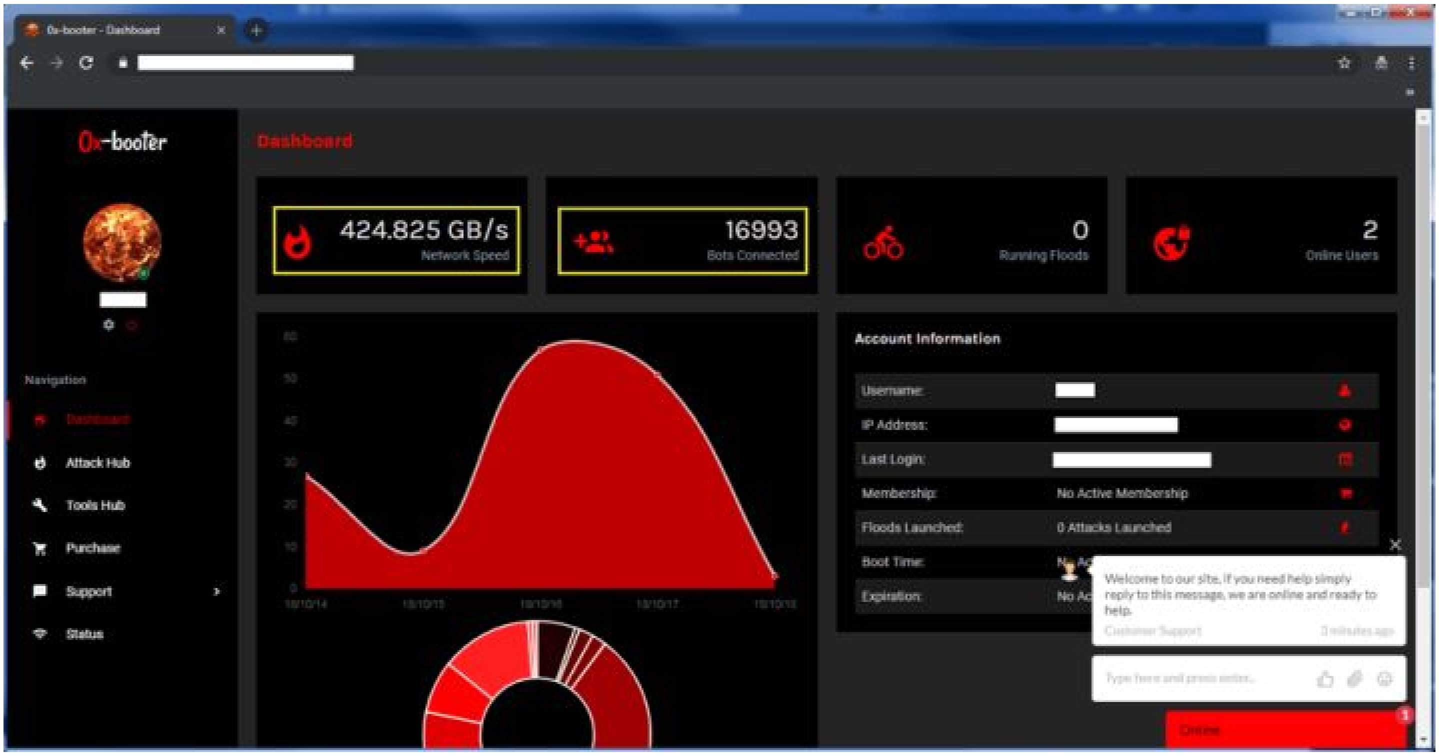
As Fortinet writes: "Distributed Denial-of-Service (DDoS) service offerings, often disguised as legitimate “booter” or “stresser” services, continue to increase in the cyber underground market. This relatively new Crime-as-a-Service trend has created an entry point for novice DDoS attackers, offering a simple option to anonymously attack nearly any website and forcing it offline for a small fee.

Due to the public release of the source code of some popular bots, building a botnet to provide these services is simpler than ever. A quick Google search returns lists of resources for botnet builders, usually with complete step-by-step instructions. Being able to re-use and even modify the source code has enabled cyber criminals to create their own versions that implement new functionalities.

[Note that Fortinet will be delivering a talk during the upcoming Botconf 2018 being held December 4-8 in Toulouse, France. During this talk Fortinet will be detailing the re-use of Mirai source code and the effect it has had upon the development of other botnets in their presentation titled: “Mirai: Beyond the Aftermath.”]

During our regular monitoring, the FortiGuard Labs team recently discovered a new platform offering DDoS-for-hire service called “0x-booter.” First appearing on October 17, 2018, 0x-booter is available to anyone who signs up on the website. This service comes with a simple to use interface which enables practically anyone to learn and use the service.

