Google Cloud

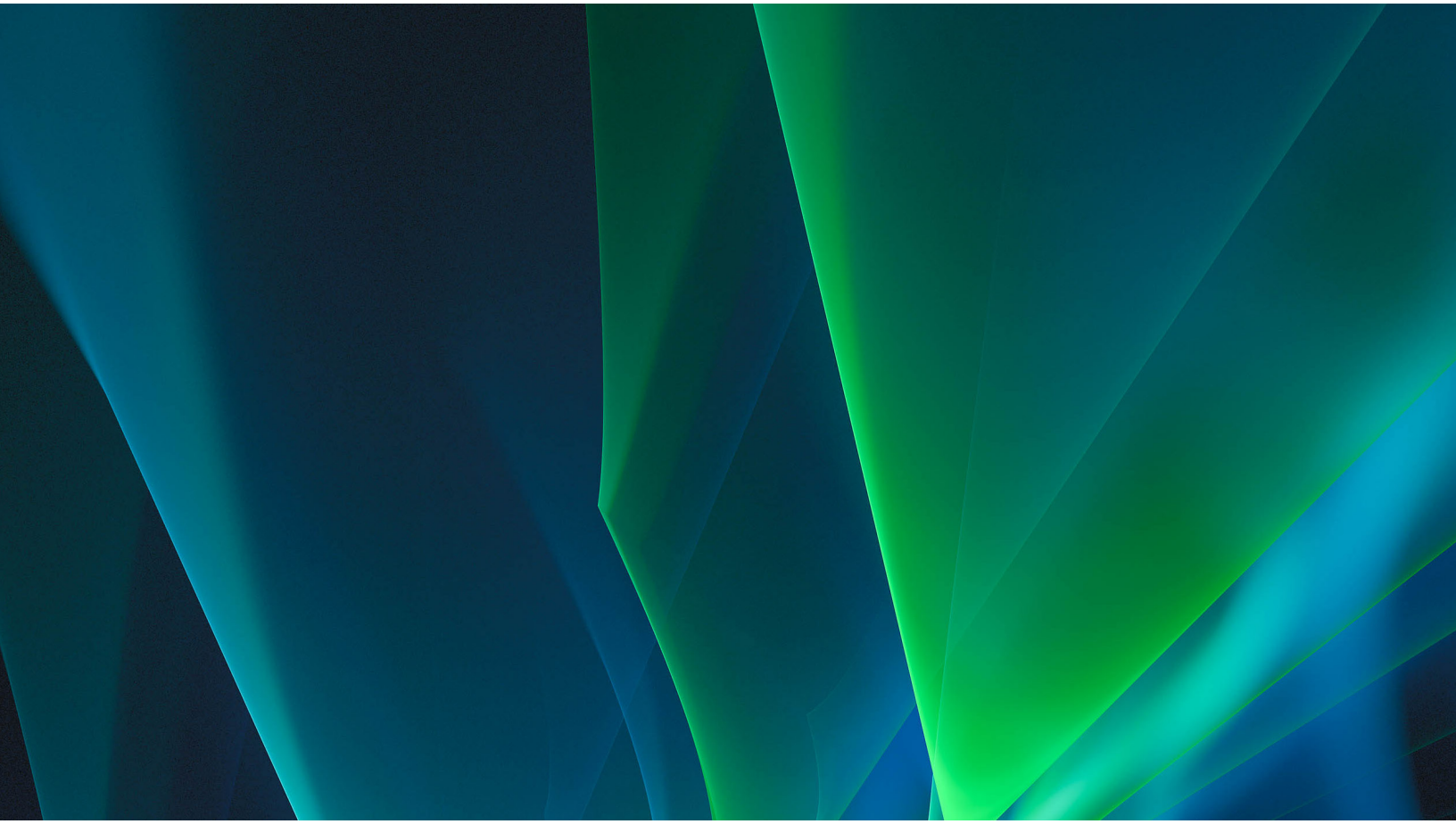
Security

# Cloud Threat Horizons Report

H2 2025

Office of the CISO

## Table of Contents



# Mission Statement

The Google Cloud Threat Horizons Report provides decision-makers with strategic intelligence on threats to not just Google Cloud, but all cloud service providers. The report focuses on recommendations for mitigating risks and improving cloud security for leaders and practitioners. The report is informed by Google Cloud's Office of the CISO, Google Threat Intelligence Group (GTIG), Mandiant Consulting, and various Google Cloud intelligence, security, and product teams.

## Executive Summary

# Heightened Cloud Threats: Actors Refine Tactics for Evasion, Persistence, and Supply Chain Compromise

Cloud environments face an increasingly sophisticated threat landscape as actors advance their methods for data exfiltration, identity compromise, and supply chain attacks, while simultaneously improving evasion and persistence techniques. This Google Cloud Threat Horizons Report offers cloud security professionals critical insights into these evolving threats, supported by intelligence and actionable risk mitigations from Google's security experts.

The dangers of prolonged access, complex recovery scenarios, and supply chain vulnerabilities are not new. Throughout 2025, Google Cloud security and intelligence teams have continued to track these risks. The findings in this report underscore a significant evolution: threat actors are not only honing their tactics for greater impact within cloud environments but are also demonstrating increased sophistication. This sophistication is evident in their targeting of recovery mechanisms, developer ecosystems, and their methods for achieving high-value compromises.

Recognizing our shared responsibility in defending against these advanced cloud threats, this report delivers timely analysis and actionable mitigations, drawing on the following key trends identified by our security and threat intelligence experts:

- **Foundational security remains the strongest defense:** Google Cloud research indicates that credential compromise and misconfiguration remain the primary entry points for threat actors into cloud environments, emphasizing the critical need for robust identity and access management and proactive vulnerability management.
- **Targeting of backup infrastructure:** Financially motivated threat groups are increasingly targeting backup systems as part of their primary objective, challenging traditional disaster recovery, and underscoring the need for resilient solutions like Cloud Isolated Recovery Environments (CIRE) to ensure business continuity.
- **Sophisticated social engineering and multi-factor authentication (MFA) bypass:** Advanced threat actors are leveraging social engineering to steal credentials and session cookies, bypassing MFA to compromise cloud environments for financial theft, often targeting high-value assets.

- **Misuse of trusted cloud services for decoy file delivery to facilitate malware infections:** In a recent campaign, threat actors used .desktop files to infect systems by downloading decoy PDFs from legitimate cloud storage services from multiple providers, a tactic that deceives victims while additional malicious payloads are downloaded in the background.
- **Browser extension supply chain risk:** To combat threat actors using compromised OAuth tokens to bypass MFA and inject malicious code via automated CI/CD pipelines, Google has introduced Verified CRX Upload controls to secure the non-human identities used in these cloud-based build processes.

To effectively navigate the evolving threat landscape in H2 2025 and beyond, organizations must prioritize a defense-in-depth strategy focusing on identity security, robust recovery mechanisms, continuous vigilance against sophisticated social engineering and deception tactics, and supply chain integrity. The following content provides cloud security decision-makers with the latest intelligence on threat actor tactics and actionable mitigations to better inform cloud security strategies.

# Foundational Security Remains Critical Against Persistent Threats

Google Cloud's latest research highlights that common hygiene gaps like credential issues and misconfigurations are persistently exploited by threat actors to gain entry into cloud environments. During the first half of 2025, weak or absent credentials were the predominant threat, accounting for 47.1% of incidents (Fig. 1). Misconfigurations (29.4%) and API/UI compromises (11.8%) followed as the next most frequently observed initial access vectors. These findings largely mirror our observations in the H1 2025 Cloud Threat Horizons Report.

