elastic security labs

Global Threat Report

Vol.1 2022

Contents

Introduction 3

Executive Summary 4

Trends & Correlations.............. 5

    Malware Signature Trends 5

    Endpoint Behavior Trends 13

    Cloud Security Trends 22

Threat Profiles 28

    BLISTER [REF7890] 29

    PHOREAL [REF4322] 31

    CUBA [REF9019] 33

    QBOT [REF3726] 35

Forecasts & Recommendations....... 37

Conclusions....................... 40

Introduction

"The future of security is open. In a world of dynamic, fast-moving, and well-resourced threat actors, digital security's best hope lies in bringing like-minded defenders together around platforms that are as open and inter-operable as possible."

– Nate Fick

U.S. Ambassador at Large for Cyberspace & Digital Policy and Formerly Elastic Security General Manager and Endgame CEO

This Elastic Global Threat Report is a product of Elastic Security Labs, our threat research branch with expertise in investigating computer network intrusions, analyzing malicious software, developing mitigations for broad categories of threats, and conducting intelligence analysis. Elastic Security Labs is a group of passionate security professionals who research security topics to improve the Elastic Security product and share what we learn with the broader community.

Our philosophy is straightforward: the best way to protect the world's data is by weaponizing defensive technologies. We create environments that are hostile to threats because that is the single most effective way to change the threat landscape. While many security vendors choose a passive, "wait-and-see" mentality, threats are constantly adapting and evolving, thereby demanding a more proactive approach.

This report describes threat phenomena, trends, and recommendations we believe will help organizations prepare for the future. Elastic discloses malware research, attack patterns, and clusters of malicious activity to the community — summarized in this inaugural report.

Throughout this report, we observe that financially motivated threats are the most active, and the groups responsible for them are acting with increasing speed. These rapidly expanding threats impact organizations that struggle with mitigation in their environments, resulting in bigger wins for adversaries.

Elastic telemetry, voluntarily shared and enriched with cutting-edge innovations, as well as public and other third-party data, provides the data-validated material for this report. Information has been responsibly sanitized to protect the identities of customers, where applicable.

Elastic primarily uses telemetry to improve feature efficacy and to provide organizations with additional security context through publications such as this. We welcome the opportunity to partner with our customers in this way to analyze their data, anonymously sharing what we learn with the larger security industry.

In order to effectively prevent cybersecurity threats, an organization needs visibility, capability, and expertise. Elastic Security delivers this foundation, and our global instrumentation allows us to quickly deploy community protections against threats. This report contains information about the threats we see and respond to — such inputs are essential for developing future Elastic features.

By sharing these insights, we at Elastic Security Labs hope to normalize the discussion of vendor visibility and demonstrate how our unique perspective empowers the developers of security technologies to maximize positive outcomes for their users and the community at large.

Executive Summary

For many, the current state of security is frustrating — an endless stream of vulnerability, exploitation, compromise, and theft. Elastic is frustrated with this state as well, and that's one reason we've chosen to work on a cure instead of just monetizing new treatments for the many expensive and disruptive symptoms.

> *We've chosen to work on a cure instead of just monetizing new treatments for the many expensive and disruptive symptoms.*

As detection and prevention technologies have experienced a dramatic increase in effectiveness, information sharing has risen to an all-time high. Even the broader public understands the implications of an ever-broadening global threat landscape and the implicit challenges faced by the stewards of cybersecurity.

Threats of all kinds have adopted new capabilities and methods while increasing their cadence of activity. As organizations have tracked decreasing mean-time to detect (MTTD) and mean-time to remediate (MTTR) metrics, threats have acted with even greater speed to undermine those efforts.

Adversaries aren't unaware of intelligence-led efforts to track, expose, and stop them, either. Instead, many financial threats have established affiliate programs and other proxy relationships that ensure they continue to make money while disambiguating themselves from government sanctions — one of few costs they seem respectfully wary of. Unfortunately for their victims, these factors rarely have a direct impact on the ground where they live and work.

Elastic Security Labs observes that a significant percentage of all threats achieve a degree of success against technical, procedural, and human mitigations. We have observed threats with mature software development and research capabilities of their own, bypassing sophisticated endpoint detection and response (EDR), antivirus, network intrusion detection systems (NIDS), and directory service policy controls. Compromised service providers, software supply chains, and built-in operating system frameworks like Berkeley Packet Filter (BPF) are no longer far-fetched fictions, but are today's attack surface.

Elastic customers and users will want to understand the contents of this report, and how our visibility of the global threat landscape affects them.

Our team is comprised of incident responders, malware and intelligence analysts, security engineers, researchers, data scientists, and other experts with decades of collective experience, and we're excited to share our knowledge with the community.

Trends & Correlations

The following trends represent the major tools, tactics, and procedures employed by threats and were identified across Elastic telemetry. Because each vendor has their own visibility, reports such as this one offer valuable insights into the methods each uses to monitor and mitigate threats. Elastic telemetry incorporates data from Elastic Endgame, Elastic Endpoint, and the Elastic Security solution, and deploys mitigations via those technologies.

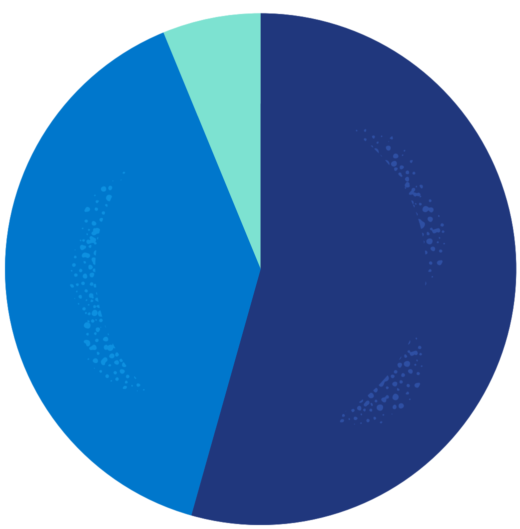
Below we present an overview of trends and correlations, with subsections for malware signatures, endpoint behaviors, and cloud security capabilities. Where applicable, those sections have been further organized for the reader.

Malware Signature Trends

YARA signatures provide one layer of defense within the Elastic Security solution, identifying malware-related threat activity based on strings or byte-sequences. Elastic Endpoint Security offers signatures at the file- and memory-level for all common endpoint operating systems. Elastic makes signatures available to the community through the protections artifacts repository as part of our free and open commitment.

To start, Elastic Security Labs began analyzing specific operating system (OS) trends for malware according to our telemetry. From this, we identified that ~54% of all malware infections were on Windows endpoints, while ~39% were on Linux endpoints.

~54% of all malware infections were on Windows endpoints, while ~39% were on Linux endpoints



Malware **by Endpoint OS**

Windows...... 54.4%

Linux........ 39.4%

MacOS........ 06.2%

Figure 1: *Malware by Endpoint OS*

¹ The Elastic Security solution telemetry is generated by a diverse population of sensors and data sources which are too numerous to describe concisely, including sensors not developed by Elastic.

As corporations continue to adopt a hybrid-cloud approach and implement more Linux-based endpoints as backend infrastructure, this creates the possibility for adversaries to weaponize binaries for this architecture and distribute them via their custom delivery techniques.

Diving a bit deeper, we identified that the largest contributor of Linux-based malware/payloads was Meterpreter at ~14%, followed by Gafgyt at ~12%, and Mirai at ~10%. While this comes as no surprise, we can confidently say adversaries are still using frameworks such as CobaltStrike and Metasploit to deploy payloads, target exploits, and set up backdoors for further command execution. This proves especially useful if a shell is established on a Linux endpoint hosted within a cloud service provider (CSP) where the default CSP CLI is installed and prebuilt scripts or modules can be called for easy discovery and enumeration.

While we do not go into detail about BPFDoor, Elastic Security Labs has published extensive research on this Linux implant, as well as a communication client binary for leveraging the implant as well.

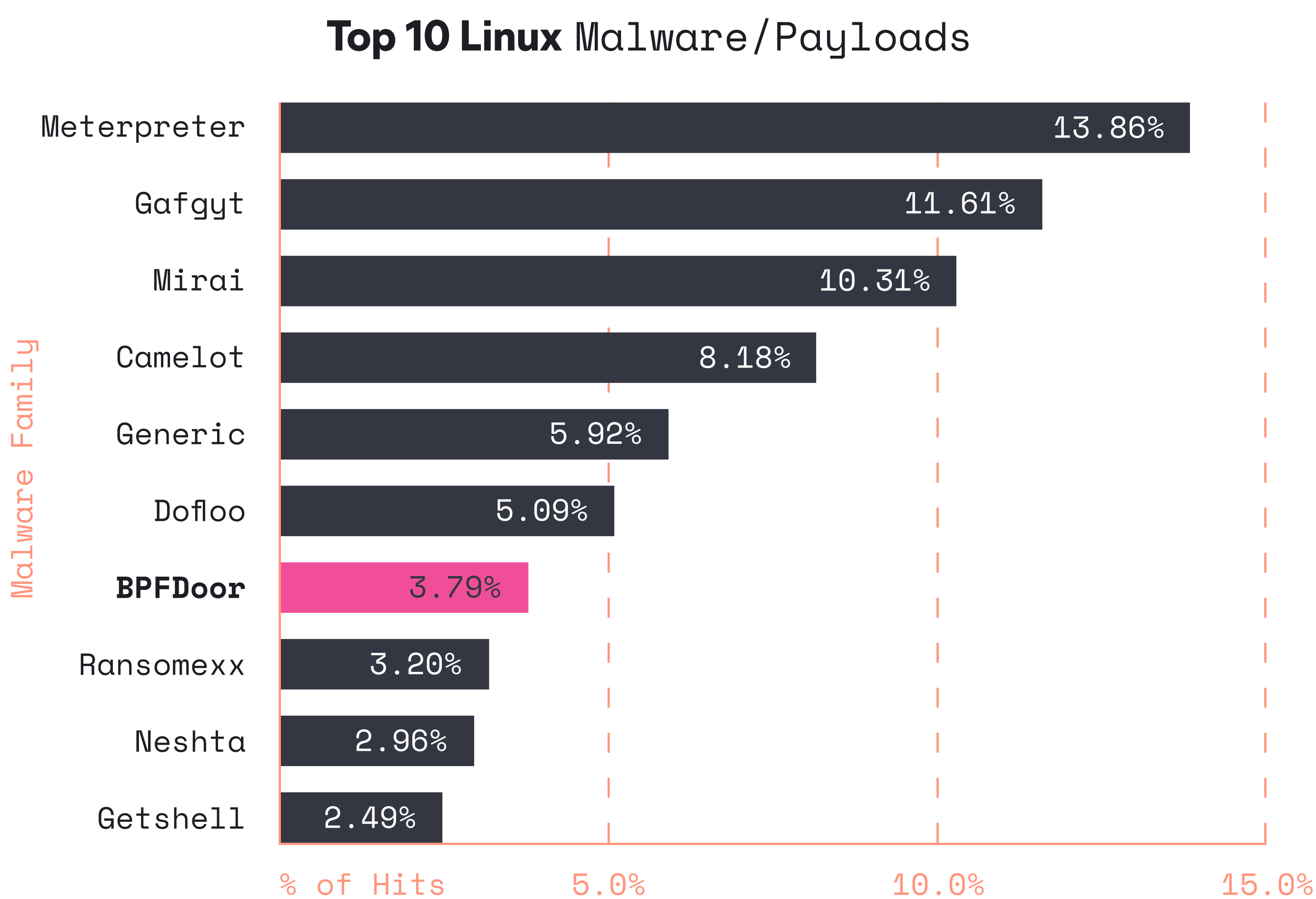


Figure 2: *Top 10 Linux malware and payloads depicting BPFDoor activity increasing*

Malware **by Category**

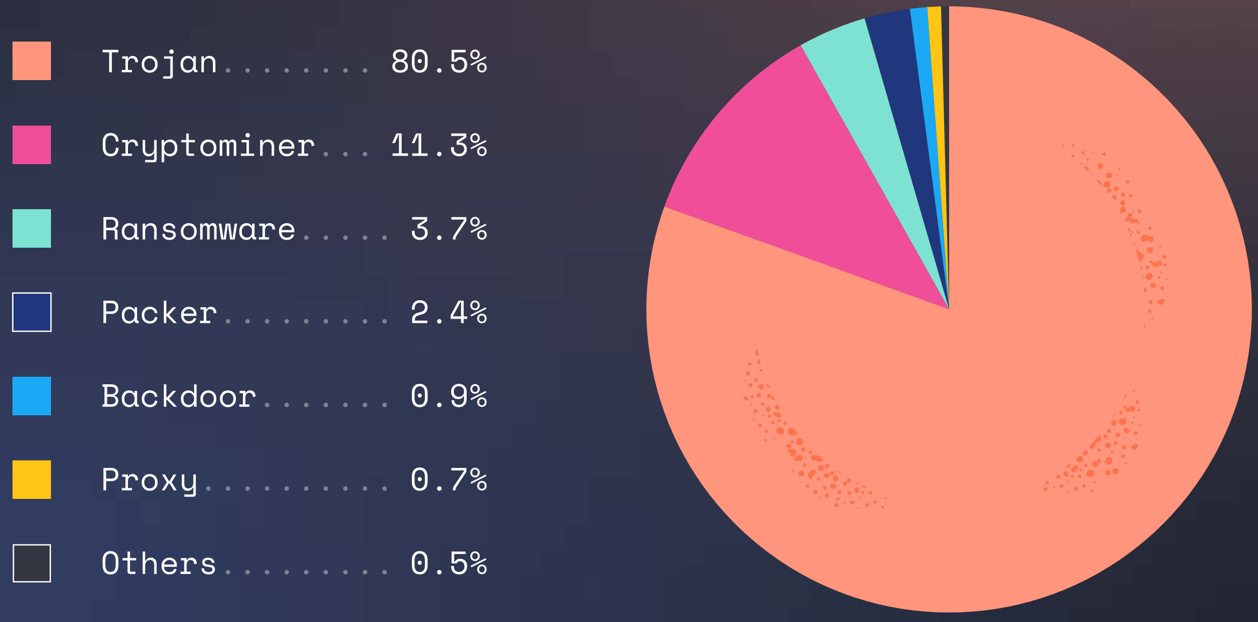


Figure 3: *Malware by Category*

Following this, Elastic Security Labs took a look at what malware categories we were seeing globally where we discovered ~81% of malware were Trojan-based, followed by Cryptominers at ~11%. Trojan's continue to be a favored way to weaponize deliverable binaries that deploy stagers and droppers to carry out the intrusion, but can be multi-purposed with additional techniques. Our team has commonly seen Trojans packed before delivery to the target to avoid potential mitigation by signature-based detection engines.

