# UNC4899 Living-off-the-Cloud: AirDrop-to-Cloud-Compromise and DevOps Workflow Abuse

Unofficial AI-assisted Research

Cloud Security Alliance AI Safety Initiative

2026-03-10

 

## Key Takeaways

A newly reported incident attributed to UNC4899—a North Korean state-nexus threat actor also tracked as TraderTraitor, Jade Sleet, and Slow Pisces—demonstrates a documented evolution in the group's tactics, extending prior social engineering patterns to include proximity-transfer vectors and systematic DevOps pipeline abuse. Rather than relying on novel malware or zero-day exploits, UNC4899 has refined its approach to weaponize the trusted tools and workflows of modern software development. In the most recently disclosed intrusion, the attack chain began with social engineering that induced a developer to transfer a trojanized archive onto their corporate macOS device via Apple AirDrop, and ended with authenticated access to a Google Cloud environment, manipulation of Kubernetes deployments, and exfiltration of CI/CD service account tokens sufficient to facilitate cryptocurrency theft.

The technique represents a systematic and operationally refined application of Living-off-the-Cloud (LotC) principles: once inside the corporate endpoint, the adversary conducted virtually all lateral movement and persistence using legitimate cloud platform APIs, container orchestration tooling, and DevOps pipeline features that are both authorized and ubiquitous in the environments it targets. This approach significantly complicates detection because malicious activity is difficult to distinguish from routine developer operations. Security teams face a detection problem that is fundamentally different from identifying novel malware—they must instead identify authorized tools being used with subtle behavioral anomalies.

This note examines the AirDrop-to-cloud-compromise attack chain, the broader LotC threat landscape UNC4899 operates within, and the specific DevOps workflow abuse techniques the group has demonstrated. Immediate actions for defending cryptocurrency-sector and cloud-native organizations are provided at the conclusion.

# Background

## UNC4899: Attribution and Operational Mandate

UNC4899 is assessed by Mandiant with high confidence to operate under the direction of North Korea's Reconnaissance General Bureau (RGB), the country's primary foreign intelligence service [1]. The group operates under several overlapping tracking designations across the threat intelligence community: Mandiant designates it UNC4899, Microsoft uses Jade Sleet, Recorded Future uses Slow Pisces, and CrowdStrike—which split its prior LABYRINTH CHOLLIMA designation into three distinct subunits in 2025—now tracks the group under PRESSURE CHOLLIMA [3][10]. The operational mandate is narrow but financially consequential: UNC4899 focuses almost exclusively on the cryptocurrency and fintech sectors, with the theft proceeds widely assessed to fund North Korea's weapons program [6].

The operational record is substantial. In June 2023, UNC4899 executed a supply chain attack against JumpCloud, a cloud-based identity and access management provider, injecting a malicious Ruby script into fewer than five downstream customers—all cryptocurrency firms—via the JumpCloud agent [4][8]. The group came to wider prominence in February 2025 when it carried out the Bybit exchange compromise, ultimately stealing approximately $1.5 billion USD in Ethereum—the largest single financial cyber theft on record at the time [5][6]. That operation began when a Safe{Wallet} developer's Apple macOS laptop was compromised through a trojanized Docker project delivered via social engineering on February 4, 2025; attackers subsequently hijacked AWS session tokens, accessed cloud storage buckets and Terraform configuration files [7], and ultimately deployed malicious JavaScript that silently redirected a Bybit cold wallet transaction [5][6].

A mid-2025 operational security lapse, referenced in threat intelligence reporting, involved UNC4899 operators reportedly connecting from infrastructure traceable to the Ryugyong-dong district of Pyongyang, briefly exposing true origin infrastructure despite routine VPN obfuscation. The incident is consistent with the broader attribution picture suggesting that the group operates as a disciplined, well-resourced unit within a formal intelligence structure. By Chainalysis estimates, North Korean threat actors as a collective stole at least $2.02 billion in cryptocurrency in 2025—a 51% year-over-year increase—with the CrowdStrike 2026 Global Threat Report attributing $1.46 billion of that figure specifically to PRESSURE CHOLLIMA [9][10].

