

Financially Motivated Threat Actor BREEZE COMET Targets Brazil

 cloud.google.com/blog/topics/threat-intelligence/financially-motivated-threat-actor-breeze-comet-targets-brazil

Google Threat Intelligence Group, Mandiant

September 1, 2026



Google Cloud

Threat Intelligence

Google Threat Intelligence Group

Mandiant

Mandiant Services

Stop attacks, reduce risk, and advance your security.

[Contact Mandiant](#)

Introduction

Beginning in 2024 Mandiant investigated a string of compromises affecting Brazilian financial services, retail, and eCommerce organizations. Google Threat Intelligence Group (GTIG) tracks this activity as BREEZE COMET (formerly UNC5669), a financially motivated threat actor specializing in manipulating payment systems and banking software in Brazil to conduct fraudulent transfers. This activity overlaps with operations publicly reported as [Plump Spider](#) and [SHADOW-AETHER-064](#). In this blog, we detail BREEZE COMET's tactics and toolkit, and provide mitigation recommendations and detections to support organizations in defending against this active and developing threat.

BREEZE COMET tactics have evolved over time to leverage a customized malware suite and compromised, trusted websites to facilitate initial access, command and control (C2), and to interact with financial software and payment APIs. BREEZE COMET's operational infrastructure may also indicate intent to expand their infrastructure footprint to other countries in Latin America and Africa. Additionally, we have evidence that BREEZE COMET is using generative artificial intelligence (AI) to support malware development, which may further increase the scale, speed, and sophistication of their operations in the future.

BREEZE COMET Targets Brazilian Financial Technology

BREEZE COMET operations target organizations with permission to conduct transactions through banking software, APIs, and payment systems such as Pix, STR, and Boletão. This typically includes banks, payment processors, retailers, exchanges, as well as fintech and banking software providers.

To achieve their objective of conducting fraudulent transfers, BREEZE COMET must maintain:

- Access to the National Financial System Network (Rede Nacional do Setor Financeiro, RSFN) through an entity with this access.
- Access to mTLS credentials that allow sending authenticated payloads with transactional orders to Pix, STR (Brazilian Reserves Transfer System), or any transactional listener to be executed with minimal restrictions in the name of an organization with available funds.
- Persistent access to multiple accounts in targeted organizations' Active Directory and/or cloud environments.
- Understanding of an organization's transfer processing procedures, network controls, fintech integrations and anti-fraud systems.

In order to support these requirements, BREEZE COMET evolved to operate in multiple compromised environments at the same time, crafting custom C2 malware to automate activities such as reconnaissance, lateral movement, persistence, and exfiltration.

Initial Compromise and Establish Foothold

BREEZE COMET has used various methods for initial access. In early compromises, Mandiant observed this threat actor use password spraying as well as voice calls impersonating IT support teams to convince users to install Remote Monitoring and Management (RMM) tools such as AnyDesk. [Axur](#) corroborates use of voice phishing, and suggests that the group has also attempted to recruit insiders at targeted organizations.

In mid-2025, GTIG observed BREEZE COMET using compromised Brazilian small government websites to stage RMM tools, infostealers disguised as legitimate tax or receipt documents (e.g., `ComprovantePDF.exe`), or backdoors such as XWORM set to persist via automated startup shortcut modifications. XWORM is a backdoor that is widely available for purchase on cyber crime forums, with leaked or “cracked” versions also available. BREEZE COMET then used these compromised government websites to facilitate social engineering operations for initial access, and as C2 endpoints. The use of compromised, trusted infrastructure allowed the threat actors to avoid detection by network domain reputation filters. GTIG also observed BREEZE COMET replicating this behavior with municipal domains in Nigeria, Paraguay, Ghana, and Venezuela, suggesting a potentially growing targeting focus. Analysis of compromised municipal domains indicated that BREEZE COMET reused the same staging infrastructure to host and deliver XWORM payloads across operations targeting multiple organizations.

In 2025, we first observed BREEZE COMET connect rogue hardware devices directly into retail store networks to establish footholds into targeted environments. From this initial network access, BREEZE COMET moved laterally to internal systems then downloaded the Netcat utility alongside custom scripts to pull down subsequent post-exploitation frameworks from external open directories. [Trend Micro](#) has reported that the group also exploited vulnerabilities in JBoss AS servers to gain initial access.