Notably, compared to H2 2024, we observed a 4.9% decrease in misconfiguration-based access and a 5.3% decrease in API/UI compromises (i.e., when an unauthorized entity gains access to, or manipulates a system or data through an application's user-facing screen or its programmatic connections). This shift appears to be partly absorbed by the rise of leaked credentials representing 2.9% of initial access in H1 2025. This highlights an urgent, evolving risk: the exploitation of credentials discovered on dark web sources, underscoring the critical need for rapid detection and remediation strategies. Google Cloud has integrations with partners to identify and notify customers of leaked credentials along with configurations to automatically disable leaked keys before threat actors can exploit them.

Another vector—remote code execution (RCE)—accounted for 2.9% of initial access in H1 2025. While this figure remains consistent with previous Cloud Threat Horizons Reports, its persistence underscores the critical need for effective, timely patch management. Recognizing this ongoing threat posed by vulnerabilities that can lead to RCE, the Google Cloud CISO Security Engineering (CCSE) Cloud Vulnerabilities Research (CVR) team proactively discovered critical Rsync vulnerabilities in Q4 2024. These flaws, if exploited by threat actors, could enable RCE leading to significant supply chain compromises. Our subsequent coordinated disclosure with the U.S. Department of Homeland Security Cybersecurity and Infrastructure Agency and industry partners in Q1 2025 demonstrates Google Cloud's dedication to improving global cloud security by addressing these pervasive threats.

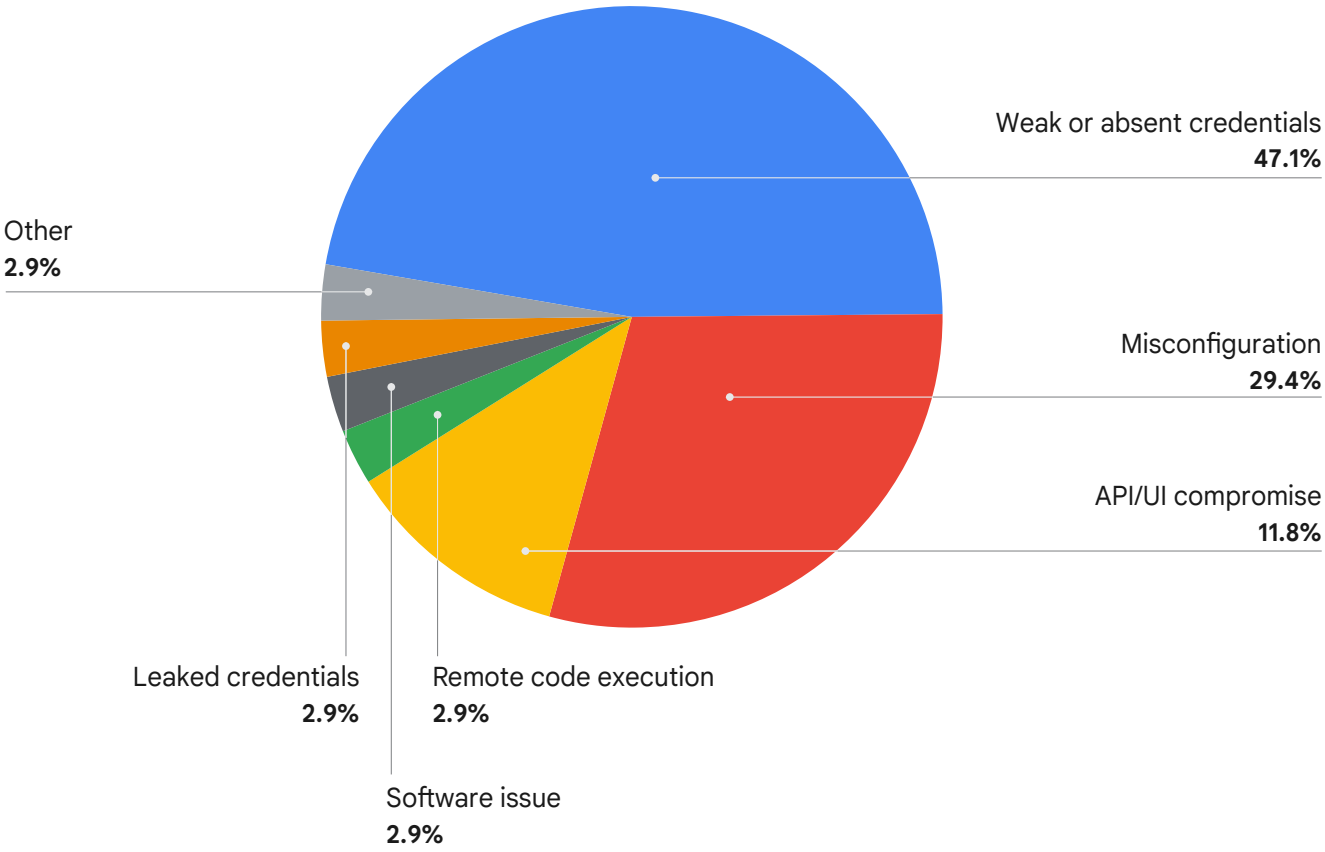



Figure 1. H1 2025 Distribution of Initial Access Vectors Exploited by Threat Actors. Data for this is provided by a larger set of observed data and as such may not in all cases be representative of all Google Cloud customers.

## Mitigations

To address the initial access vectors highlighted in our research, we recommend a defense-in-depth strategy focusing on strong identity controls, proactive threat and vulnerability management, and comprehensive security posture oversight.

### Identity and Access Management (IAM) Controls:

- Regularly audit permissions granted to both human users and service accounts to identify and remove excessive access to meet the Principle of Least Privilege. Use Google IAM Recommender to automatically identify and help remove excessive permissions to enforce the Principle of Least Privilege. Regularly review IAM policies to ensure

services and users have only the minimum access required for their roles, limiting the potential damage of a compromised credential or API key.

- Protect applications from credential stuffing by moving beyond network-level controls. Secure applications by implementing Identity-Aware Proxy (IAP) to enhance protection with zero trust. IAP enforces identity-based authentication and authorization, creating a central control point that shields your applications and reduces the attack surface that could otherwise be targeted by stolen credentials or attempts to exploit vulnerabilities.
- Proactively monitor for leaked credentials. Leverage Google Cloud's integrations that actively monitor for leaked credentials on public sources, notify you, and provide configurations to automatically disable the keys before they can be used for malicious access.

### Visibility and Proactive Defense Controls:

- Maintain a unified view of your cloud environment to monitor for misconfigurations, vulnerabilities, and active threats. Use Google Security Command Center (SCC) as a central platform to continuously monitor for misconfigurations, vulnerabilities, and threats across your Google Cloud environment.
- Maintain a robust vulnerability and patch management program. Use SCC's vulnerability detection or scan container images in Artifact Registry to identify vulnerabilities in your deployed applications and operating systems. A robust and timely patch management process is the most effective defense against the persistent threat of RCE exploits, which often target unpatched software.

# Cloud-Native Backup and Recovery for Modern Cyber Threats

Destructive cyberattacks, such as ransomware, often extend beyond technical disruptions, and can result in significant business downtime and financial losses due to interrupted operations. Financially motivated cyber criminals are targeting not only production systems and data, but also backup infrastructure and platforms, as highlighted in the M-Trends 2025 Report.

Our Mandiant Consulting incident response teams commonly observe that more traditional disaster recovery approaches, focused primarily on technical restoration, often fall short in addressing the complexities of recovering from a cyber event, particularly the need to re-establish trust with third parties. Notably, we are seeing financially motivated threat groups increasingly targeting backup infrastructures in support of their primary objective. For example, UNC2165, who has leveraged multiple ransomware families including RANSOMHUB, has accessed victim cloud-based data backups, deleted backup routines and existing data, and modified user permissions to hinder response and recovery efforts. Additionally, we have observed UNC4393, previously associated with BASTA ransomware, and UNC2465, previously associated with multiple ransomware families including DARKSIDE and LOCKBIT, targeting backup platforms.

