TLP:CLEAR

ADGM THREAT INTELLIGENCE NEWSLETTER

PREPARED BY ADGM CTI

CATEGORY	ACTIONABLE
AUDIENCE	ADGM FSRA ENTITIES
DATE	02/7/2026
OVERALL THREAT SCORE	ELEVATED
TARGET SECTOR	FINANCIAL SERVICES
TARGET REGION	MENA & GLOBAL
ATTRIBUTION	MULTIPLE
TLP	CLEAR

WEEKLY SUMMARY REPORT – 02 July 2026

11	5	1	0
Campaigns	**Vulnerability**	**Cyber Breach**	**Threat Actors**
Threat Campaigns of Potential Relevance to Financial Sector	Actively Exploited & Critical Vulnerabilities	Major Compromises and Breaches	Threat actor activities in the Middle East impacting Financial Sector

Summary

This week's cybersecurity landscape highlights escalating threats from phishing, Infostealers, ransomware, supply-chain compromises, cryptocurrency theft, and critical enterprise vulnerabilities. Threat actors increasingly leveraged trusted platforms, collaboration tools, messaging apps, browser extensions, and open-source ecosystems to steal credentials, deploy malware, and maintain access. Key developments included the emergence of evolving malware families, several software supply-chain incidents, and a third-party breach that resulted in approximately $3 million in customer losses. From a financial sector perspective, these developments reinforce the importance of protecting identities, development pipelines, cloud environments, and customer-facing applications. Credential theft, unauthorized remote access, malicious dependency abuse, and privilege escalation vulnerabilities may impact critical business operations if left unaddressed. Organizations should prioritize timely patching, strengthen monitoring across endpoints, cloud, and CI/CD environments, enforce multi-factor authentication, review third-party dependencies, and continuously assess exposure to emerging attack techniques and software vulnerabilities.

ADGM THREAT INTELLIGENCE SUMMARY

Multi-Stage LokiBot Infection Chain Observed in Recent Campaign [Campaign] [High]

Outlook Groups Abuse Enables Phishing Through Trusted Collaboration Workflows [Campaign] [High]

VBScript Campaign Deploys Remote Management Software Via WhatsApp for Access [Campaign] [High]

Payouts King Ransomware IAB Affiliate Abusing Microsoft Edge to Deploy New Edgecution Malware [Campaign] [High]

DPRK-Linked macOS Gaslight Combines Credential Theft with LLM Triage Evasion [Campaign] [High]

StealC and Amadey Infostealer Ecosystem Enables Credential Theft [Campaign] [High]

New Mistic Backdoor Linked to ModeloRAT Activity and Ransomware Access [Campaign] [Medium]

Miasma Campaign Expands Across Package and Build Workflows [Campaign] [Medium]

Shai Hulud Campaign Bridging CI Systems and Cloud Environment [Campaign] [Medium]

The Gentlemen RaaS Expands Operations with Evolving Tactics and Custom Tools [Campaign] [Medium]

Steganographic Loader Campaign Delivers Remcos Through Multi-Stage Execution [Campaign] [Medium]

Pedit COW Linux Kernel Flaw Enables Local Privilege Escalation to Root [Vulnerability] [High]

PostgreSQL High-Severity Vulnerabilities Allow Arbitrary Code Execution [Vulnerability] [High]

Critical Google Gemini CLI Vulnerability Enables RCE in CI/CD Workflows [Vulnerability] [High]

Multiple Jenkins Plugin Vulnerabilities Enable RCE, Security Bypass, and Credential Exposure [Vulnerability] [High]

Critical Remote Code Execution Vulnerability in libssh2 [Vulnerability] [High]

Polymarket Supply-Chain Attack Leads to $3 Million Loss Through Malicious Frontend Script [Cyberbreach] [High]

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Multi-Stage LokiBot Infection Chain Observed in Recent Campaign	HIGH	CLEAR	Campaign	Open Source

Executive Summary

Researchers has identified a recent LokiBot campaign that uses a multi-stage infection chain beginning with a malicious JScript email attachment, which ultimately deploys the LokiBot infostealer through PowerShell-based loading and '.NET' process injection techniques. The attack relies on multiple obfuscation and staging layers to conceal execution and hinder analysis.