Initiating a DDoS attack is made through a web user interface, which eliminates the need for direct contact between the user and the bot master. In the attack hub interface the details of the host or domain, port, attack duration, and the type of attack can all be configured before launching an attack.



Account Information

Username:	[redacted]
IP Address:	[redacted]
Last Login:	[redacted]
Membership:	No Active Membership
Floods Launched:	0 Attacks Launched
Boot Time:	No Ac
Expiration:	No Ac

Welcome to our site, if you need help simply reply to this message, we are online and ready to help.
Customer Support 3 minutes ago

Type here and press enter..

October 24, 2018—KB4462933 (OS Build 17134.376)
https://support.microsoft.com/en-us/help/4462933/windows-10-update-kb4462933

I set my Win10 "delay" to 0 to confirm that KB4462933 would install. As we have become accustomed to, the mid-week updates are non-security big fixes, and in this case the list is extensive.

- Addresses the redenomination of local currency that the Central Bank of Venezuela implemented to enter the Bolivar Soberano into circulation.

- Addresses an issue that prevents the clock and date flyout from appearing when the region format is Spanish (Spain) and the sorting method is Traditional.

- Addresses an issue that causes the GetCalendarInfo function to return a wrong value for the Japanese era.

- Addresses an issue in which applications have handle leaks when using client authentication certificates with the TLS protocol. This issue occurs when the FreeCredentialsHandle call occurs before the DeleteSecurityContext call in the application code.

- Addresses an issue in which Scheduled Tasks configured to run on a specific day of the week don't execute at the expected time.

- Addresses an issue in which the System.Security.Cryptography.Algorithms reference was not correctly loaded on .NET Framework 4.7.1 after the July 10, 2018 and August 14, 2018 patches.

- Addresses an issue that may cause some applications to stop working after unplugging a tablet.

- Addresses an issue in which application titles that were unexpectedly long were not predictably displayed using ellipses (…). In some cases, the text truncations that appear may confuse users.

- Addresses an issue in which users cannot enter East Asian text when prompted to create password hints during the upgrade process.

Firefox makes its move to v63 and brings more built-in tracking protection:

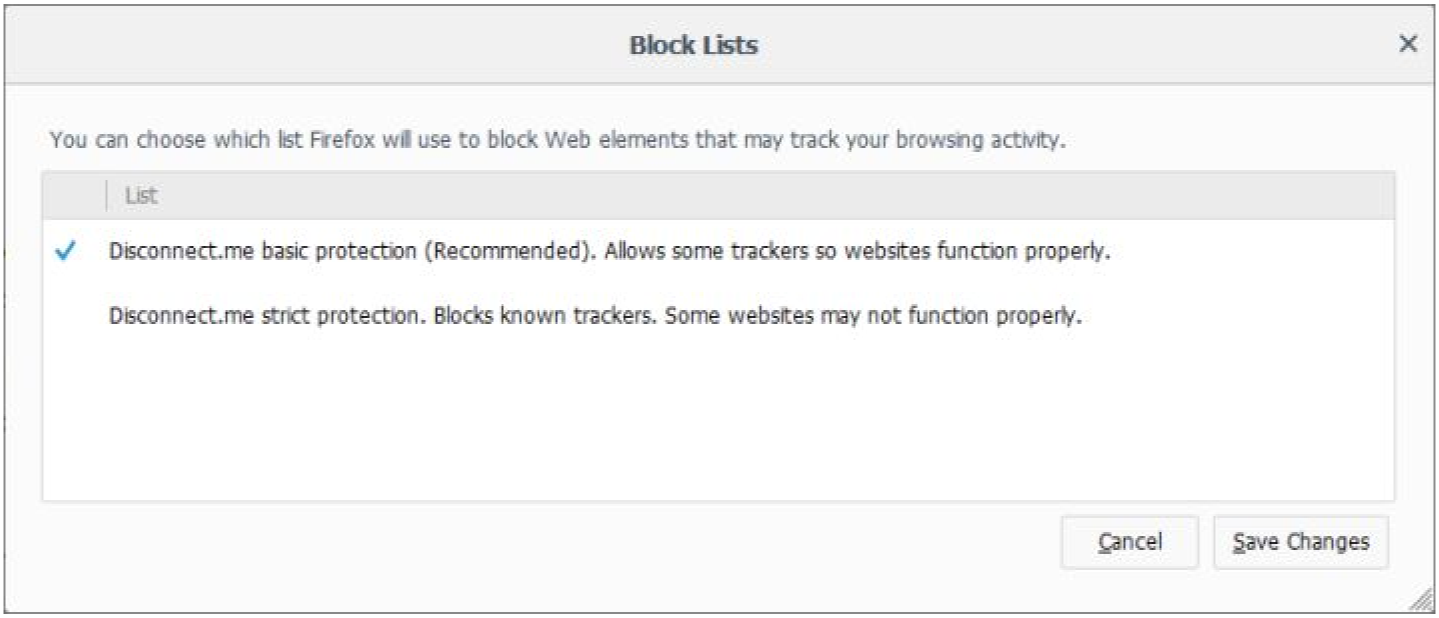
“It shouldn’t be hard to own your life online.”
That’s why Firefox gives more power to you with every update.