Escalate Privileges & Internal Reconnaissance

BREEZE COMET used publicly available reconnaissance utilities such as Impacket, ADRecon and ADVipscan, as well as with custom malware, often profiting from environments with low observability. These utilities were often observed being downloaded from GitHub repositories and executed in memory via PowerShell for defense evasion.

The threat actor deployed the custom LDAP brute-forcing utility **REALBREEZE**. Beyond traditional Active Directory compromise, BREEZE COMET specifically targets development and cloud environments to escalate privileges. The group actively mines continuous integration and continuous delivery (CI/CD) environments to steal hard-coded pipeline credentials, application programming interface (API) keys, and highly privileged cloud access tokens.

BREEZE COMET used custom scripts to search internal host files and environmental variables to identify mTLS credentials and administrative certificates necessary to authenticate against core banking systems. Observed search terms included: `boleto`, `cnab`, `remessa`, `webhook.*pix` and `instant.*payment`.

Move Laterally

BREEZE COMET abuses standard protocols to navigate the network, using hijacked service accounts to initiate unauthorized Remote Desktop Protocol (RDP) sessions and execute commands via SMB network file shares. BREEZE COMET was observed executing network scanning tools across internal subnets specifically to enumerate available SMB pathways.

To maneuver through segmented financial networks and bypass strict internal firewalls, BREEZE COMET deploys specialized routing malware: **COBALTSPIN**. Written in Rust, COBALTSPIN operates as a lightweight, evasive network tunneler, used to communicate with and maintain persistent network access to financial API infrastructure. By establishing a reverse SOCKS5 proxy over a WebSocket connection, COBALTSPIN routes network traffic securely back and forth between the C2 and internal targets, enabling lateral movement directly through boundary firewalls without requiring built-in persistence mechanisms that might trigger detection.

Maintain Presence: Orchestrating the Compromise via Bespoke C2 Frameworks

In 2024, BREEZE COMET relied on commercial RMM tools to maintain access to targeted environments. In 2025, BREEZE COMET also deployed malicious Kubernetes pods to maintain persistence and steal cloud secrets, exfiltrating them to public facing notepad websites (such as `dontpad[.]com`).

In 2025 and 2026 Mandiant identified multiple backdoors that BREEZE COMET developed to establish redundant access and expand their foothold in targeted environments.

- **LIGHTPAINT**: This custom Java-based backdoor is specifically designed to install a legitimate VPN, such as SoftEther, and configure it for automated persistence. To protect this access, GTIG observed BREEZE COMET programmatically adding inbound Windows Defender Firewall rules to allow all traffic from the deployed VPN manager, while subsequently clearing the Windows Networking Vpn Plugin Platform event logs to erase forensic evidence of the connection.
- **MILDFROST**: Operating as a passive Java JAR backdoor hiding inside the JVM process space, MILDFROST uses classes like `DnsCommandBeacon.class` to establish slow, covert DNS tunnels. It also serves as a fallback C2; it dynamically queries delegated subdomains to receive instructions and pull down fresh copies of the C++ executables.

- **KICKPLATE:** To continuously deliver auxiliary payloads and enforce host-level persistence, BREEZE COMET uses KICKPLATE. This custom Nim-based backdoor impersonates Windows Update Health Tools. It executes commands to control SOCKS5 tunnelers, update registry startup keys, and silently modify Windows services. The group supplements KICKPLATE by abusing native scheduled tasks (`schtasks.exe` running as SYSTEM) and malicious shortcut (`.lnk`) modifications in user startup folders.
- **BOATBEAM:** Adding a final layer to their redundant architecture, BREEZE COMET deploys BOATBEAM, a Golang backdoor that initiates a fake IIS HTTPS server on port 443. This artifact hides backdoor traffic by masquerading as a legitimate web server, only activating its C2 functionalities when it receives a specific session cookie.

To ensure these persistence mechanisms survive, BREEZE COMET actively impairs endpoint defenses. Telemetry confirms the threat actors executing direct PowerShell commands (`Set-MpPreference -DisableRealtimeMonitoring $true`) to disable Windows Defender's real-time monitoring across compromised hosts, guaranteeing their malware suite remains operational.