Cryptominers, while not inherently malicious, were often initially deployed as a means to use a victim's computing resources to mine a cryptocurrency of choice — often Monero because of anonymity purposes. Cryptominers are commonly deployed alongside other malware families as a contingency plan for financially motivated actors if all else fails.

While XMRig and KWorker are very common tools, cryptominers often abuse them because of public source code availability. Elastic Security Labs found another prevalent cryptominer family recently: LoudMiner. Shown in the graph below, LoudMiner detections were minor until November of 2021, followed by a spike in detections in February of 2022 where detections have since remained consistent. LoudMiner is based on popular XMRig open-source code and designed to mine Monero cryptocurrency, which includes additional features for anonymity. While detections were on Linux endpoints specifically, LoudMiner is cross-platform and utilizes the CPU for ledger transaction processing on the Monero blockchain.

Elastic Security recently open-sourced our custom YARA and endpoint behavior rules in our Protections Artifacts repository, where Linux.Cryptominer.LoudMiner logic can be found.

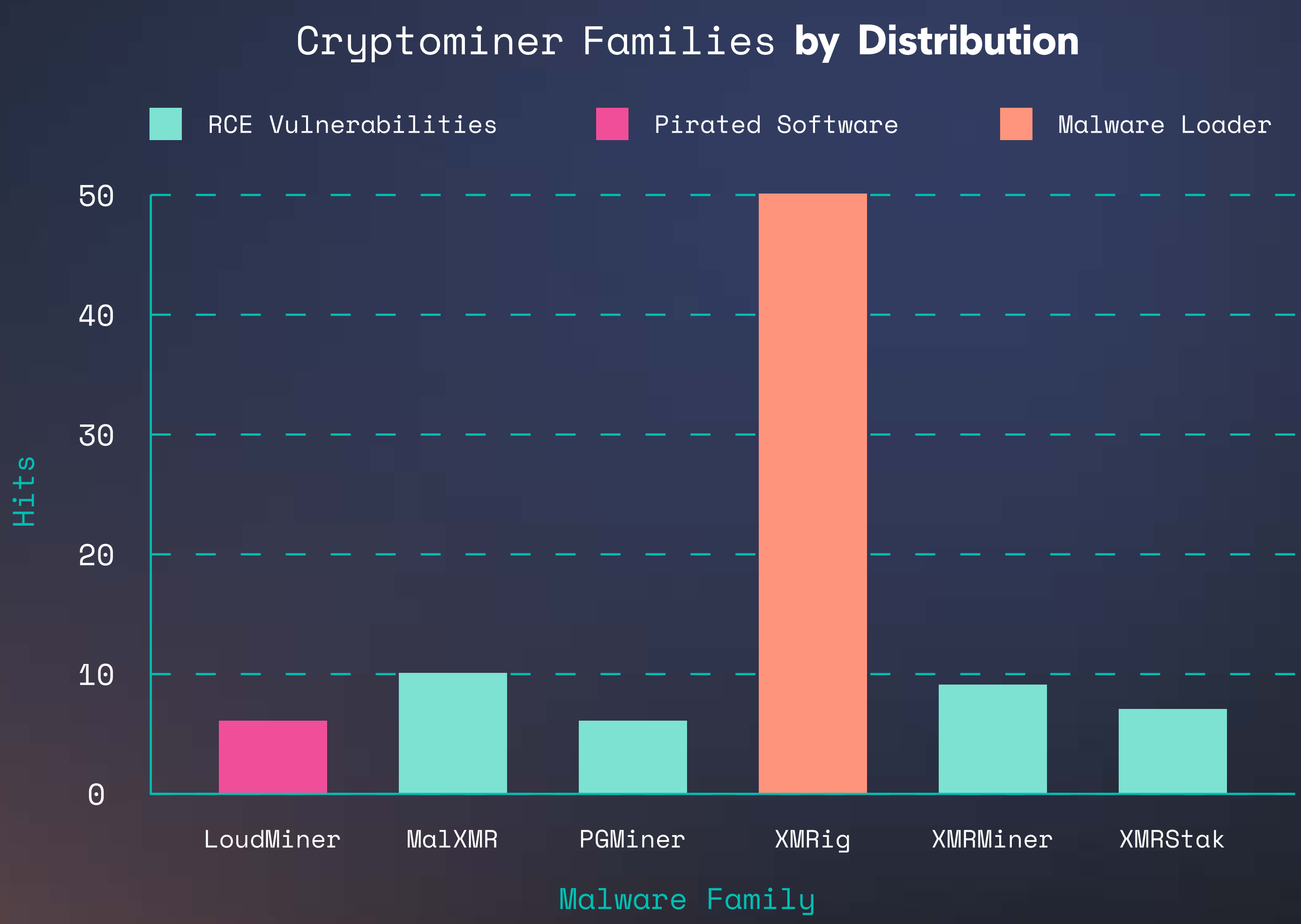


Figure 4: *Cryptominer Families by Distribution*

While continuing to review trends in malware categories, we noticed a few increases in less-commonly observed types such as backdoors and proxy payloads, as shown in the graph below. Between March and May of 2022, Elastic Security Labs noticed an increase in backdoor-related detections for various customers corresponding with our custom YARA signatures listed below.

- Linux.Backdoor.Bash
- Linux.Backdoor.Generic
- Linux.Backdoor.Tinyshell
- MacOS.Backdoor.Applejesus

Linux backdoors allow adversaries to have continued persistence and access to a compromised endpoint. In cloud-based environments, initial access may be more attainable via public-facing application exploits where the backdoor is deployed, followed by cloud environment discovery and enumeration. Access to Windows endpoints would be more attainable if leveraged within the same or adjacent VPC network setups.

elastic security labs

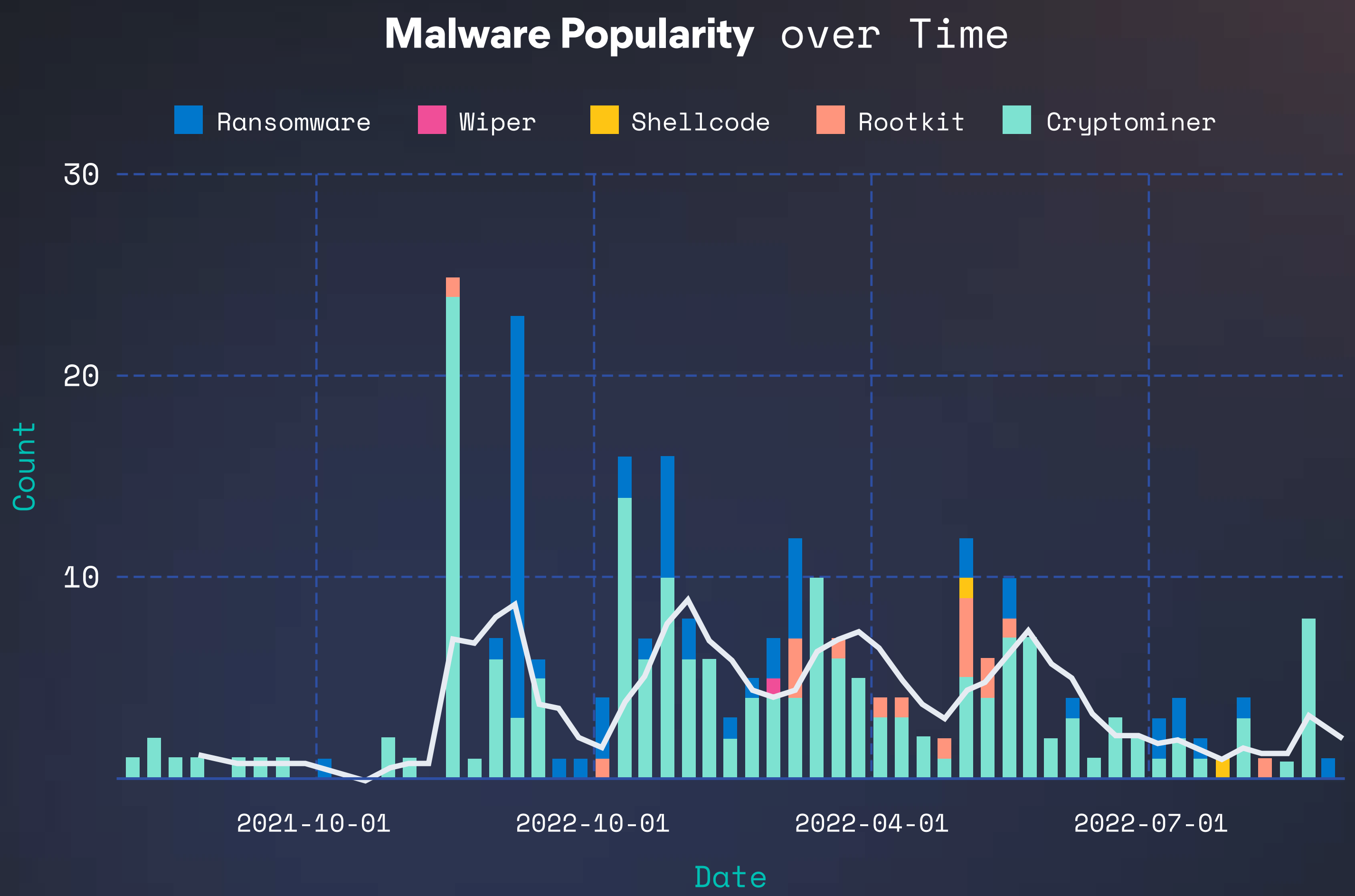


Figure 5: *Malware Popularity over Time*

Based on *Figure 5*, Elastic Security noticed a slight increase in Linux binaries with the capability to leverage a proxy for potential command and control purposes, thanks to our [Linux.Proxy.Frp] logic. When targeting Linux endpoints, adversary playbooks often include the use of a backdoor binary, as previously discussed, followed by the installation of a proxy server for command and control. This may become a more common occurrence as hybrid-cloud environments leverage more Linux backend servers with misconfigurations or poor security implementations that are publicly accessible.

As stated in *Figure 3*, Trojans were responsible for ~80% of malware detections at Elastic, a majority of which were Windows related. To dive a bit deeper, we decided to analyze trojan family popularity for Windows endpoints as shown in *Figure 6*.

Trojan Popularity for Windows Endpoints

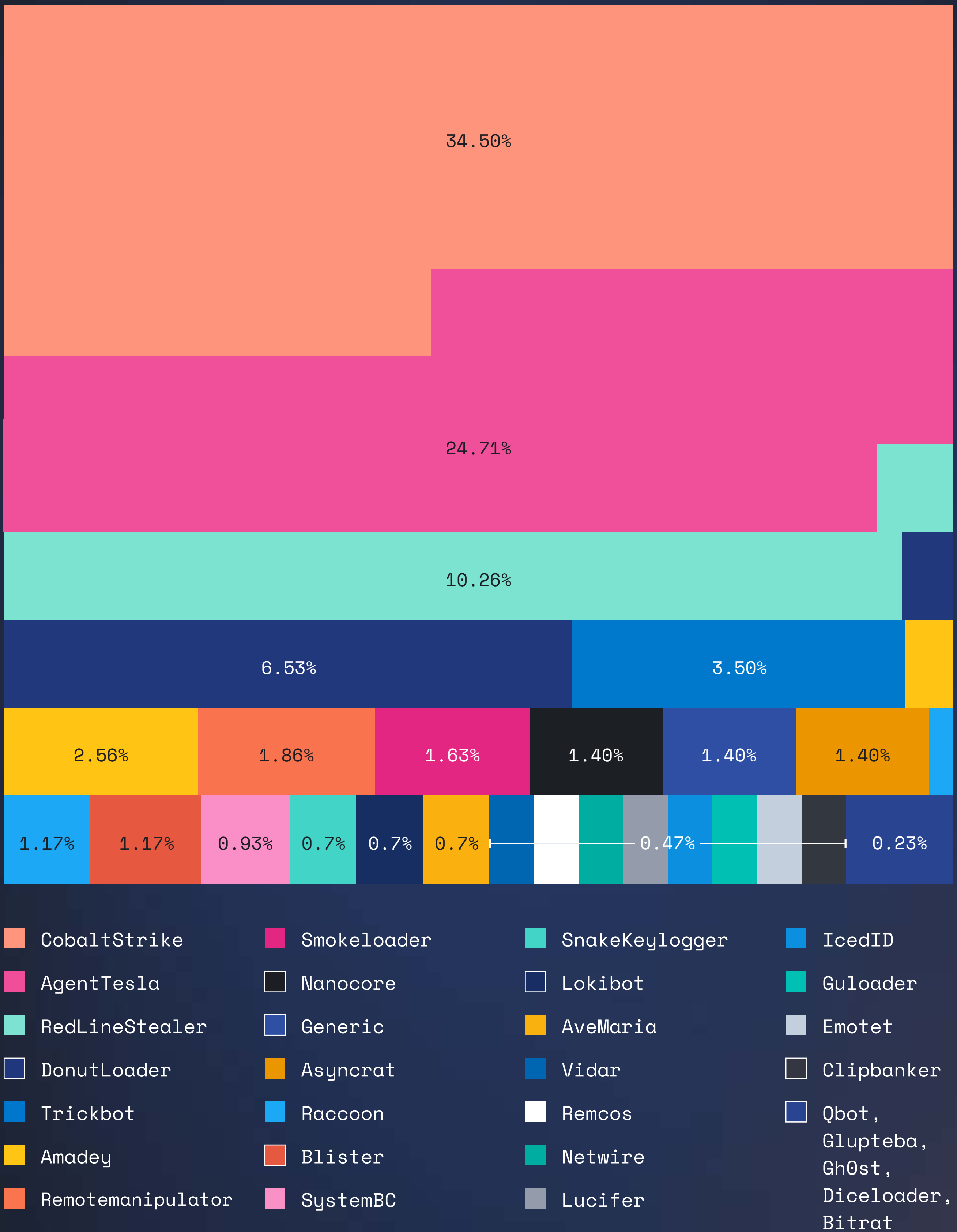


Figure 6: *Trojan Popularity for Windows Endpoints*

Commercial off-the-shelf (COTS) malware families like COBALTSTRIKE and METASPLOIT were strongly represented with in-memory detections, as well as malicious tools and mass-malware implants. To no surprise, CobaltStrike was the most popular malicious binary or payload for Windows endpoints with ~35% of all detections, followed by AgentTesla at ~25% and RedLineStealer at ~10%. Elastic Security Labs has previously analyzed CobaltStrike in depth and [discussed] the use of this tool and payloads. To continue pushing forward with openness and transparency, we also [released] a CobaltStrike beacon extractor for use. Not surprisingly, offensive tooling such as COBALT STRIKE, METASPLOIT, and MIMIKATZ continue to top our list, showing no slowdown in the usage of these tools.