## Common Recovery Challenges

Our experts observed that common recovery challenges, particularly in the aftermath of large-scale cyber attacks like ransomware, often stem from several critical issues, including backup data unavailability, production capacity limitations due to forensic investigations, prolonged recovery times, lack of accessible recovery plans if stored on the production environment, and undefined Recovery Time Objectives (RTOs) and Recovery Point Objectives (RPOs).

Additionally, ransomware can impact various infrastructure dependencies, such as authentication services (Active Directory), DNS, DHCP, virtualization platforms, and security tooling, which are essential for interfacing with backup systems for orchestration, leading to a series of required infrastructure service recovery activities before backups systems can be accessed.

## Common Recovery Architectures

Common recovery strategies and architectures often revolve around how organizations restore their systems and data after an incident, especially a widespread one like a ransomware attack. The specific approach that an organization chooses depends on the state in which an attacker leaves the environment and what has been done prior to an attack to prepare for recovery.

We recommend that organizations consider the RTO of the business when designing a recovery architecture. RTO requirements, combined with modeling associated threat tactics, can help organizations balance the considerations (including associated risks) of the recovery architecture strategy. Some of the various considerations, balanced with the overall resiliency and RTO requirements for common recovery architectures include (Fig. 2):

- **Cloud Isolated Recovery Environment (CIRE):** A recovery operations scenario from a data vault in combination with a CIRE provides additional robustness and helps address specific threat actor risks, while also requiring additional preparation and investment.
- **Isolated Data Vault:** Recovery from a data vault utilizes recent backups from a separate data vault and can help preserve data integrity and availability, but is prone to the same disadvantages as a strategy that only relies on existing production backups and systems.
- **Online/Production-Integrated Backups:** Recovery from production backups requires functioning production backup data, servers, and systems and may be prone to threat actor disruption.

Highest

Resiliency

Lowest

| Backup / Recovery Architecture | Advantages | Considerations |
|---|---|---|
| **Isolated Recovery Environment** | • Highest resiliency to destructive attacks<br>• Backup data unlikely to be directly accessible by an attacker<br>• Backup systems and dependent infrastructure is isolated and pre-staged<br>• Integrated validation capabilities for assurance | • Additional operational and administrative planning, governance, and maintenance requirements |
| **Isolated Data Vault** | • Backup data unlikely to be directly accessible or modifiable by an attacker | • Moderate resiliency to destructive attacks<br>• Requires functioning (available) production backup servers to access backup data<br>• If suspected to be compromised, forensic examination may delay availability of production systems for secure data restoration<br>• Restoration of backup platforms (and dependent systems) will likely delay restoration efforts |
| **Online / Production Integrated Backups** | • Low capital and operational expenses<br>• Ease of integration and management | • Could provide direct access to backups for an attacker, based on the interconnected nature of the backup platforms and credentials that are integrated with a centralized cloud identity provider<br>• Requires functioning (available) production backup servers to access backup data<br>• If suspected to be compromised, forensic examination may delay availability of production systems for secure data restoration<br>• Restoration of backup platforms and dependent systems will likely delay restoration efforts |

Shortest

Recovery Time

Longest

Figure 2. Backup / Recovery Architectures: Balancing Resiliency and Recovery Time

## Mitigations

To counter the evolving threats from threat actors targeting victim backup infrastructure, we recommend establishing CIRE as a strategy for enhancing cyber resiliency and minimizing the impact of cyber incidents. A CIRE is comprised of several components, including an Isolated Data Vault (IDV) or backup vault for immutable and indelible backups, an Isolated Validation Environment (IVE) for restoring, testing, and cleaning data from an IDV before production, and an IRE Services Layer for securing and isolating infrastructure maintaining the CIRE.

The concepts that provide the foundation for the CIRE can be built using the inherent capabilities of the cloud platform itself, such as:

- **Logical Isolation & Segmentation:** Enforce separate boundaries between production and cloud-based backup infrastructure by designing separate VPC networks for your production and backup environments. Use VPC Service Controls to help mitigate risk that compromised credentials could access your backup environment. Enforce identity segmentation by leveraging stringent identity and access management (IAM) controls that require dedicated roles and service accounts to protect the management and recovery plane of the CIRE.
- **Immutable Storage (Isolated Data Vault—IDV):** Leverage Cloud Storage with Object Versioning and Bucket Lock (using Retention Policies) to create a tamper-proof, ransomware-resilient vault for recovering critical data.
- **On-Demand Compute & Validation (Isolated Validation Environment—IVE):** Utilize the elasticity of Compute Engine (and potentially Google Kubernetes Engine for containerized workloads) to rapidly provision resources for data restoration. Data integrity and validation should be completed prior to restoration, including reviewing known indicators of compromise using capabilities within Google Security Command Center (SCC) or custom solutions.
- **Cloud-Native Security Services:** Integrate platform security tools such as SCC Premium for vulnerability scanning, threat detection, and virtual red teaming, Google Security Operations for SIEM/SOAR capabilities, and Artifact Registry for secure image repositories, all to continuously secure the CIRE infrastructure and processes.
- **Secure Out-of-Band Management:** Leverage secure, independent access mechanisms like Identity-Aware Proxy for granular, identity-based access to CIRE resources (e.g., Compute Engine instances) without VPNs or OS Login with IAM for controlled SSH access.
- **Geographic Redundancy:** Leverage Google Cloud's global infrastructure by replicating data using Google Cloud Storage (regional or multi-regional buckets) and recovery environments across different regions or availability zones (e.g., using Managed Instance Groups for Compute Engine or replicated database services like Cloud SQL or Spanner).

# North Korea's Social Engineering Leading to Cloud Compromises and Cryptocurrency Thefts

The cloud security landscape faces persistent threats from highly sophisticated, state-sponsored actors. Google Threat Intelligence Group (GTIG) is actively tracking one such prominent group, UNC4899, which is assessed with high confidence to be a North Korean threat actor aligned with the Reconnaissance General Bureau, which overlaps with public reporting on TraderTraitor. Active since at least 2020, UNC4899 primarily targets the cryptocurrency and blockchain industries and has demonstrated a sophisticated capability to execute complex supply chain compromises.

A notable example is their suspected exploitation of JumpCloud, which they leveraged to infiltrate a software solutions entity and subsequently victimize downstream customers within the cryptocurrency vertical, underscoring the cascading risks posed by such advanced adversaries.

## A Tale of Two Thefts

Between Q3 2024 and Q1 2025, Mandiant responded to two incidents at two separate organizations that we attribute to UNC4899, with one incident affecting a victim's Google Cloud environment and the other affecting a victim's AWS environment. The two incidents bore similarities with how the actors carried out the initial and final phases of their intrusions, but differed with how they carried out the individual activities in the attack lifecycle in between, likely due to how the victims' environments differed.

## Rooted in Socials

In the initial phase of attack lifecycle, UNC4899 targeted victim employees in both organizations by introducing themselves over social media, one over Telegram while in the other incident over LinkedIn. Under the guise of freelance opportunities for software development work, UNC4899 leveraged social engineering techniques to successfully convince the targeted employees to execute malicious Docker containers in their respective workstations. This led to the execution of downloaders, such as GLASSCANNON and secondary payloads including the backdoors PLOTTWIST and MAZEWIRE, before eventually establishing connections to actor-controlled command-and-control (C2) infrastructures. In both cases, UNC4899 conducted several internal reconnaissance activities on the victims' hosts and connected environments, before obtaining credential materials they used to pivot to the victims' cloud environments.