### The Living-off-the-Cloud Paradigm

Living-off-the-Cloud (LotC) is the strategic extension of Living-off-the-Land (LotL) techniques into cloud environments. Where LotL describes adversaries using operating system-native utilities—PowerShell, WMI, certutil—to conduct malicious operations without deploying detectable payloads [11], LotC extends this approach into cloud environments, with adversaries routing command-and-control communications, data exfiltration, and lateral movement through legitimate cloud services: Google Drive, OneDrive, Dropbox, GitHub repositories, AWS services, and cloud-native DevOps tooling [10]. The core advantage is the same in both cases: when malicious traffic is structurally identical to authorized traffic, signature-based and network-layer controls offer limited detection value.

The CrowdStrike 2026 Global Threat Report documented that 82% of all detections across monitored environments were malware-free—adversaries operated through valid credentials, trusted identity flows, and approved SaaS integrations rather than through traditional malware deployment [10]. Cloud-conscious intrusions rose 37% overall in the same period, with a 266% increase among state-nexus actors specifically. The related concept of Living-Off-Trusted-Sites (LOTS) describes a subset of LotC behavior in which adversaries weaponize major trusted domains—GitHub, Google Docs, Cloudflare—as staging and communication infrastructure, deliberately exploiting the domain reputation that enterprise proxy controls rely on to allow traffic [12]. MITRE ATT&CK documents these behaviors under techniques including T1567.002 (Exfiltration to Cloud Storage) and related cloud-focused sub-techniques.

---

## Security Analysis

### The AirDrop-to-Cloud-Compromise Attack Chain

The most recently reported UNC4899 intrusion, disclosed in March 2026 via sources including The Hacker News and InfoSec Today, describes an attack chain that warrants detailed examination because it combines several independently familiar risk factors into a sequence that many common enterprise control configurations are not currently tuned to break, though targeted mitigations are described later in this note [13][14].

The operation began with social engineering. UNC4899 contacted a developer at a cryptocurrency firm, presenting a request framed as a technical collaboration on an open-source project—a recurring cover story for the group, which has previously used job offer lures and collaborative Docker troubleshooting requests as pretexts [15]. The developer was induced to download a trojanized archive, and critically, to transfer it from a personal device to their corporate macOS workstation using Apple AirDrop.

The AirDrop transfer step deserves specific attention because it represents a deliberate exploitation of the boundary between personal and corporate device policies. Many enterprise mobile device management and endpoint security configurations focus primarily on network-based delivery vectors—email attachments, web downloads, removable media—and may not inspect or restrict inbound AirDrop transfers, which occur over a direct Bluetooth/Wi-Fi peer-to-peer connection that bypasses corporate proxies and network security monitoring [13]. An employee receiving a file from what appears to be their own personal device may apply less scrutiny than they would to an emailed attachment from an unknown sender—a trust dynamic that social engineering research consistently identifies as a significant vulnerability. Once on the corporate device, the archive was opened in an AI-assisted integrated development environment—a detail that reflects UNC4899's accurate model of the developer workflow, including the standard trust extended to code viewed within a development tool context.

Execution of the embedded malicious Python script deployed a binary masquerading as the Kubernetes CLI tool `kubectl`. The choice is operationally effective: `kubectl` is among the most privileged tools a cloud-native developer routinely uses, it makes outbound HTTPS connections as a matter of normal operation, and its presence on a developer workstation is unremarkable and would not normally trigger suspicion from endpoint or network monitoring tools. The fake binary established a backdoor to an attacker-controlled domain while mimicking the behavioral signature of a legitimate Kubernetes client.

## DevOps Workflow Abuse at Scale

Once a foothold was established on the developer's machine, UNC4899's subsequent actions were conducted almost entirely through legitimate cloud and DevOps APIs. The adversary modified existing Kubernetes deployment configurations to include a bash command that would execute on every new pod creation, automatically downloading a backdoor into freshly launched containers. This persistence mechanism is particularly durable because it survives container recycling—the standard operational response to detecting a compromised running container—and is embedded within infrastructure-as-code artifacts that may be version-controlled and deployed at scale across multiple clusters.