While AgentTesla has been very popular as a keylogger and weaponization by adversaries continues to include additional features, Elastic Security Labs also wanted to highlight SnakeKeylogger as well. Both keyloggers are actively distributed through email as malicious attachments. As previously discussed in this report, initial access via Microsoft Office documents is commonly used to distribute these malware families, followed by proxy execution from a signed and trusted binary. Credentials targeted are often related to email and FTP clients, web browsers, and more. In an environment where a hybrid cloud solution is present, this may allow for valid accounts to be used for lateral movement into the cloud environment, where actors may look to access critical or sensitive resources and data.

AgentTesla and SnakeKeylogger signatures:

- `Windows.Trojan.SnakeKeylogger`
- `Windows.Trojan.AgentTesla`

Related to keyloggers, another malware family of interest that continues to trend is RedLine Stealer, detected at Elastic Security Labs as `Windows.Trojan.RedlineStealer`. This information stealer is also commonly distributed in malicious spam (malspam) campaigns, and became popular during the COVID-19 pandemic where email subjects and themes were tied to this popular topic — often encouraging users to download and execute attachments. RedLine Stealer continues to be adopted and developed, allowing financially motivated actors to also target hot cryptocurrency wallets (if present), as an alternative to the popular cryptominers we discussed previously. As smart contracts and blockchain technology continue to gain popularity and prove useful for record-keeping and distribution tracking via immutable ledgers, Elastic Security Labs forecasts this capability to be commonly weaponized into more information stealer families.

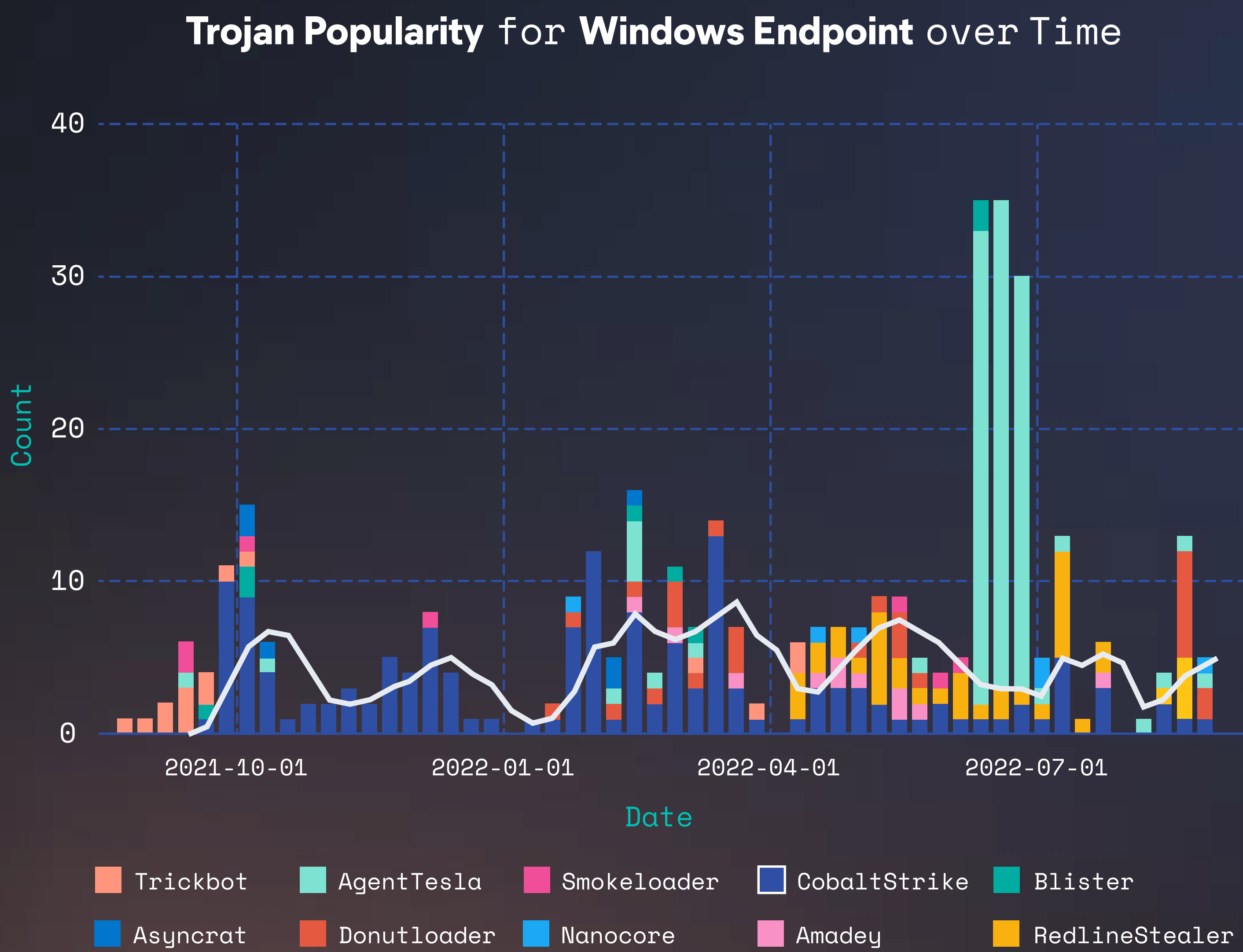


Figure 7: *Trojan Popularity for Windows Endpoints over Time — Excluding AgentTesla, Redline Stealer, and CobaltStrike*

While not as popular in recent public research, Elastic Security Labs continues to detect the abuse of the open-source loader framework, [Donut](). In research by BitDefender, the injected shellcode from an Orcus RAT binary was detected as the loader from Donut during zero-day exploitation of the Log4j2 vulnerability, which Elastic has covered in a separate [publication](). Elastic Security finds DonutLoader interesting because of its diverse capabilities to load malicious payloads and do in-memory execution of VBScript, JScript, EXE, DLL files, and dotNET assemblies. As shown in *Figure 7*, DonutLoader remains consistently popular over time, whereas the use of popular commodity malware families like Trickbot and Emotet are not detected as often.

For MacOS file signatures, MacKeeper ranked the highest at ~48% of all detections, with XCSSet in the second-place position at not quite 17%. MacKeeper is a utility software suite for macOS endpoints designed to help optimize resources and monitor internal resources. While its initial purpose is to aid MacOS users, often it can be abused by adversaries since it already has extensive permissions and access to processes and files.

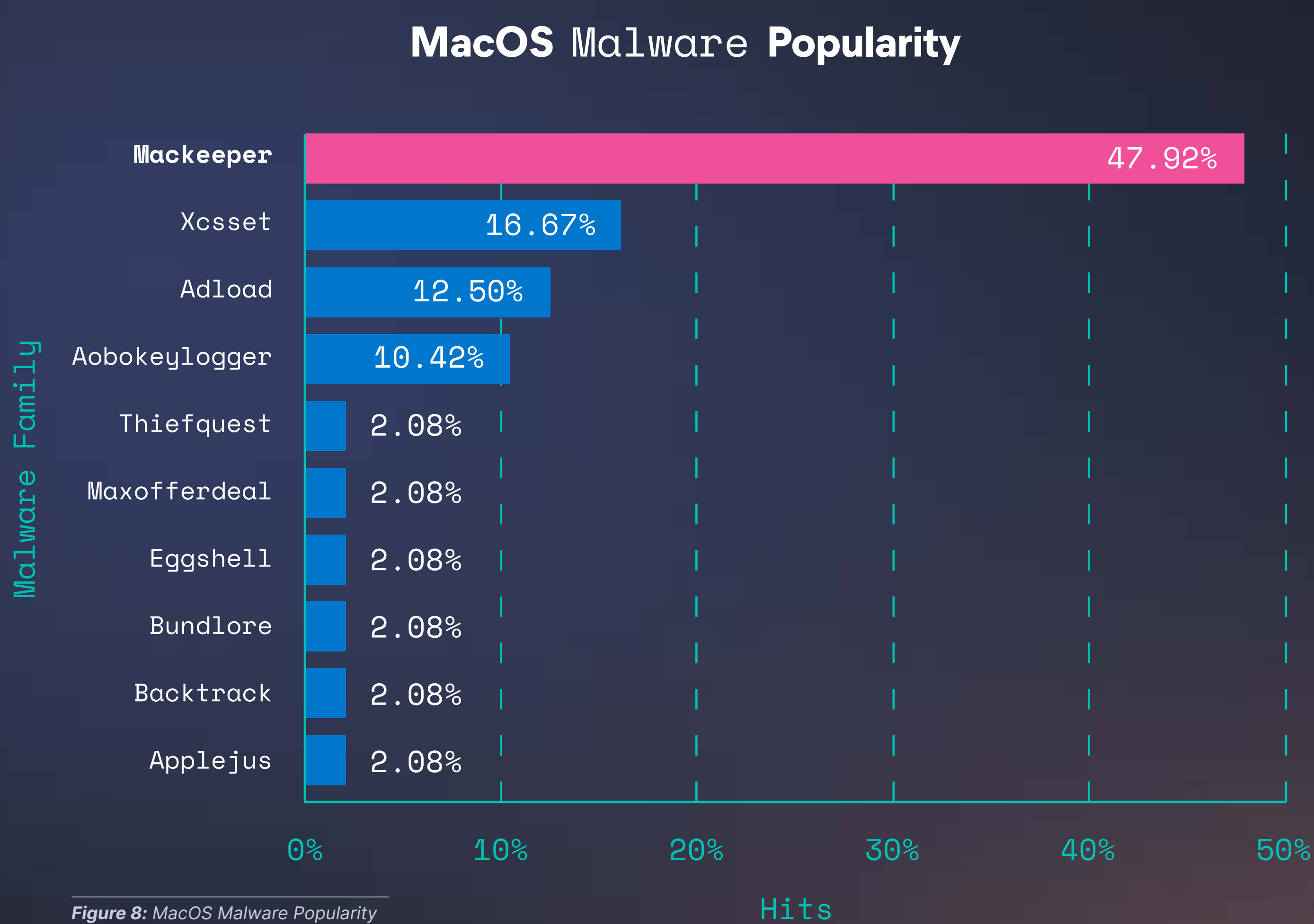


Figure 8: *MacOS Malware Popularity*

While cryptominer installation and use often targets Linux and Windows endpoints, Elastic Security Labs has observed some use on macOS endpoints, as well with variants such as mshelper and CreativeUpdater being popular. It should be noted that the distribution and victimology of macOS cryptominers could become increasingly popular and developers leverage MacOS and JavaScript for work-related tasks. Since Node Package Manager (NPM) is a common package manager for JavaScript, cryptominers could be distributed in malicious packages to macOS endpoints where development work is conducted, thus contributing to the popularity.

Endpoint Behavior Trends

Openness, transparency, and collaboration are at the heart of Elastic Security. As we continue to uphold those principles, our recent publication of protections artifacts shared endpoint behavioral logic that we developed to identify adversary tradecraft using Elastic. For this report, we have gathered global telemetry on the alerts and built-in prevention capabilities from this detection logic.

Starting with MITRE ATT&CK® mappings to our endpoint behavior rules, Elastic Security Labs found that ~34% of alerts fell within the defense evasion bucket, followed by execution at ~22% and credential access at ~10% as shown below.

This indicates *the role that defense evasion plays in not just targeted attacks, but all attacks.* In addition to bypassing security instrumentation, techniques in this category also bypass visibility, *resulting in longer dwell times for threats and greater successes.* It also suggests that defense evasion has become necessary for threats that assume security instrumentation will be present.

Endpoint Signals by Tactic

- Defense Evasion.........33.6%
- Execution...............21.7%
- Credential Access.......10.3%
- Persistence..............8.6%
- Command and Control......7.4%
- Impact...................6.2%
- Privilege Escalation.....6.0%
- Initial Access...........3.7%
- Discovery................1.7%
- Lateral Movement.........0.8%

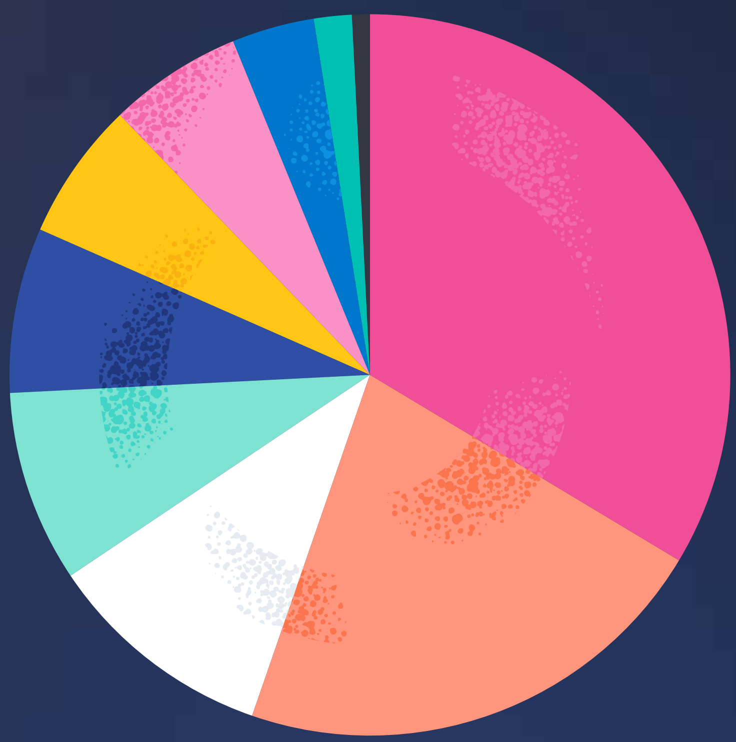


Figure 9: *MITRE ATT&CK Tactics for Endpoint Behavior Rules*

Defense Evasion

The most significant technique contributing to defense evasion was masquerading and system binary proxy execution for a combined 72% of all defense evasion techniques. System binary proxy execution describes built-in and often legitimately signed utilities that can be co-opted by threats to run malicious software such as malware. Many procedures for the known sub-techniques are known, with some security technologies struggling to identify the software they are responsible for running.

Masquerading as an otherwise legitimate process is yet another common technique used for defense evasion, seeking to evade security technologies that inspect running software, scripts, or code.

Diving deeper into specifics, Elastic found that Rundll32 continues to be heavily abused in different stages of attacks affecting Windows systems. Adversaries favor this sub-technique as it plays into the living-off-the-land (LotL) methodology by which an adversary crafts malicious DLLs that can execute from a trusted and signed Windows binary. This utility is heavily abused during many phases of the attack lifecycle.