Organizations in the financial sector should be aware that LokiBot is designed to harvest credentials from numerous applications, exfiltrate collected information, and maintain communication with command-and-control infrastructure for additional instructions. The theft of user credentials could affect environments that rely on browser, email, and other application-based authentication mechanisms.

Technical Details

- The campaign begins with a malicious email containing a JScript attachment, a delivery method commonly used to distribute LokiBot. The script executes when opened on a Windows system.
- The JScript file is heavily obfuscated, using multiple decoding routines and decoy functions to make analysis and detection more difficult.
- Once executed, the script decodes a Base64-encoded PowerShell payload and launches it to continue the infection process.
- The first-stage script also includes cleanup functions that can terminate processes and remove files after execution to reduce evidence of the attack.
- The PowerShell payload decrypts an embedded '.NET' component and loads it directly into memory, helping the attackers avoid writing additional malware files to disk.
- The '.NET' component acts as an injector, preparing and launching the next stage of the attack within a legitimate Windows process.
- During this stage, memory is allocated inside the target process and the final LokiBot payload is written into it before execution begins.
- The deployed LokiBot sample stores encrypted command-and-control information and resolves required system functions at runtime using API hashing techniques.
- After starting, the malware verifies that no other copy is running, then proceeds with credential collection and other core functions.
- The malware gathers credentials from targeted applications, sends the collected data to command-and-control infrastructure, and periodically checks for additional instructions from the attacker.

Recommendations

- Strengthen email security controls to identify and block malicious script-based attachments before delivery.

- Monitor PowerShell activity and suspicious in-memory execution behaviors across endpoints.
- Detect and investigate process injection techniques involving legitimate Windows processes.
- Enforce multi-factor authentication to reduce the risk associated with stolen credentials.
- Monitor for unusual credential access patterns and unexpected outbound communications from user devices.

For a detailed breakdown of tactics, techniques, and procedures (TTPs), refer to the Annexure – Multi-Stage LokiBot Infection Chain Observed in Recent Campaign

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Outlook Groups Abuse Enables Phishing Through Trusted Collaboration Workflows	HIGH	CLEAR	Campaign	Open Source

Executive Summary

Researchers at a security research team has identified phishing activity that abuses Outlook Groups and Microsoft 365 collaboration features to make malicious actions appear as routine business workflows. The technique leverages group invitations, shared resources, mailbox communications, and calendar items to move users through seemingly legitimate interactions before directing them toward malicious actions.

Organizations in the financial sector should be aware that this approach shifts phishing activity beyond traditional email-based attacks and into trusted collaboration environments. The technique could affect users who interact with group invitations, shared files, or calendar events, potentially leading to credential theft, token capture, malware delivery, data exposure, or further social engineering activity.

Technical Details

- The attack begins when a threat actor creates or controls a Microsoft 365 group and adds or invites targeted users where external collaboration is permitted.
- The group is given a convincing name and purpose, making it appear to be a legitimate business function or internal activity.
- Users receive a welcome message that appears legitimate because it is delivered through a trusted cloud collaboration service rather than a spoofed email.
- The attacker establishes a believable context, such as account issues, administrative actions, training requirements, supplier communications, or urgent reviews.
- Follow-up content is delivered through trusted collaboration features, including group mailboxes, shared files, and calendar invitations.

- The campaign leverages calendar-based phishing techniques that move user interaction away from the original email and into calendar events and reminders.
- Calendar entries can repeatedly reappear through notifications and reminders, increasing the likelihood that users will interact with the malicious content.
- Shared files may contain fraudulent support processes, credential-harvesting content, QR-code lures, malware-related content, or instructions designed to advance the attack.
- Users are more likely to trust the content because it is presented through familiar Microsoft 365 collaboration workflows rather than traditional phishing emails.
- Successful interaction with the workflow can result in credential theft, token capture, malware delivery, data exposure, service disruption, or additional social engineering activity.