Furthermore, Mandiant identified evidence that BREEZE COMET used large language models (LLMs) to accelerate the creation of custom scripts for network reconnaissance, credential validation, mass deployment, victim-specific pivoting, and data extraction. Analysis of recovered BREEZE COMET scripts has shown the tools are highly customized and functional, but lack human idiosyncrasies, heavily relying on unrolled code structures, verbose explanatory comments, and standardized execution headers.

```
#!/bin/bash
# RODA DENTRO DO 10.0.9.9 - DIRETO NA REDE INTERNA

echo "#####"
echo "### STEP 1: ENUM ALL LINUX (SSH PORT 22) ###"
echo "#####"

# Scan SSH em todos os ranges conhecidos
echo "=== SCANNING SSH PORTS ==="
> /tmp/ssh_open.txt
```

Figure 1: Excerpt of script showing verbose comments

Complete Mission: Mass Fraudulent Transactions

Forensic evidence analyzed by Mandiant demonstrates that BREEZE COMET used COBALTSPIR and compromised privileged accounts to access core financial applications. Within 24-48 hours of establishing this access, the threat actor executed two waves of

hundreds of fraudulent transactions, based on reporting by a client and third party forensic analysis.

Subsequently, BREEZE COMET cleared event logs across compromised hosts to hide evidence of their lateral movement, privilege escalation, and interactions with APIs associated with financial software and payment systems. The attacker also deleted directories they had created during the compromise.

Outlook and Implications

Since 2024, BREEZE COMET has steadily increased the complexity and effectiveness of their operations manipulating Brazilian financial systems and software, and has successfully executed at least one heist of tens of thousands of USD in assets. This analysis is intended to support financial services, fintech, retail, and government organizations, particularly in Brazil, to track and defend against BREEZE COMET.

While the Latin American cybercrime ecosystem has historically been defined by client-side, high-volume retail fraud, BREEZE COMET's campaigns represent a notable shift that may serve as a model for future financially motivated threats against organizations in this region. This transition from opportunistic retail banking fraud to direct intrusions into the core financial switch and instant payment infrastructure is notable not just for this shift in targeting, but also the capabilities of the threat actor.

BREEZE COMET exemplifies how threat actors are operationalizing generative AI to enhance the speed, scale, and sophistication of their campaigns. By leveraging LLMs to generate bespoke reconnaissance scripts, validate credentials, and automate deployment workflows on the fly, the actor compresses the development lifecycle. This automation also lowers the operational threshold required to coordinate synchronized, multi-environment attacks. Finally, orchestrating their usage of AI-generated tooling alongside bespoke multi-language C2 architectures demonstrates how actors can elevate their overall capabilities and lower technical barriers to entry. The progression to a multi-tiered ecosystem—combining custom-built Rust, Nim, and Go backdoors with AI-accelerated operational scripts—demonstrates a measurable maturation in BREEZE COMET's technical capability.

As threat groups increasingly leverage LLMs to streamline routine tradecraft, defenders must anticipate shorter adversary turnaround times and heightened pressure on interconnected financial ecosystems.

Remediation and Hardening

Application Control & Unapproved Remote Management (RMM) Blocking

- Enforce Application Control (e.g. Windows WDAC, macOS Gatekeeper/MDM, or Linux fapolicyd) to block execution in user-writable directories (Windows %APPDATA%, macOS ~/Downloads, Linux /tmp or /var/tmp).
- Partition Linux hosts to mount /tmp and /home with the noexec flag.
- Audit software inventory to alert on portable RMM execution and unapproved system service/daemon registrations.
- Train users on social engineering tactics impersonating IT Support.

Network Access Control & Branch Physical Hardening

- Deploy 802.1X Network Access Control (NAC) across physical Ethernet switch ports at branch/retail locations to prevent unauthorized hardware devices from obtaining an internet protocol (IP) address or communicating on internal subnets.
- Disable unused switch ports and enforce Port Security (e.g. MAC limiting) on critical network drops.
- Physically restrict access to networking closets and secure public-facing jacks.