Additionally, Regsvr32 is another commonly abused native Windows binary contributing to the high percentage of defense evasion alerts. Typically, Regsvr32 is abused from within malicious Microsoft 365 documents (maldocs) to execute malicious DLLs or register malicious files.

Technique	Signal Percentage
Masquerading	44.29%
System Binary Proxy Execution	30.00%
Access Token Manipulation	12.32%
Process Injection	7.62%
BITS Jobs	4.74%
Trusted Developer Utilities Proxy Execution	0.90%
XSL Script Processing	0.66%
Impair Defenses	0.65%
Exploitation for Defense Evasion	0.64%
System Script Proxy Execution	0.13%
Modify Registry	0.03%
Indicator Removal on Host	0.01%

Table 1: *Defense Evasion Techniques*

High priorities to monitor include mshtml, msiexec, svchost, wefault, wemgr, and runtimebroker applications. Organizations should also scrutinize standard Windows utilities when renamed or when executed from nonstandard directories — common methodologies. It isn't uncommon for adversaries to directly and indirectly tamper with Windows Defender.

Initial Access and Execution

Initial access and execution are often proximate to one another during an intrusion, with a clear relationship between stages of the intrusion.

During analysis of initial access techniques, Elastic Security identified a significant proportion of alerts from Windows endpoints that came from environments where Microsoft Office was deployed and emails with malicious links or attachments were received. Weaponized lures of several kinds (e.g., attached document, ISO object, LNK file, script object) continue to represent the most common approaches to initial access. These techniques target the recipient and most often rely on their cooperation to succeed.

> *Readers should note that decisions by Microsoft to disable macro support by July 27, 2022, made LNK and ISO objects more reliable for weaponized payloads than document lures. We forecast this to be likely, and widely distributed malware families like ICEDID are already adopting this approach.*

The following table depicts the most common initial access and execution techniques.

Rule Name	Signal Percentage
Suspicious Microsoft Office Child Process	34.00%
PowerShell Obfuscation Spawned via Microsoft Office	16.33%
RunDLL32/Regsvr32 Loads Dropped Executable	13.47%
Suspicious Execution via a Mounted Image File	5.24%
Execution from a Downloaded ISO File	5.00%
Microsoft Equation Editor Child Process	3.66%
WMI Image Load via Microsoft Office	2.93%
Suspicious Execution via Microsoft Office Add-Ins	2.56%
Potential Remote File Execution via MSIEXEC	2.44%
Initial Access or Execution via Microsoft Office Application	2.44%

Table 2: *Top 10 Triggered Elastic Initial Access Rules*

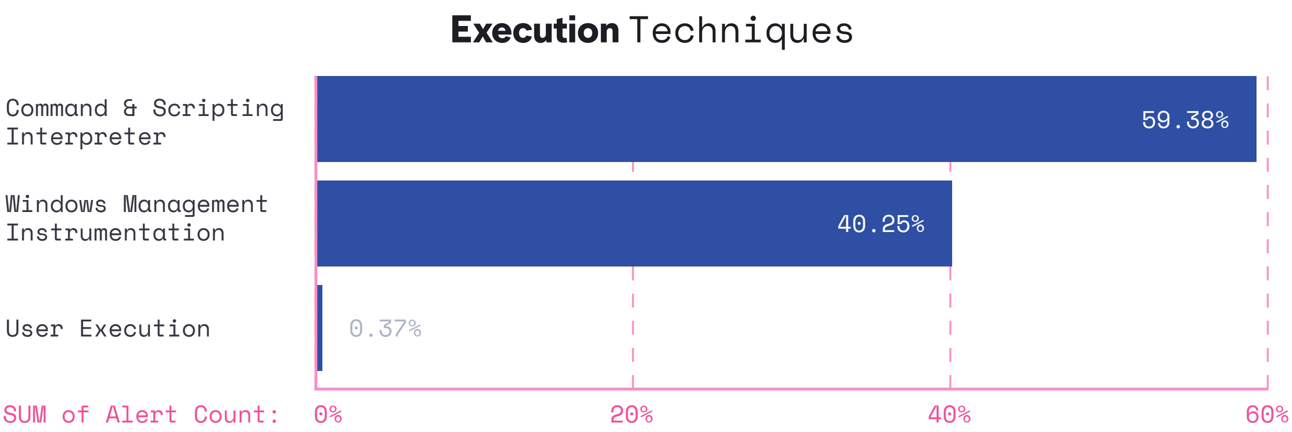


__Figure 10:__ Execution Techniques

Regarding execution, Elastic Security Labs found that ~59% of execution techniques related to command and native scripting interpreters, followed by 40% attributed solely to Windows Management Instrumentation (WMI) abuses. This is especially true for Windows endpoints where adversaries abuse PowerShell, Windows Script Host, and Windows shortcut files to execute commands, scripts, or binaries.

As shown below, again we see *RunDLL32*, a native windows binary, being abused by adversaries during later stages of an intrusion. Additionally, Elastic Security commonly observes that the Windows Script Host (`wscript.exe` or `cscript.exe`) is used to run malicious code in Visual Basic Script (VBS) and JavaScript (JS) files.

Rule Name	SUM of Alert Count
Command Shell Activity Started via RunDLL32	27.53%
Execution of a Windows Script with Unusual File Extension	25.95%
Suspicious Windows Script Interpreter Child Process	13.51%
Execution from Unusual Directory	11.56%
Suspicious Windows Script Process	9.07%
Suspicious PowerShell Execution via Windows Scripts	4.21%
Unusual PowerShell Engine ImageLoad	2.88%
Execution of a File Written by Windows Script Host	1.91%
Windows Script Execution from Archive File	0.63%
Execution of a Windows Script Downloaded via a LOLBIN	0.60%

__Table 3:__ Triggered Elastic Execution Rules

Credential Access

With the continued trend in hybrid-based deployment environments between on-premise hosting and Cloud Service Providers (CSPs), adversaries continue to rely on valid accounts as these accounts draw less suspicion to administrators. Coupled with living-off-the-land (LoTL) techniques, defenders may struggle to distinguish between expected and potentially malicious activity. As a result, credential access is not only a common tactic on endpoints, but in CSP environments as well — highlighted in the global findings section of cloud security trends.

~77% of all credential access techniques are attributed to OS credential dumping with commonly known utilities, also known as "dumping tools" as they allow the adversary to dump credentials in the form of a hash or clear text from the operating system.

Credential Access Techniques

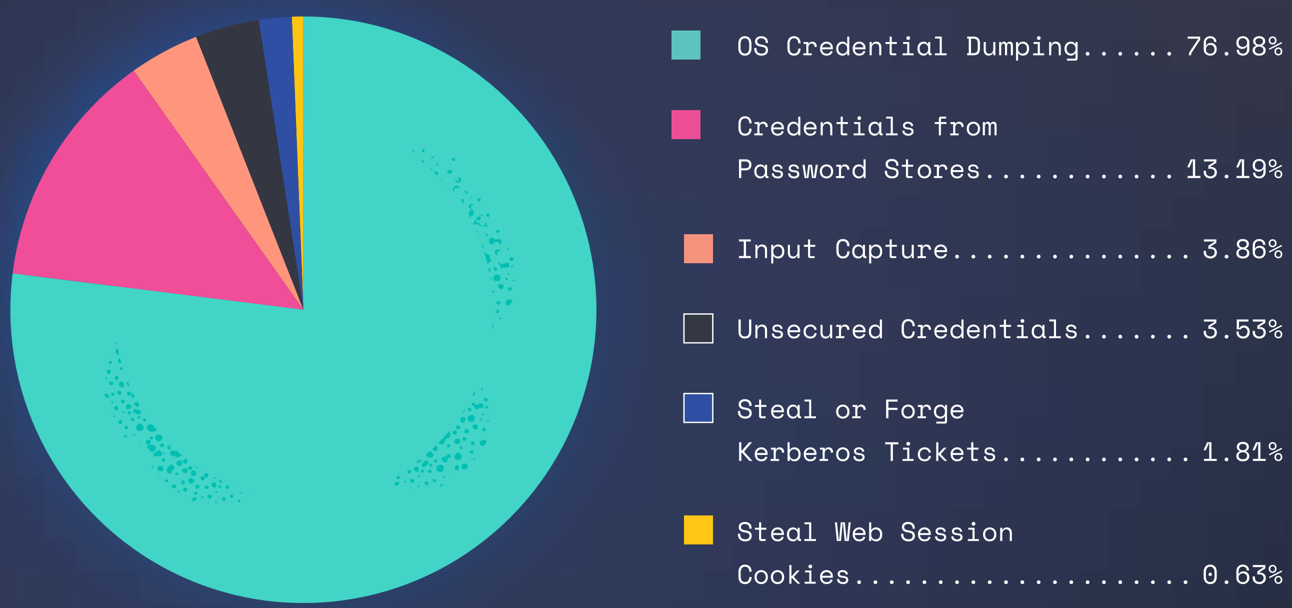


Figure 11: Credential Access Techniques

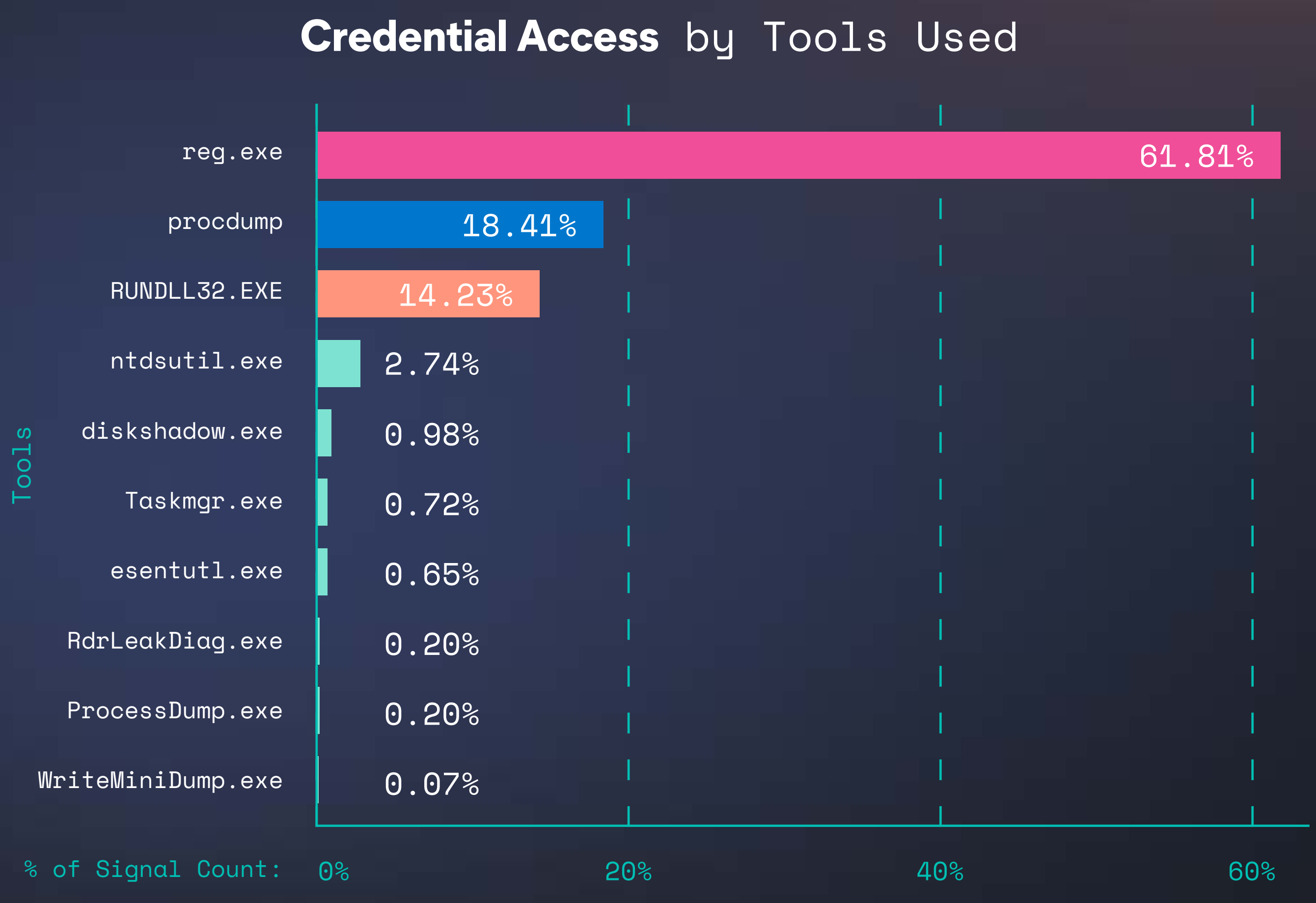
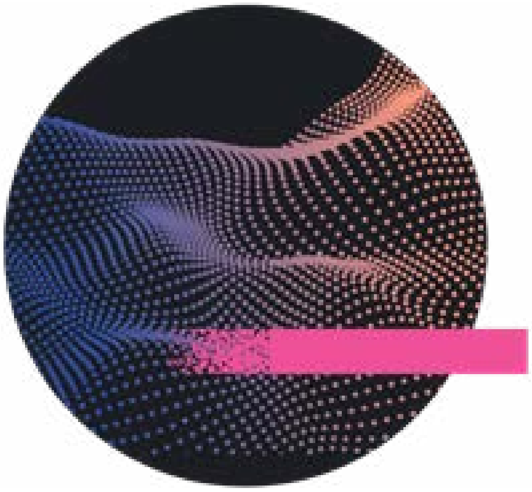


Figure 12: *Credential Access by Tools Used*

As shown below, reg, procdump and RunDLL32 are strongly represented in our data due in large part to being benign utilities which Elastic anti-malware capabilities don't normally interfere with. Commonly used tools like Mimikatz are often seen much less, due to a layered approach to malware protection. Regarding the registry in Windows endpoints, reg save and reg export can be used to directly interact with the Registry through the Security Accounts Manager (HKLM\SAM) or the LSASS policy database (HKLM\SECURITY) hives.

Below is a table of credential access signals from Elastic Security Telemetry with additional specifics on what is being detected.



Rule Name	SUM of Alert Count
Credential Access via Known Utilities	33.37%
Security Account Manager (SAM) Registry Access	17.96%
Suspicious Access to LSA Secrets Registry	11.18%
Potential Credential Access via Mimikatz	10.72%
Potential Discovery of Windows Credential Manager Store	3.99%
Potential Credentials Phishing via OSASCRIPT	3.86%
Potential Discovery of DPAPI Master Keys	3.57%
Sensitive File Access - SSH Saved Keys	3.16%
Security Account Manager (SAM) File Access	2.92%
Web Browser Credential Access via Unusual Process	2.14%

Table 4: *Rule Name and Sum of Alerts*

Persistence

Common persistence mechanisms involving the Registry Run/RunOnce subkeys and scheduled tasks were the most frequently observed. Although this is an unsurprising trend, it represents techniques in perpetual use for a significant period of time — as valid today as they were in 2012.