Built-In Tracking Controls: When you see the shield in the address bar, Firefox is blocking ads and unruly scripts from following you around the web. Select the icon to dial in the options.

Mozilla has rolled out cross-site tracking protection through third-party cookies under the name of "Enhanced Tracking Protection".

When it's enabled it will block third-party scripts from dropping cookies that allow them to track the browser across web domains. As we know, such cookies are typically use by advertisers to track a site's visitors as they move across the Internet.

Mine was properly enabled after I updated to v63, but I might have enabled it previously since I believe it is disabled by default for now. So if you enable it watch for any sites that might balk. Under the Firefox menu choose "Options" / "Privacy & Security". In the "Content Blocking" section at the top enable both "All Detected Trackers" and "Third Party Cookies"



Block Lists

You can choose which list Firefox will use to block Web elements that may track your browsing activity.

	List
✓	Disconnect.me basic protection (Recommended). Allows some trackers so websites function properly.
	Disconnect.me strict protection. Blocks known trackers. Some websites may not function properly.

Better Content Blocking = Faster Internet: Because ad trackers can do more than annoy you – they add data bloat that slows you down.

Secure Syncing Across Devices: With a Firefox Account, your data is secured with end-to-end encryption. Even we can't see what you save, sync and send.

about:preferences#privacy

A local privilege escalation vulnerability in Linux and FreeBSD:
https://www.securepatterns.com/2018/10/cve-2018-14665-xorg-x-server.html
Narendra Shinde, an Indian security researcher has discovered a flaw in X.Org X Windows Server package that impacts OpenBSD and most Linux distributions, including Debian, Ubuntu, CentOS, Red Hat, and Fedora. Exploitation of this flaw would allow a lower privileged user to create or overwrite a file or files anywhere on system, including files owned by privileged users (ex. /etc/shadow). However, the attacker would require a console session to exploit this issue.

The X.org X windows server configuration option "-logfile" is vulnerable to format string vulnerability which can be abused by a local attacker.

X.org immediately fixed the problem and issued updates. This is not super critical for the single user at home. But a local user could attack an OpenBSD or Linux system with X.org's X Server application installed. So at-risk systems should update soon since exploitation of this would give an attacker full system privileges on the target system.

Errata

An anonymous listener in Cheektowaga, NY, US observed: "As you said it, "slash dot dot" doesn't do anything special. "Slash" would be the root directory, and the parent of the root is the root, because it really has no parent. You probably meant to say "dot dot slash".

RTOS vs Embedded OS?

Miscellany

Peter Hamilton: "Salvation"

- Three threads: Now, in the far far future, and characterization by way of (lengthy) historical stories.
- In Hamiltion's "Commonwealth" reality we had wormholes… trains, etc.
- In this reality we have "quantum entanglement portals". They are so economical that they replace virtually all transportation. Freeways are turned into linear parkland. Ultra wealthy families have "homes" with "rooms" on different regions of Earth and on other planets and asteroids.
- "Threading"

SpinRite

Ben Willerson in Petaluma
Subject: It really does fix SSDs! Thank God!
Date: 23 Oct 2018 9:04:23
:
Dear Steve & Leo,
Everyone says they love the podcast. I haven't said it before, so add me to that list. My daily commute is hell. (Since Leo is in Petaluma he knows.) Although you guys don't fix that, you make it more tolerable.

My Spinright story will not be as important to you as it is to me -- it could not be. But I hope you find this since others need to know what happened. Several years ago I switched to an SSD to get for more space but also since I figured a solid state storage drive would be failure proof. What could go wrong? It cannot crash. There's nothing but reliable solid state memory inside. So I was less careful about keeping backups. I did more at first, somewhat. But everything was working until last Monday morning when I turned the PC on after the weekend and it said "Missing operating system." I had done a huge amount of work using that computer and all that data was only there since I had stopped backing up. The first thing I tried was moving the SSD to a different computer to see if I could at least get back the data. But the other computer wanted to format the SSD and it was not accessible at all. My next thought was Spinright. Like you have said, it can be used for maintenance to keep this from ever happening in the first place. But it has also recovered other people's SSDs (I was thinking about that security camera crashing story). The subject of this eMail gives away the ending. I bought Spinright from your website and it was running on that SSD within a few minutes. I don't know whether you believe in the power of prayer, but I do. I write this letter because of the relief I felt when the entire drive reappeared, with nothing lost, after Spinright finished on it.