## Hands in Different Credential Jars

During the intrusion in a victim's Google Cloud environment, the actor leveraged stolen credentials from the victim's host to remotely interact over Google Cloud CLI over an anonymous VPN service. While performing internal reconnaissance, UNC4899 enumerated several compute instances, including bastions, services, and resources, before eventually identifying hosts critical to conducting cryptocurrency transactions. They also identified several other locally stored credentials including SSH keys from the bastions and related hosts but were unsuccessful in leveraging them due to the MFA configuration applied to these credentials. When the actors attempted to initiate illicit MFA requests, the legitimate users never approved them. UNC4899 eventually determined the victim's account had administrative privileges to the Google Cloud project and disabled the MFA requirements. After successfully gaining access to the targeted resources, they immediately re-enabled MFA to evade detection. Several days after the actors initially contacted the victim on Telegram, UNC4899 successfully withdrew several millions worth of cryptocurrency (Fig. 3).

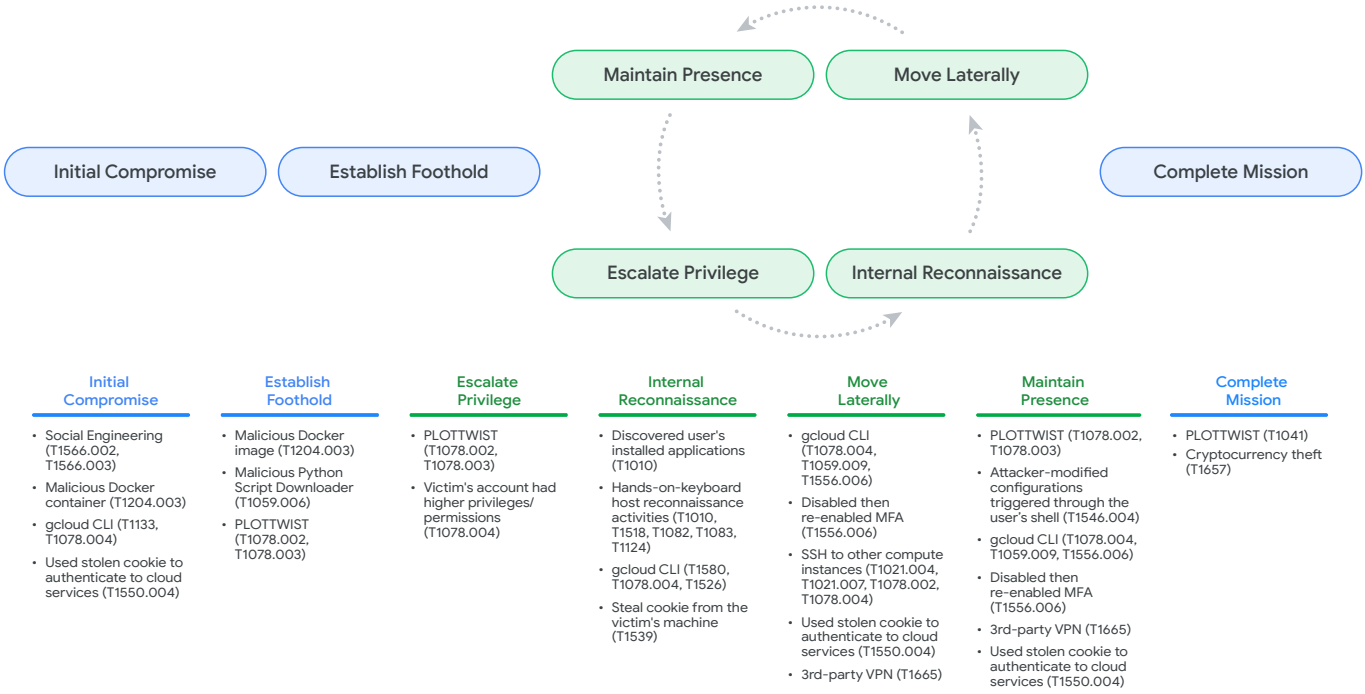



Figure 3. UNC4899 Attack Lifecycle in a Victim's Google Cloud Environment that Led to Cryptocurrency Theft

In the intrusion in a separate victim's AWS environment, UNC4899 initially used long-term access keys retrieved from an AWS credential file found in the victim's host to remotely interact with AWS CLI. However, the actors could not conduct sensitive interactions since the organization applied identity and access management (IAM) policy restrictions that enforced the need to use temporary credentials obtained through the platform's security token service. Additionally, temporary credentials could not be generated without a valid MFA device and neither could they illicitly register MFA devices they controlled since a policy required an existing MFA device to validate any enrollment process. Confronted with these hurdles, evidence indicates that UNC4899 likely resorted to exfiltrating the user's session cookies instead.

Armed with stolen cookies, UNC4899 identified relevant CloudFront configurations and S3 buckets to target, and leveraged the inherent administrative permissions applied to their access to upload and replace existing JavaScript files with those containing malicious code, which were designed to manipulate cryptocurrency functions and trigger a transaction with the cryptocurrency wallet of a target organization. Soon after, UNC4899 successfully stole several million dollars worth of cryptocurrency (Fig. 4).

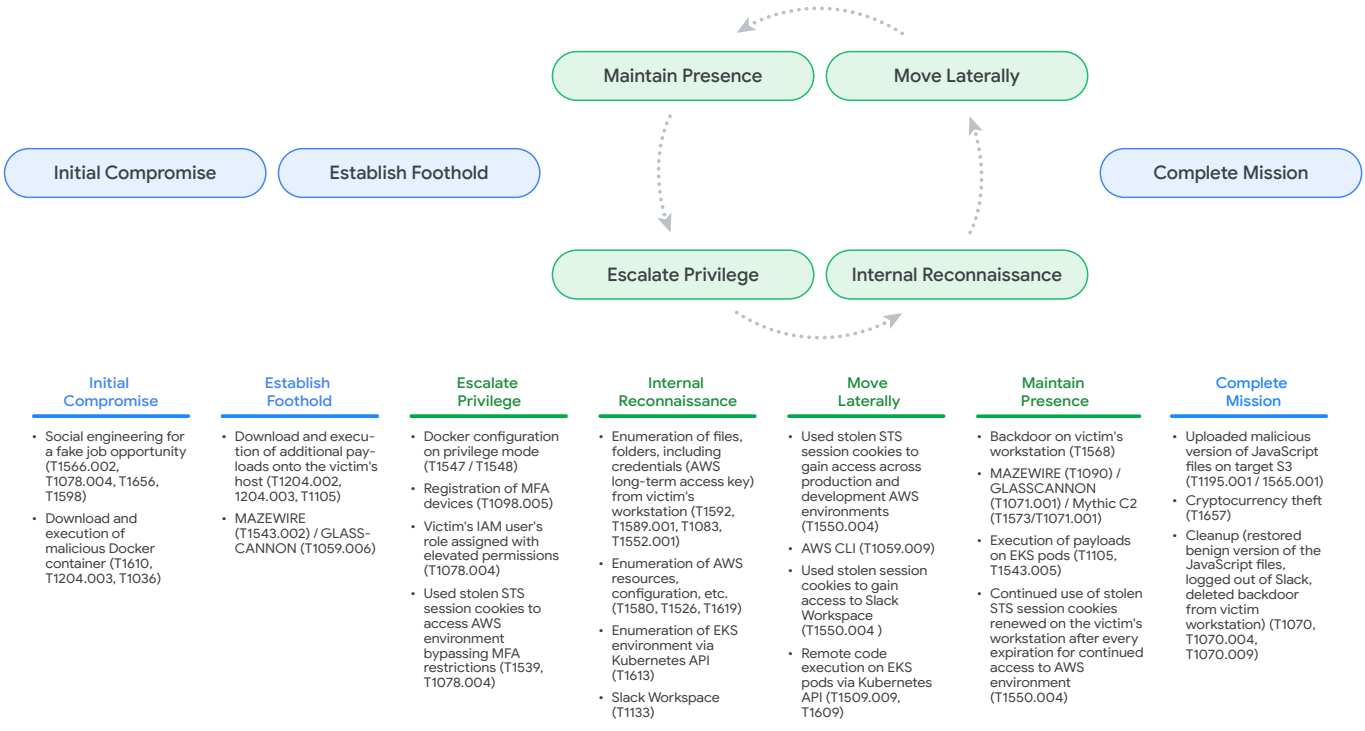



Figure 4. UNC4899 Attack Lifecycle in a Victim's AWS Environment that Led to Cryptocurrency Theft