Active Directory & Credential Hardening

- Restrict administrative utilities (e.g. ntdsutil.exe, vssadmin.exe) and alert on volume shadow copy creation/deletion.
- Enforce PowerShell Constrained Language Mode (CLM), Script Block Logging (Event ID 4104), and Antimalware Scan Interface (AMSI) to detect in-memory execution of reconnaissance scripts.
- Mandate phishing-resistant multifactor authentication (MFA) and lockout controls across all external portals (VPNs, Software-as-a-Service (SaaS)).

Deep Packet Inspection & Egress Traffic Control

- Perform SSL/TLS Decryption and Deep Packet Inspection (DPI) on outbound web traffic rather than relying on domain reputation or .gov top-level domain (TLD) allowlists.
- Block non-essential egress ports and protocols (e.g., outbound Internet Control Message Protocol (ICMP)) and restrict tunneling utilities like Chisel or GSocket).
- Segment networks to block lateral SMB (port 445) and RDP (port 3389) traffic between workstations and servers.

Kubernetes & Cloud Workload Isolation

- Enforce strict Kubernetes Role-Based Access Control (RBAC) using least privilege for service accounts.
- Use dynamic admission controllers (e.g., OPA Gatekeeper or Kyverno) and native Pod Security Admission (PSA) to block privileged containers.
- Apply egress network policies to block nodes and pods from accessing unauthorized public platforms.

Secrets Management & Financial System Micro-Segmentation

- Mandate a centralized Secrets Manager (e.g., HashiCorp Vault) with access logging; eliminate plaintext keys in code.
- Implement identity-based / Layer 7 micro-segmentation for financial workloads.
- Limit administrative access exclusively to dedicated jump hosts via privileged access management (PAM).

Indicators of Compromise (IOCs)

To assist the wider community in hunting and identifying activity outlined in this blog post, we have included indicators of compromise (IOCs) in a [GTI Collection](#) for registered users.

File Indicators

Indicator	Notes
3b22605244dbace8f0c07c2c599f88c4b831bb07e9998b869a5da2759d27ceec	COBALTSPIN
2214907e696bad85bde1d90c943ef66e413d7a5c6d7596ced25b74441200439a	REALBREEZE
c0db6ddd6222d02ad7490399d33c61ded0076f0037409dc8498924458646d78a	MILDFROST
6d4012e0dd3b56a3e52857734fa0d582cdf3c56f0e5decc8005c882d1d1c6ceb	BOATBEAM
f139b4ca15feffb7a6633ec1a431c5c604b397576b56b5c863ae8fe4fa14db4f	KICKPLATE

51fdd83b3737add7f3832bd0ad0b56863c0a8f7cf9bcc16fd787d1ae4b403ce6	XWORM
d2aa40cc53b40c6e76ac0677c4a54387b3f27ee94c85d9b2c3a3d66aeef92a66	XWORM
447e3a131e62bd33b1297739a7b959a92358a97f58554469044636a3c4f244e8	XWORM

Table 1: File Indicators

Network Indicators

Indicator	Notes
dontpad[.]com	Paste site used for data exfiltration
hxxps://procon[.]go[.]gov[.]br/ComprovantePDF[.]exe	Compromised malware Staging Domain
hxxps://cmgovernadorluizrocha[.]ma[.]gov[.]br/Comprovantepdf[.]exe	Compromised malware Staging Domain
hxxp://gcm[.]setelagoas[.]mg[.]gov[.]br/files/ti[.]zip	Compromised malware Staging Domain

hxxp://gcm[.]setelagoas[.]mg[.]gov[.]br/files/notepad[.]exe	Compromised malware Staging Domain
hxxp://gcm[.]setelagoas[.]mg[.]gov[.]br/files/tes[.]exe	Compromised malware Staging Domain
hxxps://minacu[.]go[.]gov[.]br/ComprovantePDF[.]exe	Compromised malware Staging Domain
hxxps://conseg[.]ssp[.]go[.]gov[.]br/COAF-POLICIAFEDERAL[.]exe	Compromised malware Staging Domain
hxxps://conseg[.]ssp[.]go[.]gov[.]br/ComprovanteBBpix[.]exe	Compromised malware Staging Domain
hxxps://suporte[.]camaratunapolis[.]sc[.]gov[.]br/ti/attvpn[.]zip	Compromised malware Staging Domain
hxxps://suporte[.]camaratunapolis[.]sc[.]gov[.]br/ti/1[.]exe	Compromised malware Staging Domain