The CI/CD token harvesting step escalated the threat from individual workstation compromise to infrastructure-level access. UNC4899 modified Kubernetes resources associated with the victim's CI/CD platform to inject commands that caused service account tokens—including tokens with elevated privileges—to appear in pipeline logs. This technique exploits a structural characteristic of many CI/CD implementations: pipeline logs are often treated as diagnostic artifacts rather than security-relevant data, and high-privilege service account tokens exposed in log output may persist in log aggregation

systems for extended periods before detection. The harvested token enabled authentication to sensitive infrastructure pods, container escape to the underlying host, and deployment of additional backdoors with durable persistence.

The ultimate impact involved tampering with Cloud SQL database configurations to facilitate cryptocurrency theft, though the precise mechanism of that final stage has not been fully disclosed in available reporting [13][14]. The overall attack chain demonstrates that a single compromised developer endpoint, given sufficient cloud permissions attached to its authenticated sessions, can serve as an entry point to infrastructure that an organization may believe is protected by network segmentation, MFA, and cloud IAM policies—because the attacker is authenticating with valid credentials rather than exploiting vulnerabilities.

## Structural Parallels in the DevOps Attack Landscape

UNC4899's DevOps workflow abuse techniques occur within a broader threat landscape that has seen sustained adversary investment in CI/CD pipeline compromise. The `tj-actions/changed-files` GitHub Actions supply chain attack of March 2025 offers a useful parallel: attackers who compromised a GitHub Personal Access Token used by a trusted GitHub Action injected code that caused CI/CD runner memory to be dumped into workflow logs, exposing API keys, PATs, npm tokens, and RSA private keys across more than 23,000 affected users of the action [16]. That campaign was assigned CVE-2025-30066 (CVSS 8.6) and generated a CISA advisory [17]. The initial target was Coinbase's `agentkit` open-source repository, with attackers seeking to use Coinbase's CI pipeline as a stepping stone; Coinbase's secrets were not ultimately exfiltrated.

AWS's own CI infrastructure was implicated in the Wiz Research "CodeBreach" disclosure (AWS Security Bulletin 2026-002), in which a regex filter lacking proper anchoring in an AWS CodeBuild pipeline for the `aws-sdk-js-v3` repository allowed unauthenticated GitHub users to trigger builds and potentially exfiltrate a GitHub PAT with full administrative access over multiple AWS SDK repositories [18]. Research from Orca Security published in late 2025 similarly identified vulnerable `pull_request_target` GitHub Actions workflows in repositories belonging to major technology companies, where exploitation could achieve secrets exfiltration or malicious code injection into trusted branches [19].

The pattern across these incidents is consistent with the behavior observed in the UNC4899 intrusion: CI/CD pipelines that are treated as trusted by design—because they are orchestrating authorized build, test, and deployment operations—provide a persistent, high-privilege access pathway that is under-monitored relative to its actual security sensitivity. Mandiant's M-Trends 2025 report noted cases where

adversaries modified cloud-native data pipeline configurations, specifically Azure Data Factory and AirByte instances, to exfiltrate data from integrated SaaS platforms, further illustrating how DevOps tooling designed for data movement is repurposed for data theft [20].

### The macOS Developer Endpoint as Strategic Target

Each major publicly documented UNC4899 intrusion has involved macOS as the initial endpoint target [10][14], reflecting the group's deliberate focus on developer environments. This pattern is consistent with the group's sectoral focus: developers working in cryptocurrency infrastructure are generally believed to skew toward macOS as their primary development platform, and macOS developer workstations routinely hold the cluster of credentials—cloud provider session tokens, container registry credentials, CI/CD service account files, cryptocurrency wallet key material—that UNC4899's operational objectives require. Moonlock research published in early 2026 documented a significant increase in credential theft targeting macOS [21], and Google Cloud's Cloud Threat Horizons H2 2025 report assessed that over 70% of cloud breaches originate from compromised identities [22].