# Mitigations

Google Cloud offers multiple, robust capabilities to directly counter threat actor tactics displayed by groups like UNC4899. We've mapped the threat group's methods into a targeted action plan to help harden your Google Cloud defenses:

| Threat Actor Method | Effective Security Controls |
|---|---|
| **Exploiting trust through social engineering and cloud services:** UNC4899 effectively leveraged social engineering tactics, from inviting the victims into a fake freelance opportunity for software development work, to convincing them into trusting, downloading, and executing the malicious software downloaded from unknown GitHub repositories. UNC4899 also continued to interact with the victim over social media in one of the incidents, asking exploratory questions about their workstation, cloud environment, and other sensitive information for the duration of their intrusion. | MFA requirements hindered the threat actor several times from progressing through their intrusions even in situations where they had valid cloud credentials. Effective user education may have also contributed to defending against MFA fatigue attacks, since the actors were unsuccessful in luring legitimate users into approving any of the illicit MFA requests.<br><br>Additional IAM security offerings also contributed in hindering the actors from conducting sensitive actions without proper tokens and without the presence of the MFA device. Even the actor's attempt at illicit MFA device registration required an existing, valid MFA device to complete enrollment.<br><br>To help defend against threat actors from bypassing MFA, exploiting stolen session tokens/ cookies, and exploiting trust via social engineering, organizations should:<br><br>**Fortify Identity with MFA & Session Management:** Prevent credential misuse by enforcing MFA across all accounts using Identity and Access Management and Identity Platform. Implement strict session controls, potentially enhanced by Chrome Enterprise Premium, to reduce the attack window from stolen credentials or cookies.<br><br>**Enhance Endpoint & Cloud Workload Threat Detection:** Extend visibility by integrating existing EDR telemetry with Google Security Operations for centralized analysis and threat hunting. Leverage Google Security Command Center (SCC) to identify threats and vulnerabilities on cloud-native workloads. Secure access from endpoints to cloud resources using Chrome Enterprise Premium for context-aware controls. |
| **Unsecure cloud credentials:** UNC4899 leveraged unsecured credentials found on both of the victims' hosts and environments. The actor leveraged several SSH keys to gain access to compute instances and other hosts in the environment, while long-term access keys retrieved from an AWS credential file found in the other victim's host allowed the actor to interact remotely via API over CLI. | To help prevent threat actors from leveraging unsecured cloud credentials:<br><br>**See Fortify Identity with MFA & Session Management recommendation above.**<br><br>**Implement Granular Segmentation & Zero Trust:** Contain threats by limiting lateral movement with network controls like Google Cloud Firewalls and VPC Service Controls for perimeter defense. Apply zero trust principles to application and resource access using Identity-Aware Proxy (IAP). Manage and monitor outbound traffic from sensitive segments using tools like Secure Web Proxy, Cloud NAT, and VPC Flow Logs integrated with Google Security Operations or SCC. |
| **Unsecure processes for code review and CI/CD pipelines:** UNC4899 found weaknesses in the environment and leveraged them in order to upload and replace existing JavaScript files with malicious versions in circumvention of recommended integration and deployment practices. | To help prevent threat actors from manipulating code review processes & CI/CD pipelines:<br><br>**Secure Software Development & Supply Chains:** Protect your CI/CD pipelines by securely managing build artifacts with Artifact Registry, which offers vulnerability scanning. Ensure only trusted code is deployed using Binary Authorization for environments like Google Kubernetes Engine (GKE). Enforce least privilege for pipeline service accounts with IAM recommendations and consider Workload Identity Federation for external CI/CD systems. |

# How Decoy Files Turn Trusted Cloud Services into Attack Vectors

Threat actors are increasingly co-opting trusted cloud storage services as a key component in their initial attack chains, deceptively using these platforms to host seemingly benign decoy files, often PDFs. Analysis from Google's Threat Intelligence Platform reveals that both sophisticated APT groups and cybercriminals leverage this tactic to mislead victims and facilitate malware execution or further system compromise. This abuse of legitimate services for malicious ends presents a significant challenge for defenders, as it blends malicious activity with everyday employee use of cloud platforms, underscoring the need for advanced detection and robust security hygiene.

The modus operandi that we have observed is that a threat actor tricks the user into opening a malicious link or file (e.g., a document with a harmful macro). This launches malicious code that secretly performs harmful actions while displaying a decoy PDF to the user, often stored in cloud storage services or even dropped by the malware.

## Recognizing Decoy Files in Cloud Storage

Threat groups find cloud storage platform services valuable for exfiltrating stolen data[1] by uploading it to legitimate online services and for storing harmful software[2] that can later be downloaded to infected systems during an attack. Spotting this kind of malicious use is challenging for companies because their employees regularly use these same cloud services for normal business, making it hard to tell the bad traffic from the good.

While employing cloud storage for data exfiltration and malicious code hosting, threat actors also leverage these platforms to host seemingly benign files. For example, threat actors frequently use services such as Google Drive, Microsoft SharePoint, Dropbox, and GitHub to host PDF documents for later download and delivery to victims during the initial stages of an intrusion (Fig.5).

1 "Threat Actor Spotlight: UNC2165 Ransomware and Data Theft Extortion", H1 2025 Google Cloud Threat Horizons Report, pp. 16-20.
2 "Cloud-Hosted Encrypted ZIP Files Evading Detection", April 2023 Google Cloud Threat Horizons Report, p. 13.

The purpose of these PDF files is to deceive the victim into believing they have opened a legitimate PDF or another type of document, while various malicious activities are surreptitiously executed in the background. It is important to note, however, that our observations also confirmed instances where these deceptive PDF files were distributed directly from infrastructure controlled by the threat actors themselves, rather than solely relying on compromised or public cloud services. This deceptive use of seemingly harmless files, often PDFs, hosted on trusted cloud services is a subtle yet effective technique for threat actors.

Another increasingly common trend we are following with threat actors using decoy files in cloud storage is that threat actors are using Trojanized PDF files to execute malicious code on systems. Our intelligence teams have investigated multiple campaigns where threat actors have used malicious PDF files, even limiting their execution to Trojanized PDF readers, to ensure their malicious execution is limited solely to the successful deployment of said readers.