Recommendations

- Review and restrict external group creation and membership permissions where business requirements permit.
- Monitor Microsoft 365 groups, mailbox activity, shared content, and calendar events as part of phishing investigations.
- Train users to treat unexpected group invitations, meeting requests, and shared files with the same caution as suspicious emails.
- Validate detection capabilities across email, collaboration platforms, shared content, and calendar artefacts rather than focusing solely on email security controls.
- Regularly review workflows involving group membership changes, file sharing, and external collaboration for signs of abuse.

For a detailed breakdown of tactics, techniques, and procedures (TTPs), refer to the Annexure – Outlook Groups Abuse Enables Phishing Through Trusted Collaboration Workflows

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
VBScript Campaign Deploys Remote Management Software Via WhatsApp for Access	HIGH	CLEAR	Campaign	Open Source

Executive Summary

Researchers has identified an active campaign that distributes malicious VBScript files through WhatsApp messages to users of WhatsApp Desktop and WhatsApp Web. The attackers use business and finance-themed file names to entice recipients into opening the attachment, triggering a multi-stage infection chain that ultimately installs remote management software on the victim's system.

Organizations in the financial sector should be aware that the campaign leverages trusted messaging platforms and document-themed lures to encourage user interaction. The activity could affect employees who exchange business documents through collaboration applications, potentially leading to unauthorized remote access and further malicious activity on compromised systems.

Technical Details

- The campaign is delivered through WhatsApp messages containing malicious VBScript attachments. In observed cases, compromised WhatsApp accounts were used to send the files to contacts.
- The attackers use file names that resemble legitimate business and financial documents, including invoices, account statements, payment records, and debt-related documents.
- The infection begins when a user downloads and opens the attachment from either WhatsApp Desktop or WhatsApp Web. This action launches the script through Windows Script Host.
- The initial VBScript creates a working directory and downloads additional VBScript components required for the next stages of the attack.
- The script uses multiple obfuscation techniques, including encoded content, randomized variable names, string reconstruction, and large amounts of unnecessary content to hinder analysis.
- Some variants disguise legitimate Windows utilities by renaming them and then use them to retrieve additional payloads from attacker-controlled infrastructure.
- The first secondary script attempts to modify User Account Control settings, repeatedly requesting elevated privileges to reduce security prompts for administrative actions.
- The second secondary script downloads a compressed archive, extracts its contents, and launches another script to continue the installation process.
- The attackers use several download methods to improve reliability, allowing the infection chain to continue even if one download mechanism fails.
- The final objective of the infection chain is the installation of legitimate Remote Monitoring and Management software, providing the attacker with remote access capabilities on the compromised system.

Recommendations

- Educate users to treat unexpected document attachments received through messaging platforms with the same caution as email attachments.
- Restrict or monitor the execution of VBScript files and Windows Script Host on endpoints where not required.
- Monitor for unusual downloads, script execution activity, and unexpected installation of remote management tools.
- Review security controls for collaboration and messaging platforms to identify suspicious file-sharing activity.
- Investigate reports of compromised messaging accounts and promptly reset credentials and active sessions when unauthorized access is suspected.

For a detailed breakdown of tactics, techniques, and procedures (TTPs), refer to the Annexure – VBScript Campaign Deploys Remote Management Software Via WhatsApp for Access

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Payouts King Ransomware IAB Affiliate Abusing Microsoft Edge to Deploy New Edgecution Malware	HIGH	CLEAR	Campaign	Open Source

Executive Summary

Researchers at a security research team has identified a campaign linked to an Initial Access Broker (IAB) associated with ransomware activity that deploys a new malware framework called 'Edgecution'. The attack uses social engineering through Microsoft Teams messages and a fake update website to deliver a malicious Microsoft Edge extension and a Python-based backdoor that bypass browser sandbox restrictions.

Organizations in the financial sector should be aware that the campaign combines trusted collaboration platforms, deceptive software update themes, and legitimate browser functionality to gain a foothold on victim systems. The activity could affect users who interact with fraudulent update requests, enabling unauthorized access, command execution, and further compromise of enterprise environments.