A 2024 vulnerability disclosure, CVE-2024-44131, is relevant context: the vulnerability allowed a malicious application to bypass macOS Transparency, Consent, and Control (TCC) protections and copy extensive user data from iCloud Drive without user notification or consent [23]. Apple patched the vulnerability in macOS Sequoia 15, but the disclosure illustrates that the macOS security model, while more restrictive than Windows in many respects, has specific weaknesses in the interaction between local application permissions and cloud-synchronized storage. An adversary who establishes a foothold through a developer-executed payload—as UNC4899 does—is operating within the TCC permissions that the developer themselves have granted to the IDE or development tooling being abused, often bypassing the protective intent of the TCC framework entirely.

---

## Recommendations

### Immediate Actions

Organizations in the cryptocurrency sector, financial services, and cloud-native software development should treat this intrusion pattern as a high-priority threat requiring active defensive posture review. The following actions are appropriate for immediate implementation.

- **Restrict AirDrop and proximity transfer capabilities on corporate macOS devices.** Mobile device management policies should enforce AirDrop restrictions to contacts only, or

disable the feature entirely on corporate devices. Similarly, Bluetooth-based file transfer should be evaluated in the context of the organization's threat model.

- **Audit cloud service account permissions attached to developer workstations.** Credentials stored in `~/.kube/config`, `~/.aws/credentials`, application default credential files, and similar locations should be inventoried and scoped to the minimum privilege required. Long-lived credentials with broad permissions represent disproportionate risk given the demonstrated UNC4899 capability to harvest them silently.
- **Enable CI/CD secret scanning with alerts for token exposure in logs.** GitHub's secret scanning feature, GitLab's secret detection, and equivalent controls in other platforms should be activated and configured to alert immediately on service account token patterns appearing in pipeline output. Tokens appearing in logs should be considered compromised and rotated immediately.
- **Pin GitHub Actions to specific commit SHAs.** Mutable tag references (e.g., `uses: some-action@v2`) allow supply chain substitution attacks of the type demonstrated in the `tj-actions` incident. Pinning to verified commit SHAs is a concrete, low-friction mitigation for this class of risk.

## Short-Term Mitigations

Over a period of weeks, security and platform engineering teams should implement monitoring and policy controls that reduce the attacker's ability to operate undetected within cloud and DevOps environments.

Container and Kubernetes workloads should be audited for recently modified deployment configurations, particularly those referencing external URLs in init containers, lifecycle hooks, or environment variable injection patterns. Kubernetes RBAC should be reviewed to ensure CI/CD service accounts do not hold permissions that would allow pod creation or container exec on sensitive infrastructure namespaces. Anomaly detection for `kubectl exec` and similar commands—particularly those originating from identities that do not routinely perform such operations—should be implemented through cloud audit logging and forwarded to SIEM for correlation.

Log retention policies for CI/CD pipeline output should be reviewed with the assumption that pipeline logs are a target for token harvesting. Secrets exposed in historical logs should be rotated regardless of whether active exploitation has been detected; the presence of a token in a log is sufficient grounds for rotation. Organizations using AI-assisted IDEs should evaluate whether those tools automatically execute code, display output from arbitrary files, or provide elevated trust to artifacts opened within the IDE context—the UNC4899 incident exploited precisely this workflow trust.

Social engineering awareness training should be updated to specifically address the scenario of technical collaboration offers that involve running or testing third-party code, including Docker projects, Python scripts, or CLI tools delivered through messaging platforms. The Bybit incident, the September 2024 Telegram-based PLOTTWIST deployment, and the March 2026 AirDrop incident share a common social engineering structure: an unsolicited but technically credible collaboration request that results in code execution on a developer's machine [24].

## Strategic Considerations

At the strategic level, the UNC4899 intrusion pattern surfaces a structural tension in cloud-native security: the same ease of API-driven automation that makes DevOps productive also makes it an efficient attack surface. Cloud IAM policies, Kubernetes RBAC, and CI/CD pipeline permissions are frequently designed for access control within a trusted organization but are not designed to constrain a fully authenticated adversary operating with harvested credentials. Zero Trust architecture principles—continuous verification, least privilege, assume breach—provide a well-aligned strategic framework for this threat model, addressing the core gap exploited by credential-abusing adversaries.