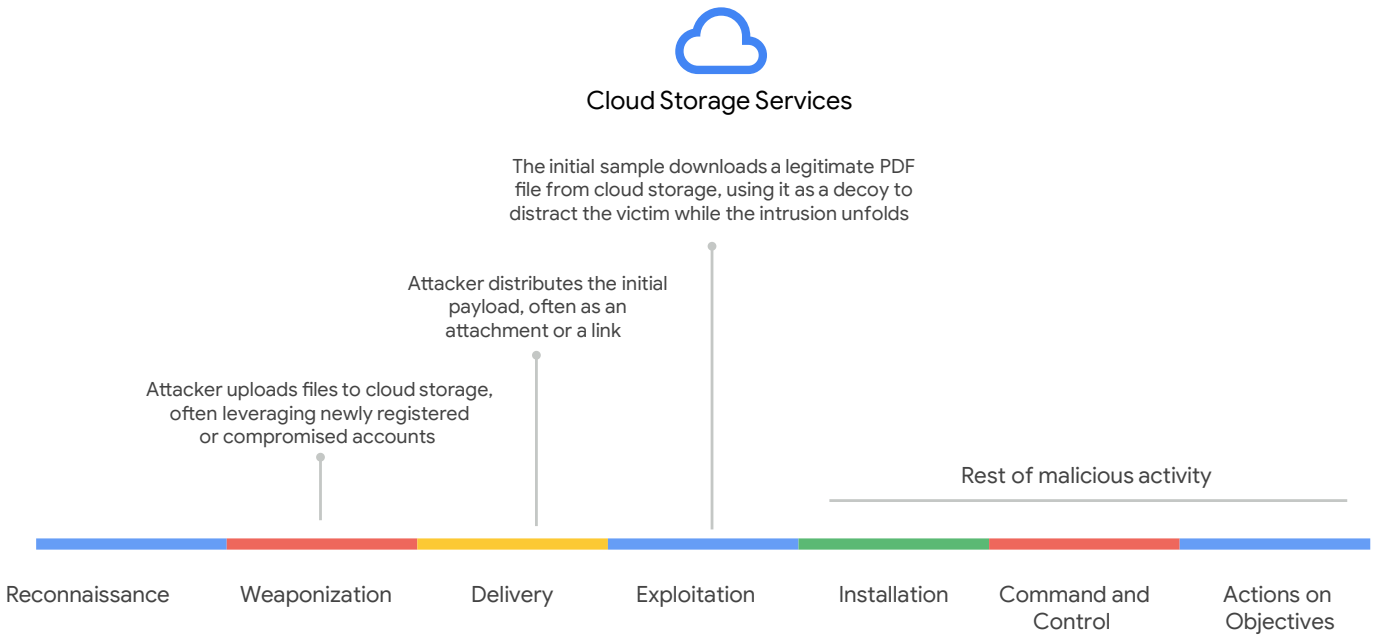



Figure 5. Common Kill Chain for Decoy File Distribution in Cloud Storage Services

Key elements exploited by threat actors in these scenarios are the following:

**Decoy document:** The PDF (or other document type like a Microsoft Word file or spreadsheet) serves as a "front" for other malicious files or links that execute malware when the victim opens the malicious file or link, they see what appears to be normal content—perhaps an invoice, a report, an article, or even a blank document. This reassures the victim that nothing is amiss, reducing suspicion and the likelihood of them immediately closing the file or investigating further.

**Trust in cloud services:** By hosting these decoy files on reputable cloud storage services or code repositories like GitHub (which can also store various file types), threat actors increase the perceived legitimacy of the download. Users are generally less suspicious of links leading to these well-known domains compared to a random, unfamiliar website. Furthermore, this technique is effective because it often sidesteps traditional security measures. Basic network firewalls or email gateway filters might not flag downloads from trusted cloud domains, especially if the decoy file itself lacks a known malicious signature. The true malicious activity is deferred, occurring only after the user interacts with the seemingly innocuous file. At the same time it's relatively easy and cheap for threat actors to create accounts on cloud storage platforms, sometimes anonymously or using stolen credentials, giving them readily available infrastructure.

**Background malicious activities:** While the victim is looking at the decoy document, a variety of malicious actions can be initiated in the background by other malicious files or links, including downloading and executing malware, running scripts for reconnaissance, establishing persistence, exploiting vulnerabilities, and data exfiltration.

Decoy files, such as the PDFs highlighted in this analysis, often operate with considerable subtlety. While these documents may not always be inherently malicious themselves or directly execute harmful code upon opening, their true purpose within a threat actor's strategy is to deceive the victim and serve as an unobtrusive entry point for broader, more damaging intrusions.

The effectiveness of using decoy files can be amplified when threat actors integrate them into sophisticated social engineering schemes. For example, APT42 operations often spoof legitimate institutions and high-profile individuals in their spear-phishing emails to gain trust and lower the victim's guard. This can involve impersonating trusted entities like journalists, event organizers, or even specific high-ranking individuals within organizations, sometimes using typosquatting domains to enhance the facade's believability (Fig. 6).