These persistence mechanisms rely on basic operating system functionality with consistent benign use cases, which is one reason adversaries rely on them. This information is supported by the chart below where we see ~87% of all persistence techniques are boot or logon autostart execution.

Persistence Techniques

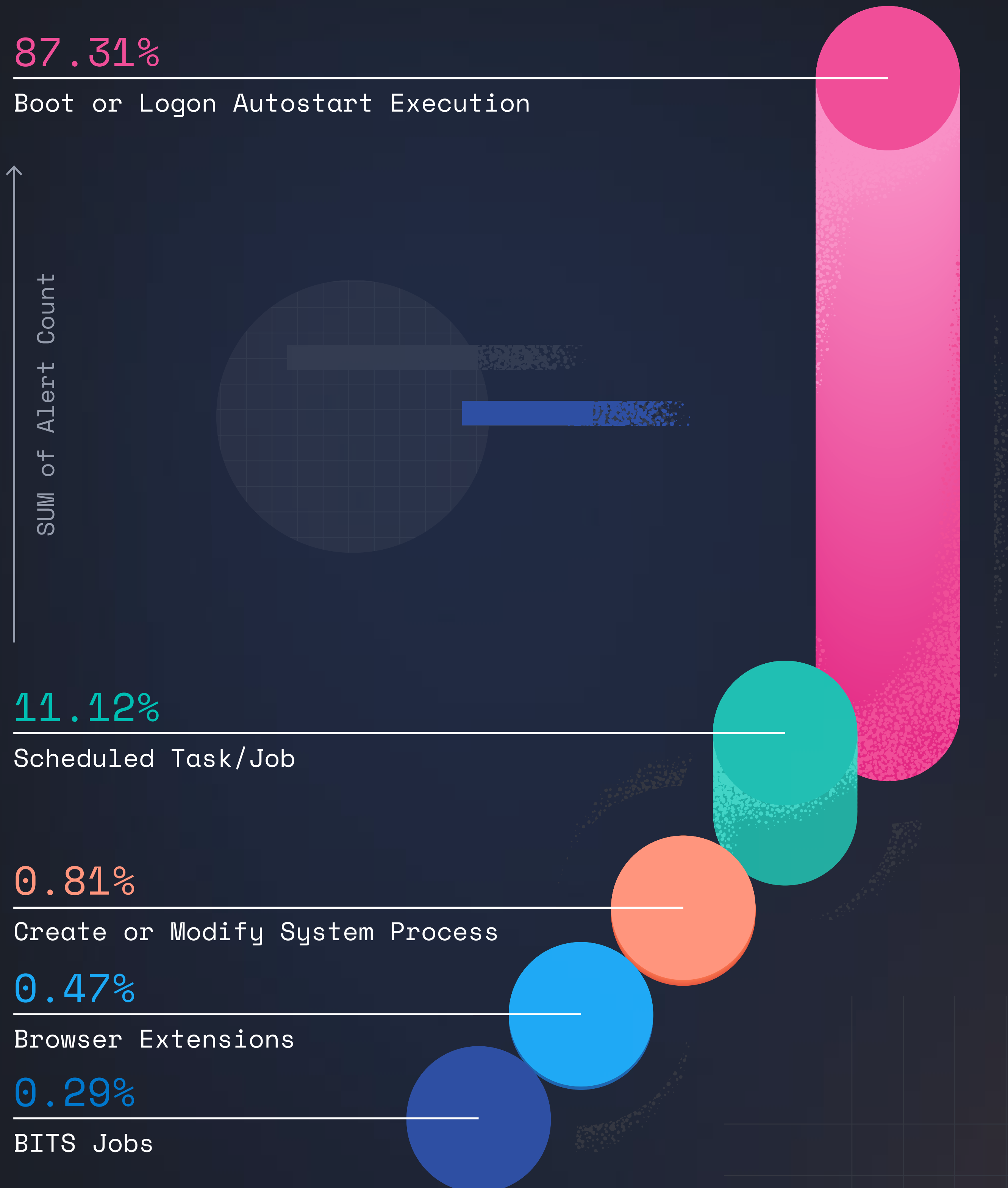


Figure 13: *Persistence Techniques*

Elastic Security Labs found that the majority of these signals related to the creation or modification of registry run keys or files in the startup folder of Windows endpoints. The outliers from the expected activity being these actions not performed by the local system account, typical processes such as PowerShell, Windows Update Agent, Windows Installer, and a few others.

A particularly common action we identified was the creation of Windows Shortcut (LNK) files by Windows' Service Host (svchost) in a user's StartUp folder, which often points directly to a native binary such as cmd.exe or powershell.exe to execute a previously installed payload or script. This is often done as a persistence mechanism to ensure the malicious code runs even if the endpoint is rebooted. Elastic Security Labs observed this same technique during our finding and analysis of BLISTER.

Due in large part to unpatched, on-premises Exchange servers, Elastic Security Labs saw a notable but statistically insignificant correlation between Exchange vulnerabilities announced earlier this year and the presence of webshells (web-based malware that threats can call on-demand).

Cloud Security Trends

Elastic Security utilizes global telemetry from the customers who leverage the pre-packaged detection rules to analyze cloud-based threats and potential attacks. This telemetry gives Elastic Security Labs tremendous insight into the potential threats customers see daily within Microsoft Azure, Amazon Web Services (AWS), and Google Cloud.

> *For the following analysis, Elastic Security Labs focused on cloud-based events from our customers and received between April and August of 2022.*

Global Findings

Elastic Security Labs analyzed telemetry from customer clusters leveraging our SIEM prebuilt detection rules for Microsoft Azure, AWS, and Google Cloud — a subset of all customers sharing telemetry. Approximately 57% of these events were attributed to AWS specifically, followed by ~22% for Google Cloud and ~21% for Microsoft Azure. Comparing this to the overall threat detection alerts, including endpoints, cloud-based detections accounted for ~5% of all global SIEM detection alerts.

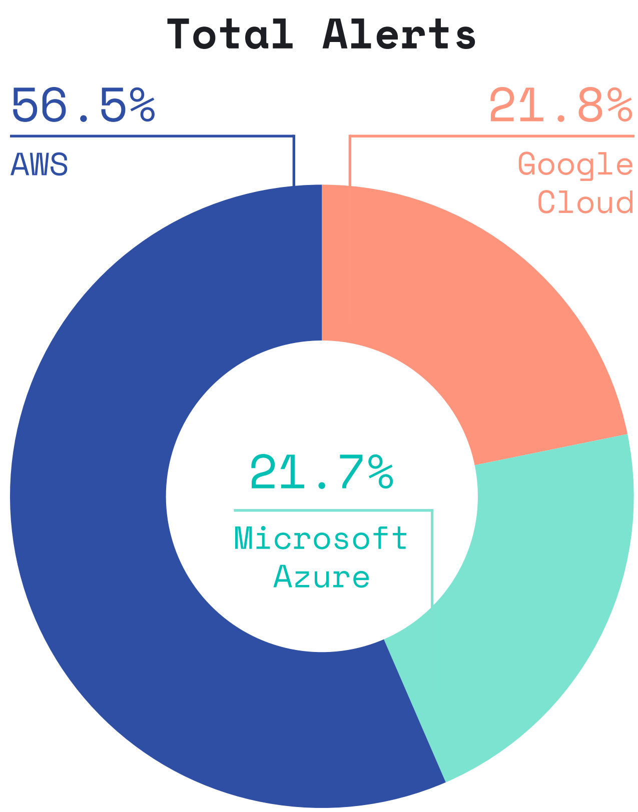


Figure 14: Percentages of Total Cloud-Based Detection Alerts by CSP

Events related to AWS detections amount to more than half of all cloud-based events we receive. According to Statista, AWS holds about 34% of the market share for cloud infrastructure service providers, though our customers may prefer AWS and this could influence our visibility. AWS and Microsoft Azure expose more than 200 services in their portfolios and more than 100 more are available for Google Cloud, which suggests that the attack surface for these CSPs is substantial.

Focusing on tactics and techniques, Elastic Security's prebuilt detection rules are mapped to MITRE's ATT&CK matrix for each CSP. Nearly 33% of all cloud alerts were related to Credential Access across all CSPs and Elastic noted that Initial Access frequently coincided.

As companies continue to embrace the services and features offered by CSPs, traditional on-premise environments are slowly transitioning to hybrid models and in some cases full cloud deployments. This increases the need to manage not only Identity and Access Management (IAM) users but service-based accounts that access these resources and applications. If left insecure (e.g., weak passwords allowed, roles excessively privileged, API keys and access tokens stored in clear-text or cached on virtual machines), risks associated with these environments, regardless of deployment, increase significantly.

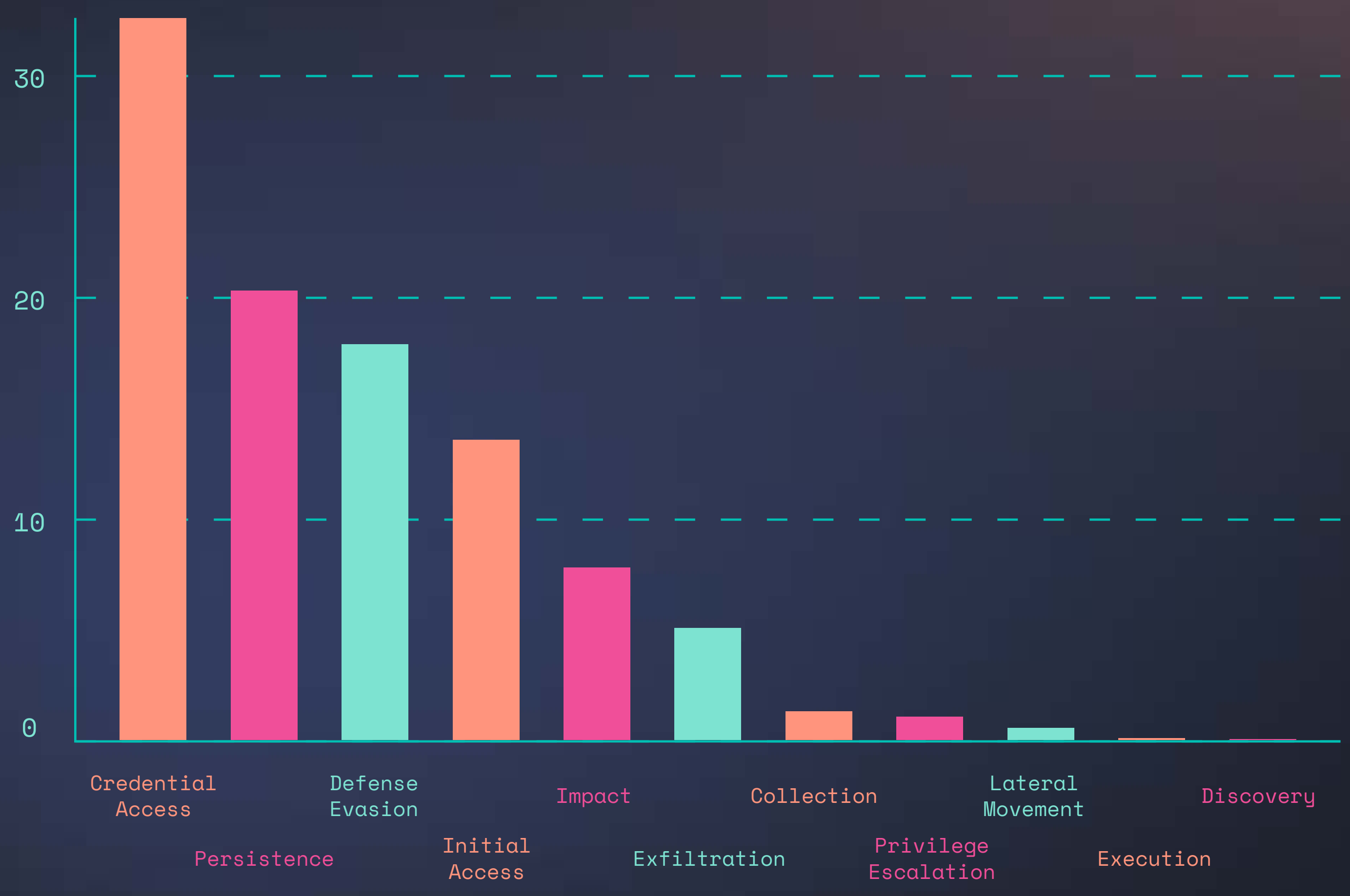


Figure 15: *Percentages of total MITRE ATT&CK tactic names for cloud-based detection alerts*

Elastic Security Labs found that about 58% of initial access attempts used a combination of traditional brute-force attempts and previously-compromised password spraying. Nearly 41% of credential access alerts were attempting to steal application access tokens (versus other credentialed materials), with approximately 1% of attempts relying on default insecure credentials. While 1% may seem like a small percentage, we observed that these attempts impacted Google Cloud and Microsoft Azure but not AWS — which may represent bias in our visibility.

The risks associated with credential theft are often great, as these services permit adversaries to move laterally from workspaces into CSP management consoles if a valid account is compromised and roles or permissions are not scoped properly.

Credential Access Techniques

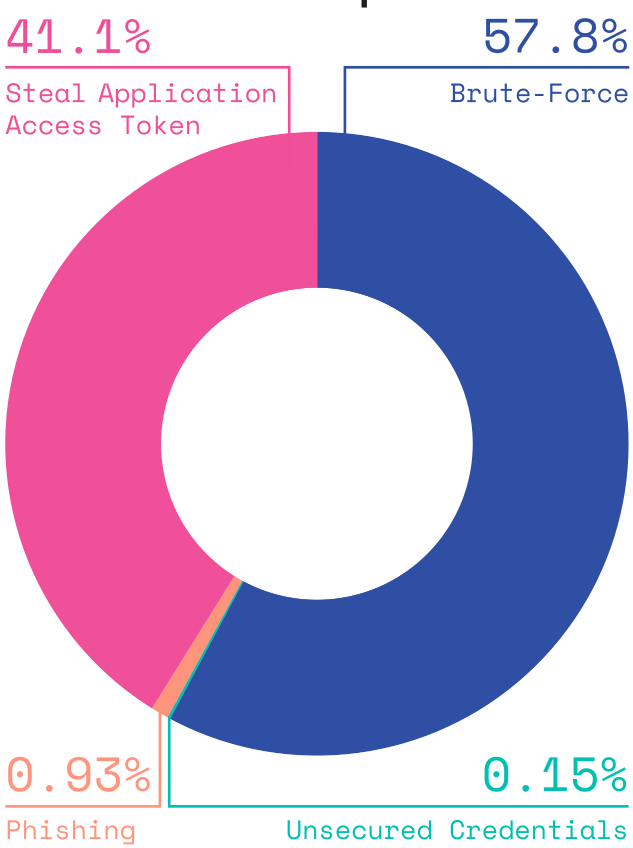


Figure 16: *Percentages of MITRE ATT&CK Techniques for Credential Access Tactic*

While CSPs compete to provide similar services to customers and thus the threat attack surface may be similar, Elastic Security Labs chose to briefly review detection alerts for each CSP to identify any specific threat trends that should be brought to our readers' attention.