Organizations should invest in cloud detection and response capabilities that establish behavioral baselines for how their developers normally interact with cloud APIs, CI/CD systems, and container orchestration. Deviation from those baselines—particularly lateral movement from a developer identity into infrastructure pods, or service account token usage from unexpected geographic locations or client types—should generate high-priority alerts. Given that UNC4899 has demonstrated the ability to operate within a compromised environment for multiple days before achieving its final objective—as illustrated by the 17-day window between initial macOS compromise on February 4 and the fraudulent transaction in the Bybit intrusion [5][7]—detection at any point in the kill chain before cloud database access is reached would limit impact significantly.

The nexus of this attack surface is the developer laptop: it sits at the intersection of personal and corporate trust boundaries, holds credentials for production infrastructure, and is routinely used to execute code from external sources as a matter of professional function. Endpoint detection and response (EDR) solutions on developer macOS devices, combined with cloud access security broker (CASB) enforcement of what cloud resources an endpoint is authorized to access, represent the most direct defensive investments for organizations that consider themselves within UNC4899's targeting envelope.

---

# CSA Resource Alignment

This research note connects to several active CSA frameworks and guidance documents relevant to the threat behaviors described.

The **MAESTRO Agentic AI Threat Modeling** framework is relevant to the AI-assisted IDE abuse vector identified in this intrusion. As agentic development tools acquire broader permissions to read codebases, execute commands, and interact with external services, they create expanded attack surfaces that threat actors can exploit by deceiving developers into opening malicious artifacts within the IDE context. MAESTRO's AI Deployment Layer and Trust Boundary threat categories are directly applicable to modeling how a trojanized file opened in an AI IDE can achieve code execution.

The **Cloud Controls Matrix (CCM)** provides specific applicable controls. CCM domain SEF (Security Incident Management, E-Discovery, and Cloud Forensics) is implicated in the challenge of detecting LotC intrusions where malicious activity blends with authorized operations. CCM domain IAM (Identity and Access Management) addresses the credential hygiene and least-privilege principles that would limit the blast radius of the developer endpoint compromise demonstrated here. CCM domain DSP (Data Security and Privacy) covers the Cloud SQL tampering and data exfiltration dimensions of the attack.

**Zero Trust guidance** published by CSA applies directly to the core structural vulnerability exploited: that cloud environments protected by perimeter and IAM controls can be fully traversed by an adversary holding valid credentials. The CSA Software-Defined Perimeter (SDP) specification and Zero Trust Architecture guidance both address the need for continuous verification even within the cloud perimeter—precisely the control gap that UNC4899's credential-abuse approach exploits.

The **AI Organizational Responsibilities** framework is relevant to the governance dimension: organizations adopting AI-assisted development tooling should assess the permissions and trust levels granted to those tools, including what access AI IDEs have to credential files, shell execution, and cloud API interaction. The UNC4899 incident illustrates an emerging attack vector in which an AI-assisted IDE served as the execution context for a malicious payload in a high-impact nation-state intrusion—a development that underscores the importance of formally assessing the permissions and trust levels granted to AI development tooling in enterprise environments.

---

# References


[1] Mandiant / Google Cloud Threat Intelligence, "North Korea Leverages SaaS Provider in a Targeted Supply Chain Attack," Google Cloud Blog, 2023. https://cloud.google.com/blog/topics/threat-intelligence/north-korea-supply-chain

[3] SecurityBrief Asia, "CrowdStrike Splits LABYRINTH CHOLLIMA Into Three Units," 2025. https://securitybrief.asia/story/crowdstrike-splits-labyrinth-chollima-into-three-units *(URL returning 403 as of 2026-03-10; see [10] for PRESSURE CHOLLIMA attribution)*

[4] TechTarget, "Mandiant: JumpCloud Breach Led to Supply Chain Attack," 2023. https://www.techtarget.com/searchsecurity/news/366545498/Mandiant-JumpCloud-breach-led-to-supply-chain-attack

[5] The Hacker News, "Safe{Wallet} Confirms North Korean TraderTraitor Hackers Stole $1.5 Billion in Bybit Heist," March 2025. https://thehackernews.com/2025/03/safewallet-confirms-north-korean.html