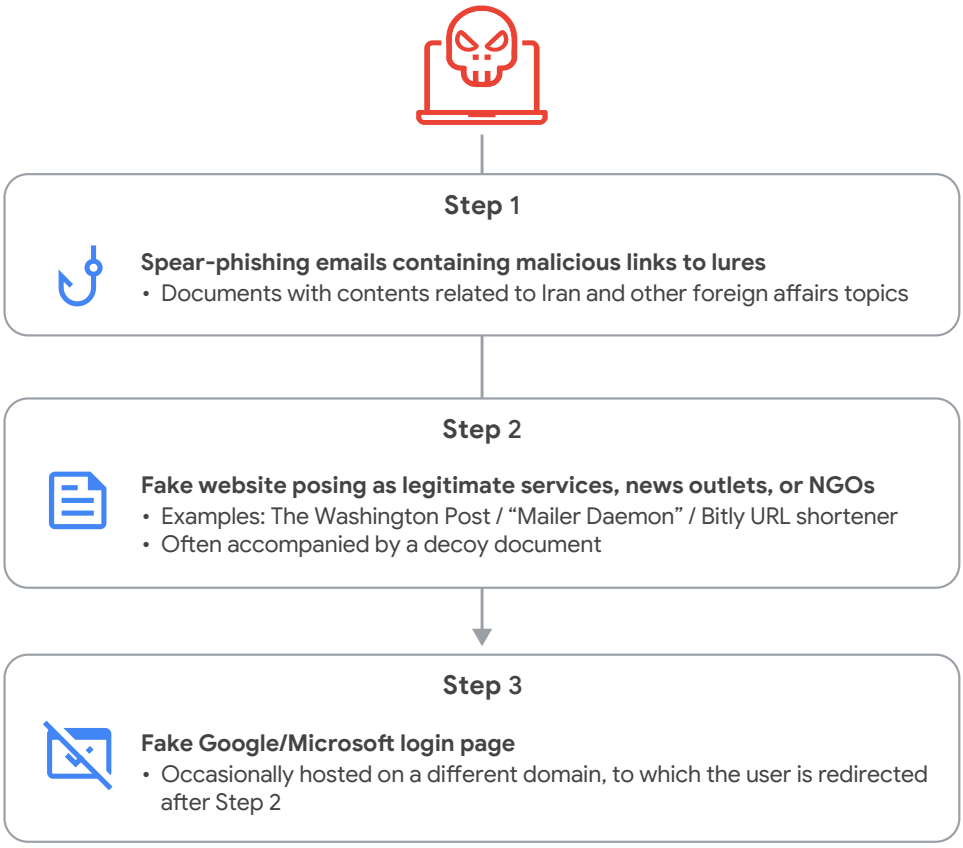



Figure 6. APT42 Credential Harvesting Campaign Attack Lifecycle

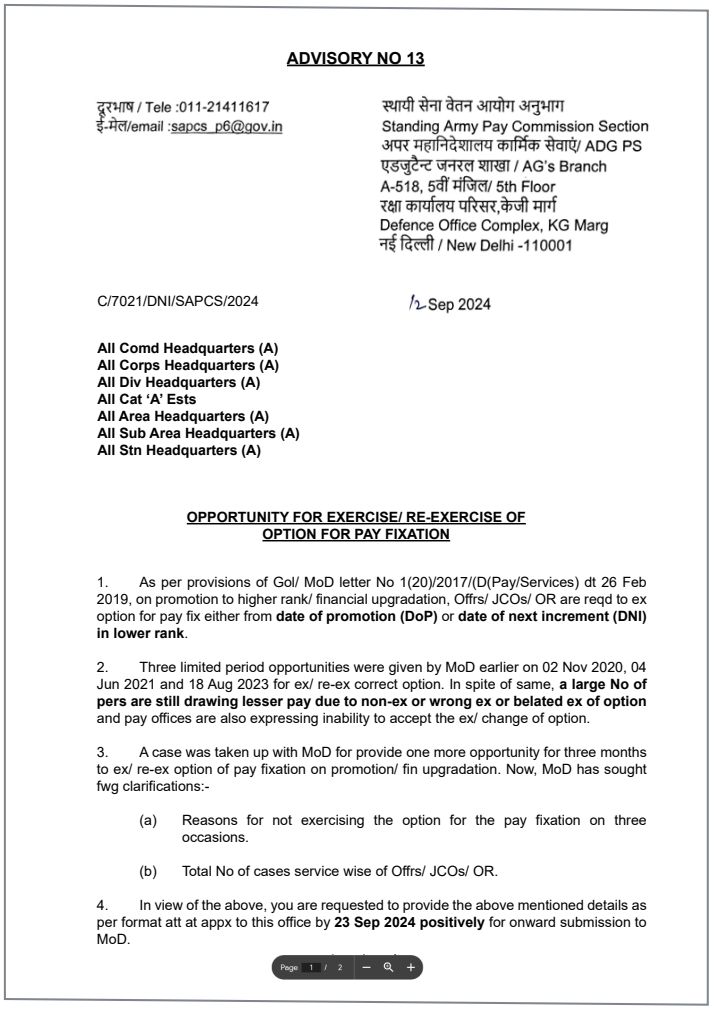
## Decoy Files in Linux-Targeted Campaigns

In Q1 2025, using Google Threat Intelligence platform, we identified multiple uploads by threat actors involving the malicious use of .desktop files in combination with benign PDF files. These .desktop files, native to Unix-like environments and typically used as application shortcuts, were observed being actively leveraged by threat actors. While direct confirmation is pending, analysis strongly suggests these .desktop files are functioning as a first-stage dropper mechanism. Their primary objective appears to be the download of external, secondary payloads from threat actor-controlled infrastructure, which are then executed on the victim's system. Concurrently, these malicious files trigger the display of a PDF document, often hosted on a cloud storage service. This decoy is strategically designed to occupy and reassure the victim, while the malicious payloads downloaded during the intrusion operate stealthily in the background without raising immediate suspicion.

For example, we observed threat actors using decoy files disguised as official government communications. The subject of one advisory was "OPPORTUNITY FOR EXERCISE/ RE-EXERCISE OF OPTION FOR PAY FIXATION". This decoy file was stored in multiple cloud service providers' cloud storage services, which the threat actors used when distributing .desktop files (Fig. 7).

## Mitigations

Given that threat actors leverage diverse trusted cloud storage platforms to host decoy files and malicious files, a comprehensive and platform-agnostic defense strategy is crucial. The primary goal is to identify and prevent interaction with suspicious files before they can execute malicious activity and to contain any potential exploits if a file is opened.



**ADVISORY NO 13**

दूरभाष / Tele :011-21411617
ई-मेल/email :sapcs_p6@gov.in

स्थायी सेना वेतन आयोग अनुभाग
Standing Army Pay Commission Section
अपर महानिदेशालय कार्मिक सेवाएं/ ADG PS
एडजुटैन्ट जनरल शाखा / AG's Branch
A-518, 5वीं मंजिल/ 5th Floor
रक्षा कार्यालय परिसर,केजी मार्ग
Defence Office Complex, KG Marg
नई दिल्ली / New Delhi -110001

C/7021/DNI/SAPCS/2024

12 Sep 2024

**All Comd Headquarters (A)**
**All Corps Headquarters (A)**
**All Div Headquarters (A)**
**All Cat 'A' Ests**
**All Area Headquarters (A)**
**All Sub Area Headquarters (A)**
**All Stn Headquarters (A)**

**OPPORTUNITY FOR EXERCISE/ RE-EXERCISE OF OPTION FOR PAY FIXATION**

1. As per provisions of GoI/ MoD letter No 1(20)/2017/(D(Pay/Services) dt 26 Feb 2019, on promotion to higher rank/ financial upgradation, Offrs/ JCOs/ OR are reqd to ex option for pay fix either from **date of promotion (DoP)** or **date of next increment (DNI) in lower rank**.

2. Three limited period opportunities were given by MoD earlier on 02 Nov 2020, 04 Jun 2021 and 18 Aug 2023 for ex/ re-ex correct option. In spite of same, **a large No of pers are still drawing lesser pay due to non-ex or wrong ex or belated ex of option** and pay offices are also expressing inability to accept the ex/ change of option.

3. A case was taken up with MoD for provide one more opportunity for three months to ex/ re-ex option of pay fixation on promotion/ fin upgradation. Now, MoD has sought fwg clarifications:-

(a) Reasons for not exercising the option for the pay fixation on three occasions.

(b) Total No of cases service wise of Offrs/ JCOs/ OR.

4. In view of the above, you are requested to provide the above mentioned details as per format att at appx to this office by **23 Sep 2024 positively** for onward submission to MoD.

Page 1 / 2

Figure 7. Example of Decoy File

Google Cloud recommends the following mitigations to help organizations defend against threat actors abusing cloud storage services for decoy delivery and subsequent malicious activity:

- **User Awareness and Social Engineering Defense:** Conduct frequent, engaging security awareness training that highlights the sophistication of social engineering tactics. Emphasize that even seemingly legitimate links or attachments from familiar cloud services can be malicious, especially when combined with urgent or enticing pretexts.
- **Inbound File Inspection and Analysis (Pre-Execution):** Utilize advanced email security and secure web gateway solutions that go beyond basic filtering. These should include capabilities for URL Sandboxing/Rewriting, since first payloads that are distributed usually are downloaded from external resources, even from cloud storage services. Then these payloads are responsible for loading decoy files while other payloads are downloaded/dropped to perform malicious activities.
- **Endpoint Detection and Response:** Monitor unusual process trees like PDF files or readers launching command and scripting interpreters (e.g., powershell.exe, cmd.exe). Monitor processes from uncommon paths connecting to cloud storage services such as files stored in temp folders trying to connect to your cloud storage service or opening documents like a PDF or a Microsoft Word document. Monitor the creation of PDF files in temp folders or in the desktop by uncommon processes or processes with specific extensions (e.g., .exe, .dll, .lnk, .desktop). These types of event-based threats can be detected by writing custom rules within Google Security Operations using its YARA-L 2.0 detection language.

# Hardening the Chrome Extension Supply Chain Against Account Compromise

If the account of a Chrome Web Store (CWS) developer is compromised, it can have a significant impact on users and their organizations because it can allow a threat actor to distribute malicious updates to every developer with the compromised extension installed. An account compromise could occur when a developer is tricked into granting a malicious OAuth client access to their resources, a trend we expect to continue following the wave of OAuth phishing attacks in late 2024. CWS mitigates this risk by enforcing controls against the human identities of developer accounts with features like mandatory multi-factor authentication (MFA) and frequent re-authentication checks for the developer dashboard. However, since non-human identities used for automated code build processes cannot use MFA and still represent a risk the Google CWS team introduced the Verified CRX Upload feature to provide defense-in-depth and counter malicious updates stemming from compromised developer accounts.