AWS Findings

As stated earlier, AWS accounts for nearly 34% of the current CSP market, with a vast catalog of offerings. During analysis, Elastic Security Labs found that credential access, initial access, and persistence amounted to 74% of all detection alerts. While privilege escalation is something we might assume to be higher, it represented fewer than 2% of alerts.

AWS Techniques

Technique	Percentage
Steal Application Access Token	56.36%
Execution	14.78%
Impair Defenses	9.43%
Transfer Data to Cloud Account	7.82%
Data Destruction	6.12%
Service Stop	1.47%
Valid Accounts	1.17%
Use Alternate Authentication Material	0.85%
Abuse Elevation Control Mechanism	0.74%
Account Manipulation	0.45%
Other	0.34%
Account Access Removal	0.23%
External Remote Services	0.23%

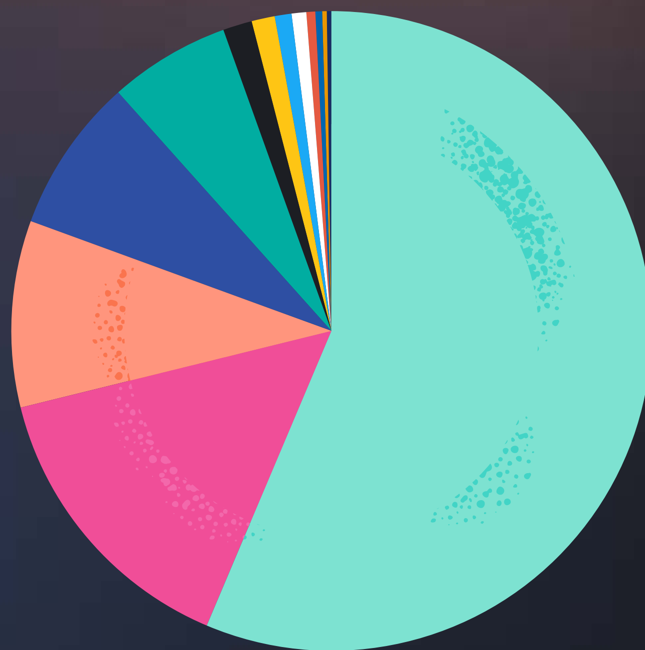


Figure 17: *Percentages Regarding ATT&CK Techniques Identified in AWS Techniques*

As shown above, ~57% of all techniques observed in AWS related to attempted application access token theft. This commonly relates to temporary credentials that are retrieved from AWS' Security Token Service (STS) via the GetSessionToken API call. With execution, Elastic Security Labs observed the abuse of the AWS CLI to leverage the Systems Manager resource in order to call prebuilt documents on a managed instance in EC2. This can be especially useful for discovery and enumeration or can be more advanced and used to target a Windows host in combination with a document created for PowerShell execution locally.

While it may seem trivial and produce benign true-positive signals, Elastic Security Labs creates detection logic for each CSP in which logging capabilities, storage, or files are deleted. Oftentimes, CSP environments are set up where logging is ingested into a single pipeline or storage resource and becomes a single point of failure if deleted, as any trace of an adversary intrusion would not exist and be troublesome for incident response (IR) engagements. Nearly 9% of AWS signals involved a combination of log stream, group, and alarm deletion.

We saw that 57% of AWS alerts came from Elastic Compute Cloud (EC2) environments. When an EC2 environment is improperly configured and permits remote access with default credentials, the most common adversary playbook is to identify IAM users, tokens, and access keys. EC2 instance metadata frequently includes semi-sensitive information. Under those conditions, it is trivial for threats of even low maturity to obtain. Roughly 39% of alerts generated by EC2 CloudTrail emphasized targeting IAM user accounts — secret keys in those environments may be at greater risk.

Microsoft Azure Findings

Microsoft Azure may carry a greater risk than AWS or Google Cloud as businesses transition away from on-premises models into hybrid or wholly cloud environments. Critical and often targeted resources such as Active Directory (AD) and SharePoint are managed in Microsoft Azure. Elastic Security found 96% of alerts observed from Microsoft Azure related to authentication events. Valid accounts were used most often in an attempt to retrieve OAUTH2 tokens, conduct phishing attacks, and other techniques.

The combined effect of several malicious techniques together was also present with Microsoft Azure, where a technique like creating an Microsoft Azure Automation Runbook later resulted in creating backdoors or a Consent Grant Attack enabled an attacker to request details about valid users. Oftentimes, Microsoft Azure service principals are the target for initial valid account compromise, as these identities are created for use with applications, automation tools, and local services where code can be executed on a VM as the local SYSTEM user.

If an adversary successfully compromised a valid account and has access to a virtual machine, the typical intrusion is to leverage Microsoft Azure's command-line interface (CLI) for additional discovery and enumeration. Using commands such as account, resource, keyvault, and network allow for an adversary to view sensitive Microsoft Azure subscription information about that specific authorized user. Related to this, the [run-command] was often observed being executed on Microsoft Azure VMs. This feature in Microsoft Azure uses an agent that is installed on Microsoft Azure Virtual Machines when they are provisioned. The risk associated is based on the pre-configured PowerShell scripts that come installed with the feature which, if abused, allow an adversary to easily modify or gain insight into the Microsoft Azure infrastructure, execute scripts on the VM, or access critical resources and applications.

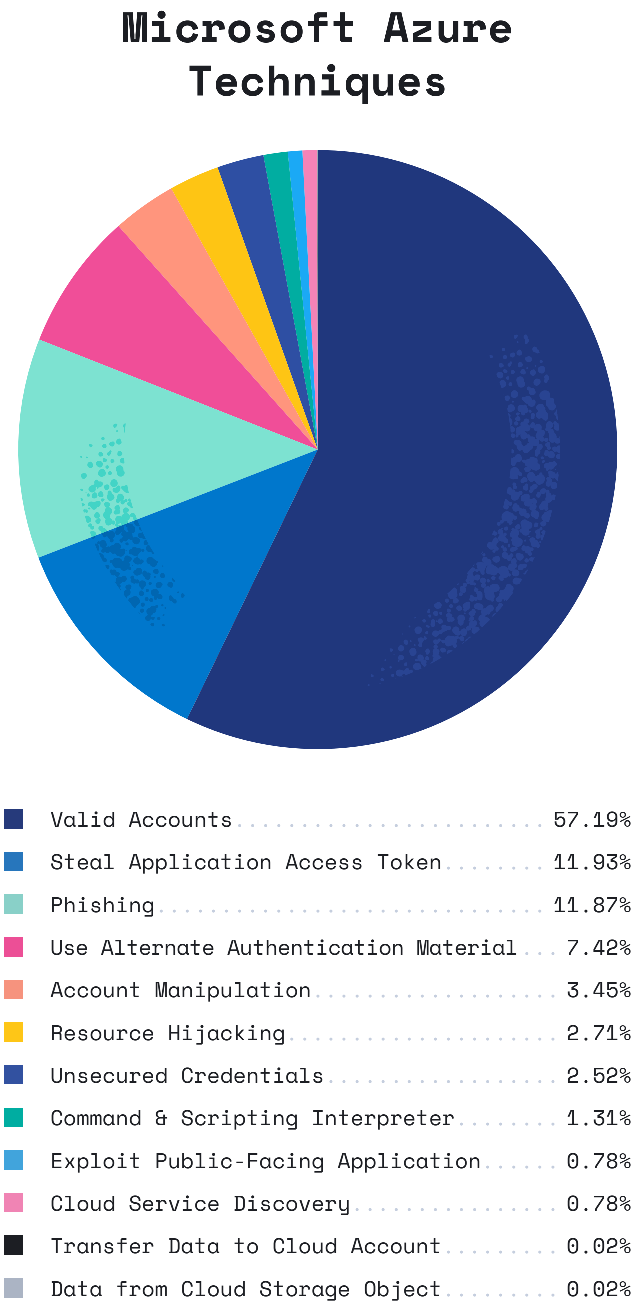


***Figure 18:** Percentages Regarding ATT&CK Techniques Identified in Microsoft Azure Environments*

Microsoft Azure PowerShell is another commonly abused native resource that adversaries can take advantage of if a valid account with non-restrictive permissions or roles is compromised. Several prebuilt modules exist to interact with not only the VM the account is on, but resources and applications such as Microsoft Azure SQL, Microsoft Azure Storage Blobs, and, of course, Microsoft Azure Active Directory.

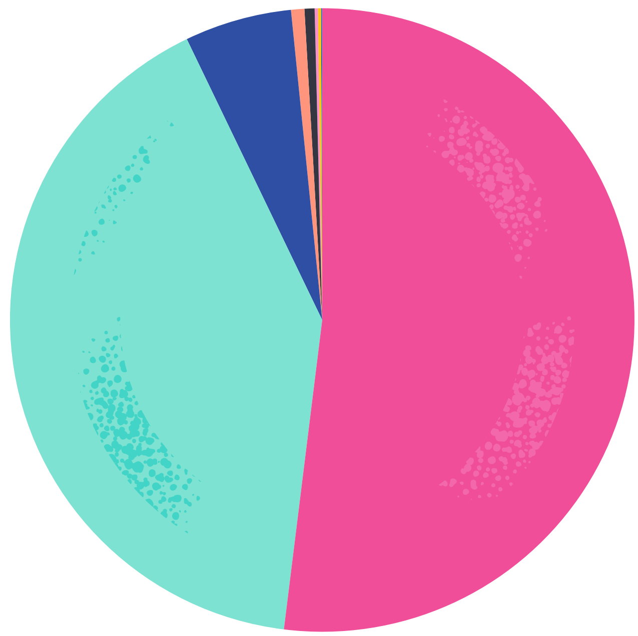
Google Cloud Findings

In the same way Microsoft Azure and Office 365 represent a combined attack surface, Google Cloud's attack surface is increased by Google Workspace where applications such as Google Drive and Gmail are vital to enterprises. While this report does not focus specifically on Google Workspace, Elastic Security Labs actively develops [publicly available] threat detection rules.

Google Cloud provides customers and developers with a straightforward architecture and setup that allows for manageable cloud bursting, chatbots for data loss prevention (DLP), mobile app backend support, virtual endpoints in Google Compute Engine (GCE), and much more.

Regardless of the reasons that customers rely on Google Cloud, service account compromise remains rampant when default account credentials aren't changed. About 54% of alerts from Google Cloud environments were related to service account abuses, though it is not clear how many of those relied on default credentials. While service account key generation is a valid persistence method, it's often unnecessary if the "iam.serviceAccountTokenCreator" is simply applied to a valid account the adversary has already compromised.

Google Cloud Techniques



Account Manipulation	51.96%
Impair Defenses	40.90%
Data from Cloud Storage Object	5.54%
Create Account	0.69%
File & Directory Permissions Modification	0.52%
Account Access Removal	0.17%
Data Destruction	0.16%
Valid Accounts	0.03%
Transfer Data to Cloud Account	0.03%

Figure 19: *Percentages Regarding ATT&CK Techniques Identified in Google Cloud Techniques*

Threat Profiles

This section summarizes threat profiles developed by Elastic over the past year, and for which we have unique telemetry insights. Elastic leverages our telemetry data to discover and track threats.

Four major activity groups are represented:

- **BLISTER** [REF7890]
- **PHOREAL** [REF4322]
- **CUBA** [REF9019]
- **QBOT** [REF3726]

For each group listed, we'll provide a conventional diagram referred to as the Diamond Model, which describes the relationships between adversaries, infrastructure, capabilities, and victims. To improve readability, we have pared down overlaps with groups tracked by other vendors, but readers should note that this doesn't indicate agreement or disagreement with those vendors.

The Diamond Model

We utilize the Diamond Model to describe high-level relationships between the adversaries, capabilities, infrastructure, and victims of intrusions. This model is often used in an intrusion-centric way, but here we employ it with an adversary focus to demonstrate observations over many incidents.

Terminology

Activity Group: Individuals, groups, or organizations believed to be operating with malicious intent. We prefix these activity groups with the string REF and a sequence of numbers to distinguish our visibility from the visibility of others.

Attack Pattern: Describes ways that adversaries attempt to compromise targets.

Intrusion Set: Adversarial behaviors and resources with common properties that are believed to be orchestrated by a single organization.

Blister [REF7890]

On December 22, 2021, Elastic discovered a novel form of malware loader we dubbed BLISTER. Upon execution, the BLISTER loader decrypted a backdoor from within itself and executed it in memory — impacting a single customer environment. This activity group overlaps with a few third-party labels.

Figure 20 depicts the code-signing metadata present in that initial loader, which indicates that the binary had been prepared nearly two months prior to its discovery by Elastic. The digital signature listed “Blist LLC” as the signing entity, hence the name of this loader family. Readers should be aware that a “BLISTER’d" payload could be signed by any one of several low-cost certificate authorities—none of which would be legitimately expected to sign the backdoored binary.

“BLISTER-ing” a benign application has the effect of ensuring that most of the binary’s code is also benign. Machine-learning capabilities that do not incorporate benign software in training data can be fooled by this approach, and the use of valid code-signing signatures often fools human analysts. This offense-in-depth methodology can be effective at bypassing a wide range of enterprise mitigations.

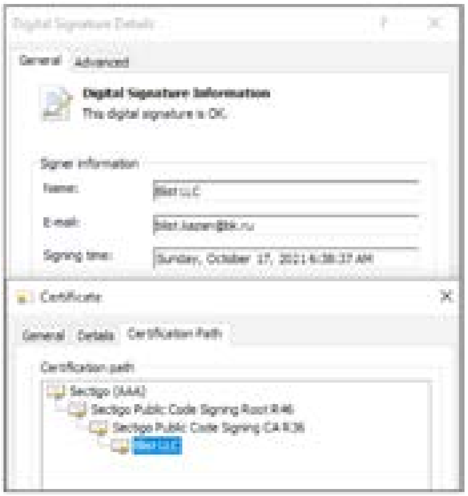


Figure 20: BLISTER loader code-signing metadata

What is the Threat?