[6] FBI Internet Crime Complaint Center (IC3), "North Korea Responsible for $1.5 Billion Bybit Hack," PSA250226, February 26, 2025. https://www.ic3.gov/psa/2025/psa250226

[7] Sygnia, "Sygnia Investigation: The Bybit Hack," 2025. https://www.sygnia.co/blog/sygnia-investigation-bybit-hack/

[8] Codenotary, "Decoding the UNC4899 Supply Chain Attack," 2023. https://codenotary.com/blog/decoding-the-unc4899-supply-chain-attack

[9] Chainalysis, "North Korea Drives Record $2 Billion Crypto Theft Year, Pushing All-Time Total to $6.75 Billion," 2026. https://www.chainalysis.com/blog/crypto-hacking-stolen-funds-2026/

[10] CrowdStrike, "CrowdStrike 2026 Global Threat Report: Evasive Adversary Wields AI," 2026. https://www.crowdstrike.com/en-us/blog/crowdstrike-2026-global-threat-report-findings/

[11] Fortinet, "Living-off-the-Land (LotL) Attacks," Fortinet Cyber Glossary, 2025. https://www.fortinet.com/resources/cyberglossary/living-off-the-land-lotl

[12] MITRE ATT&CK, "T1567.002: Exfiltration to Cloud Storage," https://attack.mitre.org/techniques/T1567/002/


 

[13] The Hacker News, "UNC4899 Breached Crypto Firm After Developer AirDropped Trojanized File to Work Device," March 9, 2026. https://thehackernews.com/2026/03/unc4899-used-airdrop-file-transfer-and.html

[14] InfoSec Today, "UNC4899 Breached Crypto Firm After Developer AirDropped Trojanized File to Work Device," March 2026. https://www.infosectoday.io/unc4899-breached-crypto-firm-after-developer-airdropped-trojanized-file-to-work-device/

[15] Elastic Security Labs, "Bit ByBit," 2025. https://www.elastic.co/security-labs/bit-bybit

[16] Unit 42, Palo Alto Networks, "GitHub Actions Supply Chain Attack," March 2025. https://unit42.paloaltonetworks.com/github-actions-supply-chain-attack/

[17] CISA, "Supply Chain Compromise of Third-Party GitHub Action, CVE-2025-30066," Alert, March 18, 2025. https://www.cisa.gov/news-events/alerts/2025/03/18/supply-chain-compromise-third-party-github-action-cve-2025-30066

[18] Wiz Research, "CodeBreach: Critical Vulnerability in AWS CodeBuild CI Pipeline (AWS Security Bulletin 2026-002)," 2025. https://www.wiz.io/blog/wiz-research-codebreach-vulnerability-aws-codebuild *(Note: The vulnerability identifier is AWS Security Bulletin 2026-002 per the Wiz disclosure; a CVE assignment was not confirmed in the source article.)*

[19] Orca Security, "Pull Request Nightmare Part 2: Exploits," 2025. https://orca.security/resources/blog/pull-request-nightmare-part-2-exploits/

[20] Mandiant / Google Cloud Threat Intelligence, "M-Trends 2025," 2025. https://cloud.google.com/blog/topics/threat-intelligence/m-trends-2025/

[21] Moonlock, "6 Key Trends in macOS Malware 2026," 2026. https://moonlock.com/macos-malware-trends-2026

[22] Google Cloud, "Cloud Threat Horizons H2 2025," 2025. https://cloud.google.com/resources/content/cloud-threat-horizons-report-h2-2025

[23] Jamf, "TCC Bypass Steals Data from iCloud (CVE-2024-44131)," 2024. https://www.jamf.com/blog/tcc-bypass-steals-data-from-icloud/

[24] Ainvest / The Hacker News, "Bybit Hack Linked to Compromised Developer Laptop, DPRK Ties," March 2025. https://www.ainvest.com/news/bybit-hack-linked-compromised-developer-laptop-dprk-ties-2503/

---

*Editorial note: Reference [2] from the source draft (a duplicate entry for the THN article at [13]) has been removed. The CrowdStrike PRESSURE CHOLLIMA tracking designation previously cited to [2] is now supported by [10]. See styling metadata in frontmatter for full fix inventory.*