## Background

Developers typically upload their extensions to the CWS as ZIP files. CWS reviews the contents of the items and signs them with two keys:

**First Key:** "Google" key indicating Chrome Web Store has processed this extension.

**Second Key:** "Developer" key, historically, developers would sign using the developer key themselves, but because the developer key determines the identity of the extension, it can't be rotated and in practice most developers preferred to have Google manage the key material rather than risk handling it themselves.

## Evolving Threat Account Authentication

While CWS managing the signing keys mitigated certain risks, it shifted the critical security perimeter to developer account authentication. A malicious actor gaining unauthorized access to a developer's CWS Developer Dashboard or misusing API access (e.g., by misleading extension developers into granting them the OAuth permission) could potentially upload malicious updates.

Before CWS accepts an upload, it verifies that the uploading account has proper access to the extension. To mitigate the impact of a developer being phished, CWS implemented MFA for registered developer accounts and mandates frequent re-authentication for the developer dashboard.

However, API access using OAuth tokens, a form of non-human identity, is not subject to these same security advancements. If a developer grants the

OAuth scope that allows the extension management permission to the OAuth client associated with a malicious project, that malicious OAuth client will be able to leverage the account owner's established reputation to widely distribute a malicious extension.

CWS employs automatic and human reviews and Chrome has protection mechanisms like requiring explicit user consent for new permissions. However, threat actors are capable of cleverly disguising malicious code because some harmful functionality might not be fully apparent or ascertainable during the review process, as its safety can be context-dependent.

## Mitigations

Google actively monitors for, and has suspended, known malicious Google Cloud projects attempting to abuse CWS APIs. Furthermore, CWS has tightened the approval process for future Google Cloud projects requesting the OAuth scope for extension management, reducing the attack surface for API-based compromises.

**Implement Verified CRX Upload for Stronger Authentication:** Launched in May 2025, developers can opt-in to a more secure Verified CRX Upload flow, which introduces a second factor for the extension upload process. We recommend developers opt in to the Verified CRX upload feature and apply extra measures when possible such as implementing MFA (even in programmatic access cases like APIs), separating ACLs clearly, and passing cryptographic proofs through minimally-trusted layers.

When opting an extension into Verified CRX Upload, the developer gives Google a public key. After that, the developer can no longer upload unsigned ZIP files for that extension and must instead upload a CRX file signed with the corresponding private key. Unlike the "developer" key, this key can be rotated if lost or leaked by contacting CWS admins and may involve a waiting process and/or other interactive defenses. Crucially, the private key remains under the developer's control and is not uploaded to CWS.

Verified upload acts as a second factor for the act of uploading to CWS. While account authorization (e.g., login, OAuth token) serves as the first factor, the second factor requires using a unique private key to sign the extension package before uploading it. A malicious actor who compromises a developer's account password, session cookies, or even an OAuth token, would not be able to upload a malicious update unless they also gain access to the developer's private signing key. This adds a crucial layer of protection against account compromise affecting the integrity of the extension supply chain (Fig. 8).

Sophisticated users can take the feature even further by separating their signing environment permissions from their uploader permissions, which can further improve their resilience to supply-chain attacks within their own organizations.

**Promote Secure Developer Key Management Practices:** The Verified CRX Upload feature is currently opt-in, allowing developers to transition to a more secure workflow. To ensure added security is effective, developers should store their private key in a highly secure location, independent of their Google

account credentials. The private key must not be uploaded to public repositories or stored in personal cloud storage, such as Google Drive, because if threat actors compromise developer account credentials, they would also gain access to this key. Storing the private key securely using a dedicated keystore solution, like PKCS#12 or Java Keystore, is strongly recommended.

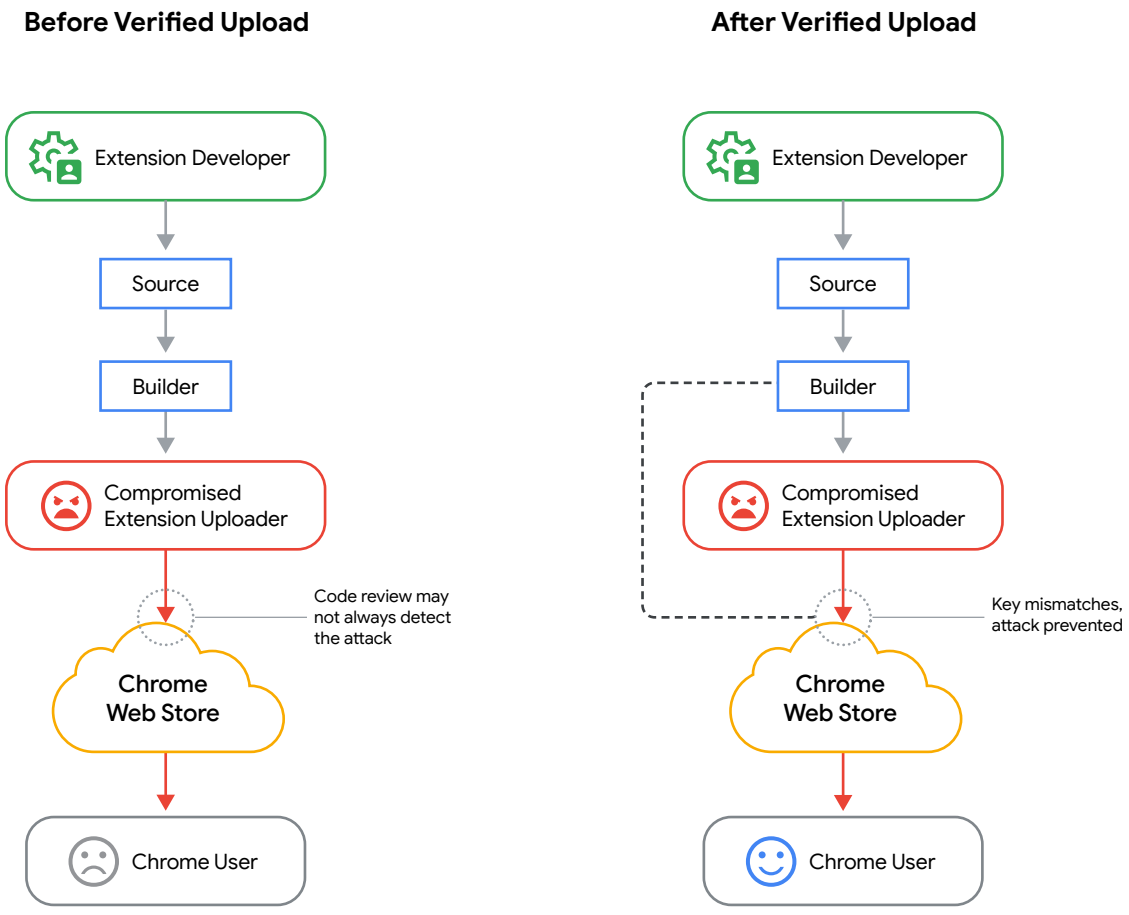



Figure 8. How Verified CRX Upload Helps Protect Users

# Contributors

Jason Bisson

Chary Chen

Anton Chuvakin

Michael Cote

Charles DeBeck

Elliot Eaton

Jack Fermon

Christopher Liebchen

Crystal Lister

Angelus Llanos

Jose Luis Sanchez Martinez

Bob Mechler

Noah McDonald

Glenn Staniforth

Chris Linklater

Matthew McWhirt

Joachim Metz

Simon Scannell

John Stone

Google Cloud