Bless you and bless Spinright. Thank you.

Closing The Loop

Andrew Cooper in Sydney, Australia
Subject: Inside the Intranet should not be considered safe
Date: 25 Oct 2018 14:40:16
:
Steve,

First, let me say that I've been listening to Security Now almost since it began. I really enjoy the way you cover topics from the high level right down into the technical nitty-gritty.

However, something's been niggling at me for a while. There have been many times when you've been discussing a vulnerability in some protocol or service and you've implied, or explicitly said, that as long as it's not exposed to the public internet then it's okay. I heard it again today in your discussion on the Live 555 server on SN686. This may have been true in the past but it hasn't been for a while.

For many years now the larger security conscious organisations have known that it's not safe to trust the intranet. It started over 10 years ago with talk of APTs (Advanced Persistent Threats) and the "disappearing perimeter", and today has developed into a mindset that informs all security discussions and decisions. Microsoft's phrase is, "assume breach".

While this mindset is especially true for medium and large organisations, it should also be a consideration for smaller networks. Even if your employees or family members are 100% trustworthy, a well executed social engineering attack could result in some code running on a PC, or even a phone, that can then use a vulnerability in an internal service to dig its way deeper into the network and pivot to other targets.

Of course this mindset needs to also consider the value of potential targets, and the likelihood of attack. It's probably overkill for most home networks, and even some small offices, but I don't think it's helpful to encourage the thinking that if it's being the firewall it's okay.

That said, thank you for all the hard work you put into the podcast every week. Please keep it going.
Regards,
Andrew

(Phishing as a way into the Intranet...)

Securing the Vending Machine

What looks good but doesn't work…

- No WAN communication required for spending money from the smartphone. (BLE/NFC)
- No Internet connection required for the vending machine.
- WAN communication is only required with headquarters to reload the wallet.
- Presumably the Vending machine could also reload the wallet.

All vending machines and headquarters share a single secret symmetric key.
The wallet is either encrypted under the secret key or is signed by it.
In either case this simple expedient prevents the user from altering the wallet's contents.
Since vending machines are susceptible to physical attack, their shared secret is usable but unextractable from an HSM (like any TPM) chip. These are now readily available and add negligible cost to the endpoint.

Why does this fail?

Because of the "double spending" problem: We have no way of preventing the phone from taking a pre-purchase snapshot of its wallet, having the vending machine decrease the balance in the phone, then reset the wallet to its pre-purchase state. The phone cannot "see" into and/or modify the wallet itself, but it doesn't need to. It simply reverts to its previous state with the pre-purchase balance.

What this means is that the phone **cannot** be the holder of the user's balance. The balance **must** be at must be external to the phone.

IF every vending machine had a connection to the Internet, then this sort of double spending could be easily prevented by having the vending machine reach around the user to contact headquarters. But in this case, the fact that the hacker in our original story was able to hack his app's wallet strongly suggests that the app's developers did not have access to "connected" vending machines and were attempting (and failing) to work around that lack.

So… we now stipulate that the vending machines are NOT connected to the Internet. How do we solve this problem in a way that prevents double spending?

Let's assume that we only have the minimum communication links we have already defined:

- We have the phone's inherent Internet connection and thus able to contact headquarters.
- And we have the vending machine with a Bluetooth LE and/or NFC link to the phone.

So the vending machine is linked to the phone, which is, in turn, linked to home base. In other words, the user's phone can serve, and, indeed, must serve, as the intermediary between the vending machine and central headquarters where the user's balance is securely stored.

The user selects an item to be purchased from the vending machine either through the vending machine's traditional physical UI (push buttons) or the phone app if we want to get extra fancy.

The vending machine generates a nonce and a description of the transaction which it signs with a globally shared secret key. As before, it’s possible to robustly protect keys embedded in hardware. (Note that the vending machine might also send its current physical inventory along as an added bonus, thus informing headquarters of any pending need for restocking.)

The vending-machine-signed transaction packet is squirted to the phone over one of the two local RF links, which the phone forwards to headquarters. The phone can examine the packet for confirmation if it wishes, but it’s unable to tamper with it since it will never have the required secret key.

Headquarters accepts the transaction description, debits the user’s account, and returns the user new balance (for the phone’s app) along with the transaction with an approved flag (containing the vending machine’s nonce) signed with their shared secret for the user’s phone to return to the vending machine.

The user’s phone forwards the approved and signed transaction packet to the vending machine, retrieves the candy bar… we’re finished.

Note that the presence of a vending machine generated nonce prevents the phone from reusing headquarter’s transaction approval the next time the user wants to purchase a candy bar. Without a single-use nonce, the headquarters approved and debited transaction might be reused. So each nonce issued by the vending machine is single-use. It is placed onto a “pending approval” list and is deleted upon its return and use.

~30~