BLISTER is a stealthy malware loader used to execute payloads via various techniques. Because nearly any benign application can be modified into a BLISTER-like loader, one characteristic of this methodology is that BLISTER’d loaders inherit code-signing metadata which often bypass brittle security controls.

What is the Impact?

REF7890 uses BLISTER to load implants including the offensive security tool (OST) Cobalt Strike and the BitRat backdoor. These provide a wide range of remote access capabilities to an infected host. In all Elastic observations, BLISTER was used to facilitate data theft and extortion.

What was Elastic’s Response?

Elastic provides out-of-the-box detections and preventions for the BLISTER loader. Additionally, Elastic publicly released YARA rules, hunting queries, a detailed campaign and malware analysis, and a configuration extractor.

Learn More

- BLISTER Loader
- BLISTER Malware Campaign
- BLISTER Configuration Extractor

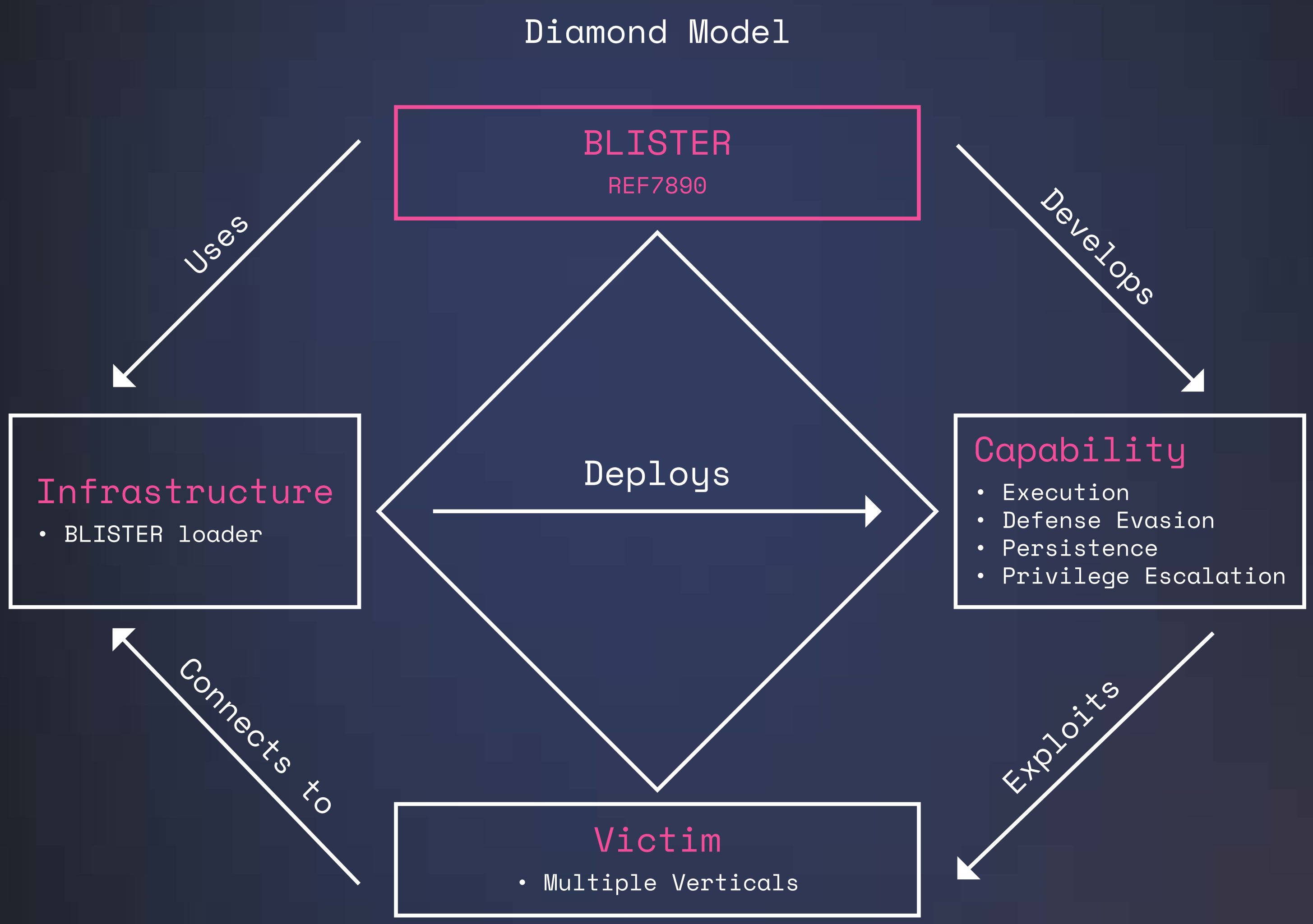


***Figure 21:** BLISTER Diamond Model*

Phoreal [REF4322]

Earlier this year, Elastic identified the presence of the PHOREAL backdoor (often referred to as the RIZZO backdoor) affecting Vietnamese financial institutions. What made this infection stand out was the use of in-memory evasions which were previously unseen in the implant. These particular alerts were interesting because they all occurred within the same cluster, and unusually they targeted the control.exe process. The Windows control.exe process handles the execution of Control Panel items, which are utilities that allow users to view and adjust computer settings. Elastic notes overlap with the APT32 and OCEANLOTUS third-party labels.

What is the Threat?

PHOREAL is a full-featured backdoor allowing initial access and post-exploitation operations to obtain victim data. PHOREAL operators have been active since at least 2014, though methodology and capabilities have evolved over time.

What is the Impact?

REF4322 largely targets victims with both private and public sectors in Southeast Asia, specifically Vietnam. Elastic assesses with moderate confidence that this threat pursues state objectives, and victimology suggests economic, political, and industrial espionage as objectives.

What was Elastic's Response?

The Elastic Security team detailed how to triage one of these threat alerts, extracted observables for endpoint and network filtering, and produced a new malware signature for identification and mitigation of the threat across the fleet of deployed Elastic Agents.

Learn More

- PHOREAL Malware Targets the Southeast Financial Sector

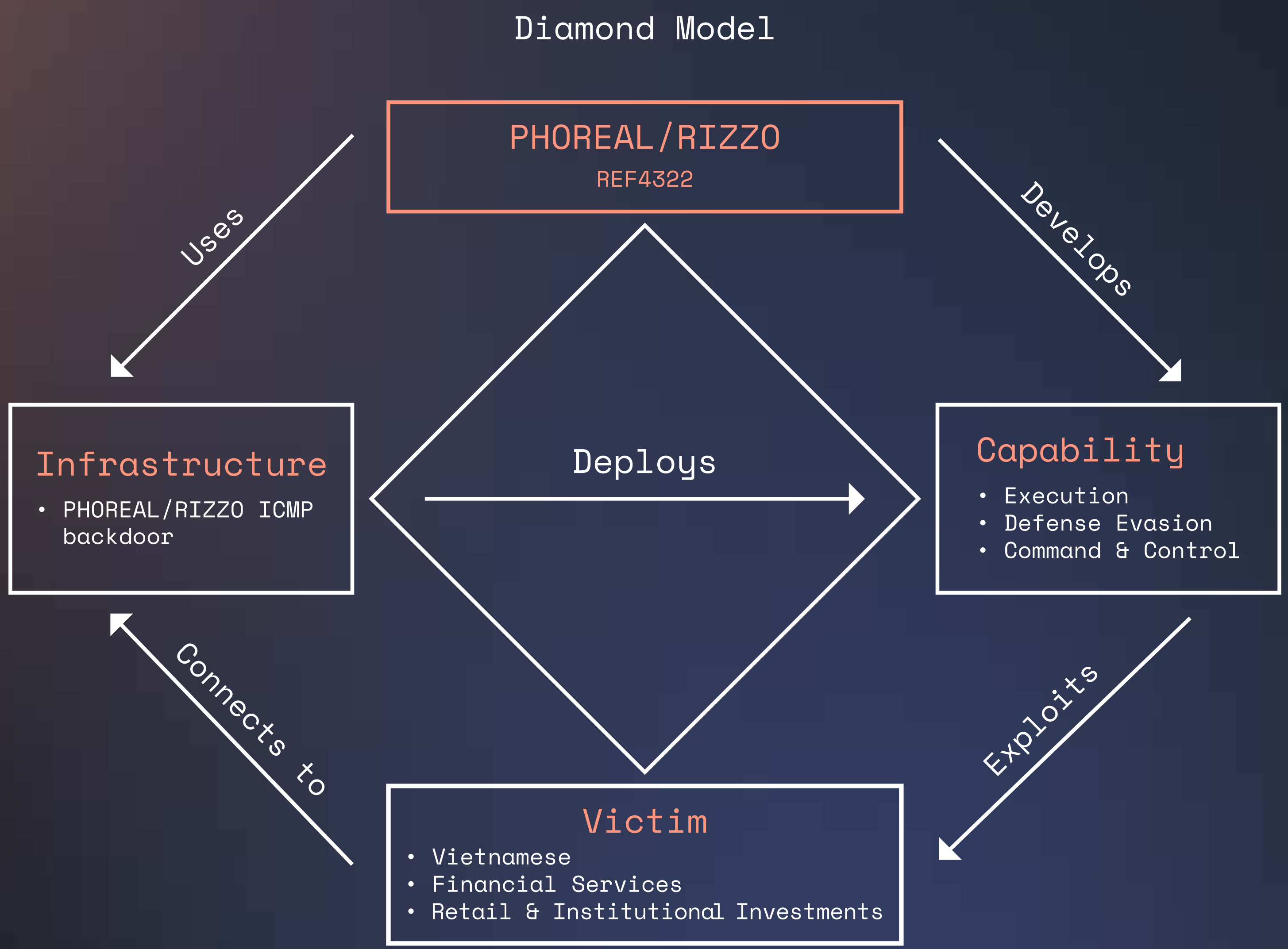


***Figure 22:** PHOREAL Diamond Model*

CUBA [REF9019]

In June of this year, Elastic observed multiple intrusion attempts related to the CUBA ransomware group, based on the common use of the CUBA ransomware payload and shared infrastructure. While victims in a variety of industries were impacted, Elastic was unable to identify a common method of initial access. Some organizations were compromised via unpatched vulnerabilities, others appear to have been compromised via unrelated malware in their environments that had not been effectively remediated. Elastic notes overlap with the UNC2596 third-party label.

Activity patterns associated with REF9019 involved a variety of malware implants such as COBALTSTRIKE, METASPLOIT, BUGHATCH, and SYSTEMBC, as well as valid remote support utilities like GoToAssist.

A summary of techniques observed in a selected, single victim environment:

- Exploit Public-Facing Application
- Command and Scripting Interpreter - PowerShell, Windows Command Shell
- Scheduled Task/Job - Scheduled Task
- Boot or Logon Autostart Execution - Registry Run Keys/Startup Folder
- Create Account - Local Account
- OS Credential Dumping - LSA Secrets
- Data Encrypted for Impact
- Hide Artifact - Hidden Window
- Masquerading - Match Legitimate Name or Location
- Obfuscated Files or Information
- Reflective Code Loading

What is the Threat?

This activity group leverages the CUBA ransomware and diverse remote access capabilities to target North American and European retailers and manufacturers. The threat group has followed an effective but repetitive set of Tactics, Techniques, and Procedures (TTP)s for initial access, lateral movement, exfiltration, ransomware deployment, and extortion.

What is the Impact?

REF9019 targets North American and European retailers and manufacturers, engaging in extortion after encrypting and stealing sensitive files.

What was Elastic's Response?

Elastic has publicly released YARA signatures, hunting queries, and endpoint protections to detect this ransomware family.

Learn More

- CUBA Ransomware Campaign Analysis
- CUBA Ransomware Malware Analysis

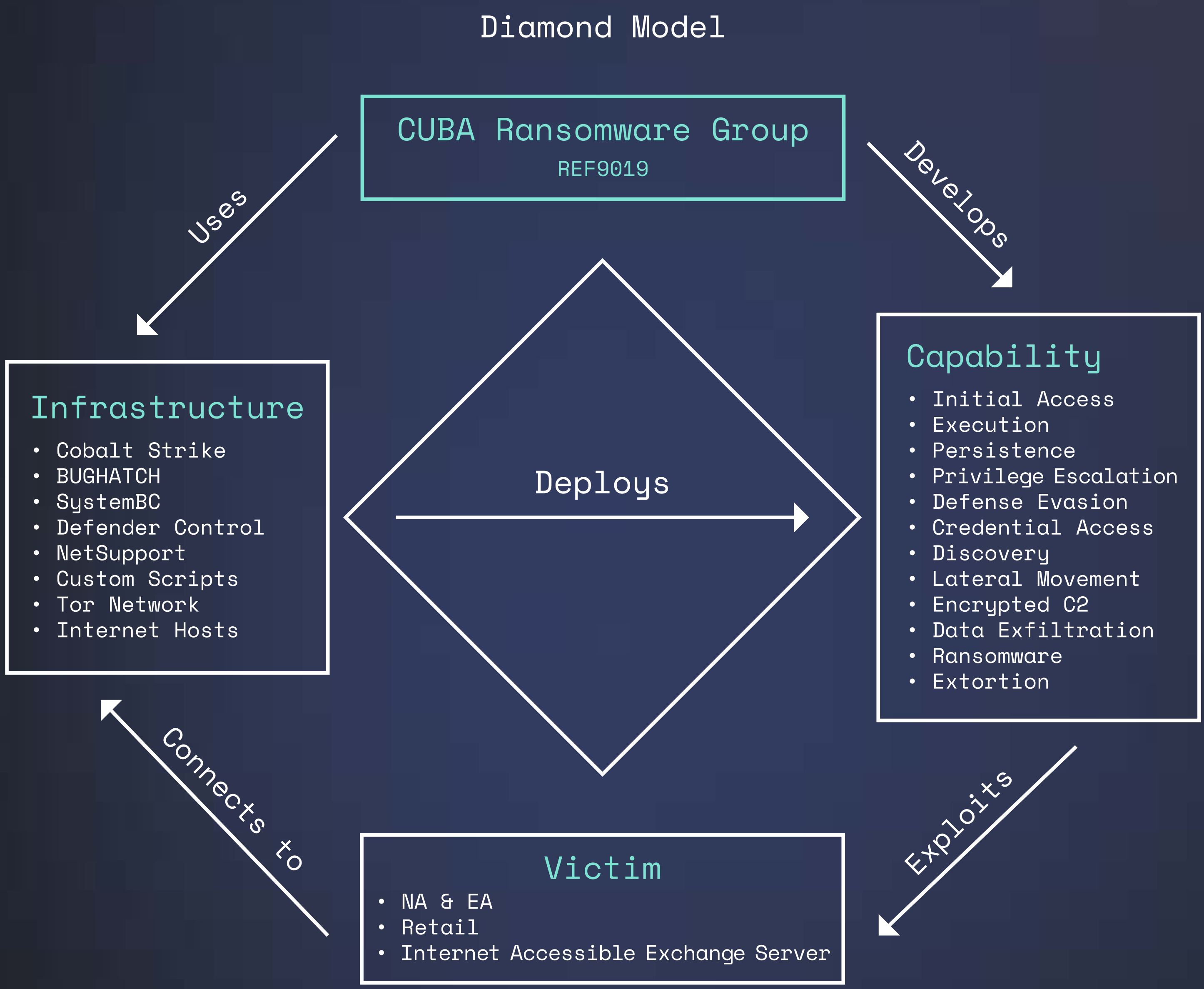


Figure 23: *CUBA Diamond Model*

QBOT [REF3726]

QBOT (also known as QAKBOT) is currently one of the most prolific malware families in use around the world and has been developed since approximately 2007. Used to facilitate ransomware-related crimes, this malware is notable for its use of multistage execution and employing an exception list to avoid infecting systems in eastern European states. Elastic tracks this activity pattern as REF3726 and notes overlaps with a significant number of vendor labels as a widely available capability for purchase.