hxxps://tisup[.]camaratunapolis[.]sc[.]gov[.]br/SoftEther[.]exe	Compromised malware Staging Domain
hxxp://suporte[.]ourinhos[.]sp[.]gov[.]br/files/s[.]zip	Compromised malware Staging Domain
hxxp://suporte[.]ourinhos[.]sp[.]gov[.]br:443/files/s[.]exe	Compromised malware Staging Domain
hxxp://suporte[.]ourinhos[.]sp[.]gov[.]br/files/a[.]exe	Compromised malware Staging Domain
hxxps://servicos[.]salto[.]sp[.]gov[.]br/j[.]jar	Compromised malware Staging Domain
hxxps://www.mrtb[.]gov[.]ng/apps/attvpn[.]vip	Compromised malware Staging Domain
hxxp://credeb[.]gov[.]gn/r[.]zip	Compromised malware Staging Domain

hxxps://sit[.]baer[.]gob[.]ve/r[.]exe	Compromised malware Staging Domain
hxxps://jmcov[.]gov[.]py/cxv[.]exe	Compromised malware Staging Domain

Table 2: Network Indicators

Detections

Google Security Operations (SecOps)

Google SecOps customers have access to these broad category rules and more under the "Mandiant Hunting Rules" rule pack. The activity discussed in the blog post is detected in Google SecOps under the rule names:

- "Network DNS Connections To Pastebin"
- "Powershell Downloadstring Method With Suspicious Arguments"
- "Powershell Loading Net Assembly"

YARA Rules

```
rule M_Utility_REALBREEZE_2 {
  meta:
    author = "Google Threat Intelligence Group"

  strings:
    $s1 = "IP/REDE" wide
    $s2 = "SENHA" wide
    $s3 = "U\x00S\x00U\x00\xc1\x00R\x00I\x000\x00:"
    $s4 = "Arquivo de Texto (*.txt)|*.txt" wide
    $s5 = "get_SamAccountName"
    $s6 = "get_txtHostname"

  condition:
    uint16(0) == 0x5A4D
    and all of them
```

```

}

rule G_Tunneler_COBALTSPIN_1
{
    meta:
        author = "Google Threat Intelligence Group"

    strings:
        $p00_0 = {488985[4]72??4c8b47??4c8b6f??488985[4]eb??4989f04989c5488b85}
        $p00_1 = {4d8bae[4]4d85ed4c897d??897d??4c8975??89b5[4]74??498bbe[4]4d89ee}
    condition:
        uint16(0) == 0x5A4D and uint32(uint32(0x3C)) == 0x00004550 and
        (
            ($p00_0 in (560000..600000) and $p00_1 in (1500000..1600000))
        )
}

rule G_Backdoor_BOATBEAM_1
{
    meta:
        author = "Google Threat Intelligence Group"

    strings:
        $p00_0 = {4d89d84889ce488bbc24[4]e9[4]0f82[4]4c89ac24[4]4c89e74d29ec4c896424}
        $p00_1 = {e8[4]498903498973??498953??4d8943??488942??488957??4889f8488b4c24}
    condition:
        uint16(0) == 0x5A4D and uint32(uint32(0x3C)) == 0x00004550 and
        (
            ($p00_0 in (1500000..1600000) and $p00_1 in (2700000..2800000))
        )
}

rule G_Backdoor_MILDFROST_1
{
    meta:

        author = "Google Threat Intelligence Group"

    strings:
        $s1 = "sc tcp ok" fullword
        $s2 = "fl comando vazio" fullword
        $s3 = "noop" fullword
        $s4 = "wait:" fullword
        $s5 = "shell:" fullword
        $s6 = "exec:" fullword
        $s7 = "upload," fullword
        $s8 = "dl|" fullword
        $s9 = "tc|" fullword
    condition:

```

```
uint16(0)==0x5a4d and 7 of them
```

```
}
```