Modern incarnations of this malware rely on native capabilities, such as the regsrv32.exe execution proxy, to gain initial access. This can be achieved through diverse initial access mechanisms such as exploitation of client software, weaponized document lures, backdoored legitimate software, and software supply-chain compromise. Figure 24 depicts a common initial execution method.

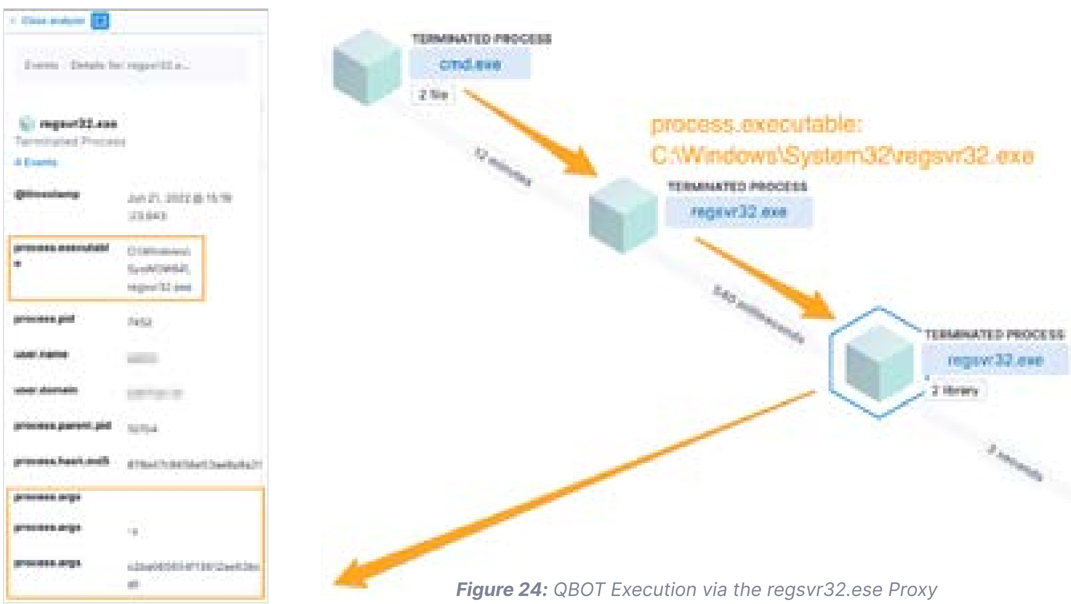


Figure 24: *QBOT Execution via the regsvr32.ese Proxy*

What is the Threat?

QBOT is a prolific modular trojan that has been active since approximately 2007 as a mechanism supporting financially motivated threats.

What is the Impact?

REF9019 targets North American and European retailers and manufacturers, engaging in extortion after encrypting and stealing sensitive files.

What was Elastic's Response?

Elastic publicly released endpoint protections, prebuilt Detection Engine logic, YARA rules, and a configuration extractor. Additionally, Elastic used the QBOT research to associate 138 adversary-owned or controlled IP addresses with 339 malicious files.

Learn More

- Exploring the QBOT Attack Pattern
- QBOT Malware Analysis
- QBOT Configuration Extractor

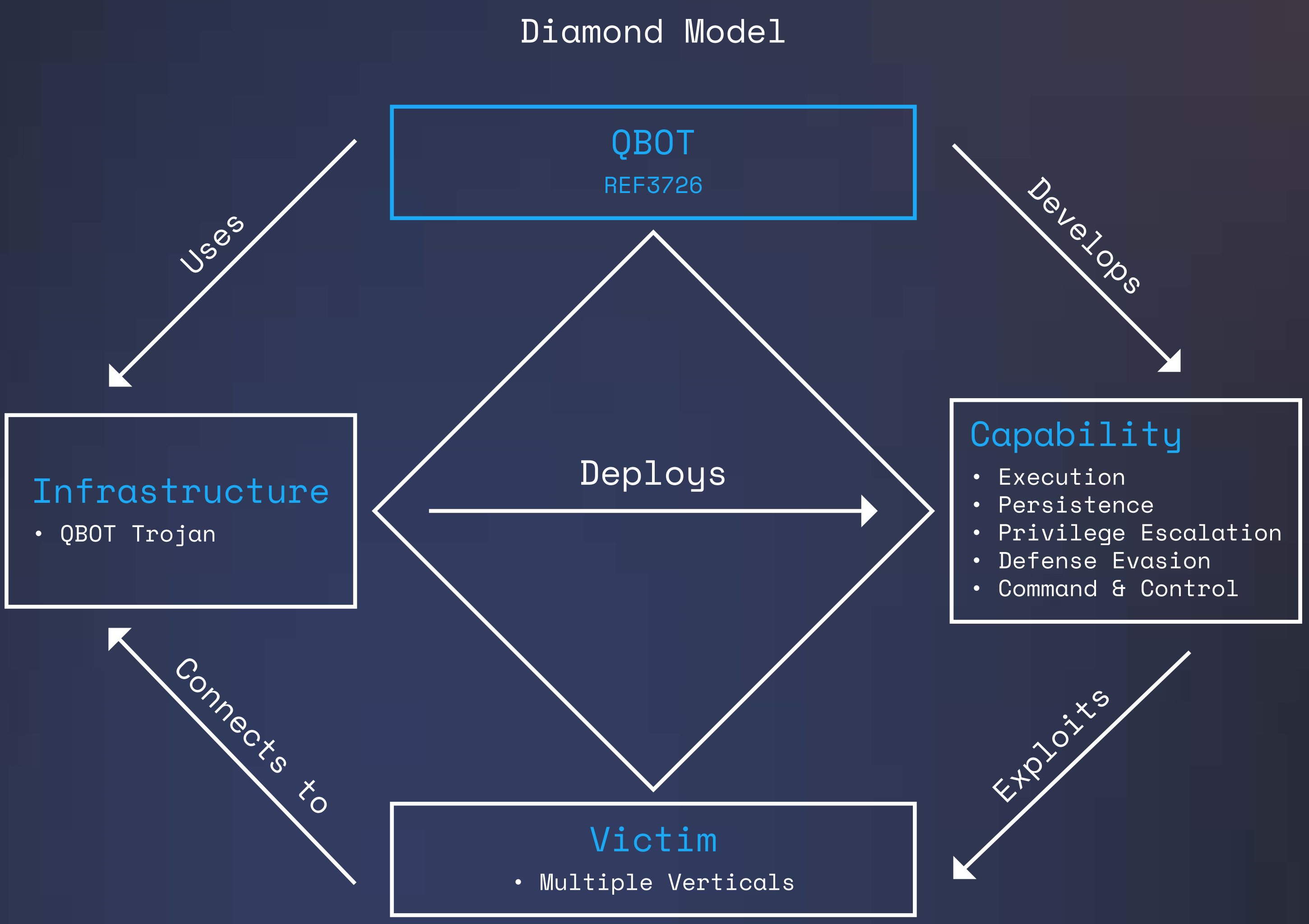


Figure 25: QBOT Diamond Model

Forecasts & Recommendations

Based on trends, correlations, and ongoing research into the evolving global threat landscape, Elastic offers the following forecasts and recommendations. Rather than a prediction, readers should note that forecasts indicate possible outcomes that are influenced by too many factors to precisely anticipate. Readers should also consider recommendations to be suggestions and not guarantees from cyber threats.

Forecast 1:

Adversaries will continue to abuse built-in binary proxies to evade security instrumentation.

Threats of all kinds continue to leverage native tools like Rundll32.exe as a binary proxy to load malicious software, closely mimicking the intended purpose of those tools. They remain popular because they work, and are expected to be effective long-term.

Recommendation 1

Enterprises should closely monitor the use of system binary proxy software, and establish knowledge of benign and malicious patterns of activity. These tools are ubiquitous and play an important role during initial access. Constraining their use has an immediate effect on preventing compromise.

Forecast 2:

LNK and ISO payloads will replace more conventional script and document payloads.

Due to decisions by Microsoft to alter Windows system security this year, LNK and ISO payload types have become necessarily viable for threats of all kinds. Few organizations scrutinize these object types, resulting in more frequent compromise.

Recommendation 2

With few exceptions, organizations should treat LNK and ISO attachments as risky. The use of these objects can be controlled by policy, and detection engineers can provide strategies for detecting their use.

Forecast 3:

Valid IAM accounts will continue to be a target for adversaries.

As a gateway to other intrusion objectives, the theft of credentials is an essential step for many adversaries during the early stages of an attack. During this year, Elastic observed several threat groups stealing valid credentials to authenticate cloud resources and bypass the need for exploitation.

Recommendation 3

It's essential to understand normal patterns of activity for different types of accounts as well as individuals to identify when those accounts are being abused. Cloud resource visibility at the CSP, orchestrator, and worker levels may be necessary to identify malicious patterns of activity affecting cloud-hosted applications or services being accessed using valid accounts. Access to metadata APIs or credentials used in DevOps should be likewise monitored for unusual resource access outside of normal account behavior.

Forecast 4:

Service accounts managed by each major CSP (Google, Amazon, Microsoft), which are not configured with least-privilege permissions or are misconfigured, will be the target of adversaries.

Service account credentials are often a stepping stone from initial access to persistent access, and can be exposed accidentally. Commits to public GitHub repositories were just one way we saw threat actors obtain these kinds of credentials, and increasingly this approach will be viable as enterprises seek to automate more of their infrastructure. Additionally, enterprises should be aware that some threats disable auditing to evade detection.

Recommendation 4

Understand and implement secure access to cloud resources, and plan for adversary interference. Organizations with more than one source of visibility (CSP auditing, orchestrator logs, endpoint sensor instrumentation) demonstrated a greater ability to detect and mitigate attacks against cloud-hosted platforms.

Forecast 5:

Linux virtual machines used for backend DevOps, but deployed in cloud environments, may see an increase in targeting — with an emphasis on threats to account enumeration and credential access.

While developers continue to represent a brittle attack surface for organizations, enterprises that do not understand those risks will be compromised. Understanding the scope of content deployed to virtual machines, alongside the access granted inherently by these systems, is critical to determining how much of a risk is posed to the organization

Recommendation 5

Understand that adversaries can easily and quickly pivot from an exposed key in GitHub to CSP access, and from there possess the capability to create, modify, or destroy resources — including installing malware, stealing data, or implementing a deliberate misconfiguration.

Forecast 6:

Organizations over-investing in detection capabilities that do not also support mitigation will struggle with security response against all categories of threats.

Simply put, enterprises cannot effectively and quickly respond to threats using detect-only instrumentation. Those unable to deploy capabilities to mitigate threats centrally are at a profound disadvantage against fast-moving threats — both targeted and otherwise.

Recommendation 6

Assess your ability to rebuild infected systems, reset compromised account credentials, sinkhole a DNS entry, block all traffic to/from an IP address, isolate a host, and restore business-critical data from backups. Prioritize instrumentation that supports those outcomes and select tooling that allows for automated mitigation strategies where possible.

Conclusions

The global threat landscape is perpetually evolving, with new threats and capabilities filling the niche of those that preceded them. Phishing lures drop malicious macros in favor of ISO or LNK objects, reflecting the ways technology shapes those capabilities. Yet phishing attacks continue to be the most common method of initial access. This is revealing, as it highlights a kind of risk continuity that enterprises still haven't quite mastered.

It's also informative, illustrating that some aspects of the threat landscape aren't easily addressed through technology. Instead, success depends on the right visibility, capabilities, and expertise. In nearly all cases, we found that those three factors cooperated to achieve success or conspired to result in failure.

We observed that visibility played a big role in how enterprises understood their attack surface. Until relatively recently, many organizations didn't consider cloud-hosted applications or systems to be part of their enterprise. Due perhaps to the poor default visibility afforded by those solutions, few organizations had reason to rely on them as part of their security instrumentation.

Elastic primarily uses telemetry to improve feature efficacy and to provide organizations with additional security context through publications such as this. We welcome the opportunity to partner with our customers in this way to analyze their data, anonymously sharing what we learn with the larger security industry.

Our understanding of the global threat landscape is bound to change along with the landscape itself. Investments in data collection and our sensory apparatus indicate that visibility is the first step towards comprehension, and comprehension empowers us to act. For those who lack visibility, it may not be practical or possible to achieve action, which is increasingly relevant due to the rapid tempo that many threat groups have evolved to use.

Organizations without expertise faced perhaps the greatest obstacles — relying on vendors and service providers to set up, manage, and operate their security infrastructure. This dependency, which is motivated by a variety of reasons, often left targeted entities at a disadvantage, whether the threat was a newly announced vulnerability, a threat group determined to extort, or collateral from a geopolitical event.

This year, Elastic stepped out from behind the curtain and emerged as a security company. With a long history of openness, the community gained access to our research with Elastic Security Labs and much of our security technology through our Protections Artifacts resource. These seemingly unrelated endeavors were, in fact, part of our approach to the ever-changing nature of the threat landscape. Visibility, capability, and expertise are how we intend to help you create hostile environments to threats. If we can find them once, in one place, we can interfere with them everywhere, at once.

We will be here when you need us. We'll be here when you're ready to join us, too.

Learn about Elastic Security and protect against the threats covered in this report (and other vulnerabilities) by visiting our Elastic Security Labs page. You can also follow us on Twitter to see when we release news-breaking threat research.

Find more research:

www.elastic.co/security-labs

Follow us on Twitter:

@elastic

elastic security labs