Technical Details

- The attack begins with Microsoft Teams messages that impersonate internal IT staff and instruct users to install what appears to be a required update.
- Victims are directed to a fake update website that provides several download and execution options, including scripts and files designed to deploy the malware.
- When executed, these scripts prepare the environment, extract embedded files, and create scheduled tasks that launch Microsoft Edge automatically.
- The deployment package contains two key components: a malicious Microsoft Edge extension and a Python-based backdoor used to perform host-level actions.
- The attackers abuse the Chrome native messaging protocol, allowing the browser extension to communicate with software running outside the browser sandbox.
- A native messaging configuration is created so the extension can launch and exchange commands with the Python backdoor.
- A scheduled task starts Microsoft Edge in headless mode and loads the extension silently, making the malicious activity invisible to the user.
- The extension connects to command-and-control infrastructure and maintains communication through periodic heartbeat messages.

- Commands received from the attackers are relayed to the Python backdoor, which can collect system information, access files, execute commands, run PowerShell code, and manage processes.
- The malware establishes a persistent foothold that enables the threat actor to conduct follow-on activities and support broader ransomware intrusion operations.

Recommendations

- Train users to verify unexpected IT support requests and software update instructions received through collaboration platforms.
- Monitor Microsoft Teams-related social engineering attempts and investigate suspicious update-themed communications.
- Review and restrict unauthorized browser extensions and monitor for extensions loaded outside standard enterprise deployment processes.
- Monitor for abnormal use of browser native messaging functionality, scheduled task creation, and headless browser execution.
- Investigate systems exhibiting unexpected command execution, PowerShell activity, or browser processes communicating with unknown external infrastructure.

For a detailed breakdown of tactics, techniques, and procedures (TTPs), refer to the Annexure – Payouts King Ransomware IAB Affiliate Abusing Microsoft Edge to Deploy New Edgecution Malware

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
DPRK-Linked macOS Gaslight Combines Credential Theft with LLM Triage Evasion	HIGH	CLEAR	Campaign	Open Source

Executive Summary

"macOS.Gaslight" is a Rust-based macOS backdoor leveraging multiple attack vectors, including initial compromise of macOS hosts, Telegram-based Command and Control, AES-GCM (Advanced Encryption Standard in Galois/Counter Mode) encrypted and certificate-pinned communications, interactive shell access, and data exfiltration via an embedded Python stealer targeting browsers and system artifacts. It uniquely incorporates prompt-injection payloads to disrupt AI-assisted analysis, alongside anti-analysis techniques like token self-redaction to evade detection and investigation.

Organizations in the financial sector should be aware that the malware provides remote access, file exfiltration, and extensive data collection capabilities on compromised macOS systems. The activity could affect environments that rely on macOS devices for business operations, particularly where browser-stored credentials, user data, and sensitive system information are present.

Technical Details

- The malware is a Rust-based macOS implant and infostealer that contains embedded prompt-injection content designed to interfere with LLM-assisted malware analysis workflows.
- The embedded prompt-injection content consists of numerous fabricated system messages intended to mislead automated analysis processes and discourage accurate investigation.
- Once active, the malware establishes command-and-control communications through a Telegram Bot API polling mechanism. This allows attackers to manage infected systems and receive collected data.
- The malware uses encrypted communications based on AES-GCM and applies certificate pinning to make network inspection and traffic monitoring more difficult.
- The command-and-control channel can operate through enterprise proxy environments, helping the malware maintain connectivity in managed corporate networks.
- The implant supports an interactive command interface that enables attackers to execute shell commands, terminate processes, upload files, and control operations on the compromised device.
- To maintain long-running access, the malware creates power-management assertions that prevent the system from entering sleep mode during operations.
- The malware receives configuration settings at runtime, indicating that operators can customize the deployment and behavior of the tool.
- A built-in Python-based data collection component gathers information from multiple browsers, including Chrome, Brave, Firefox, and Safari, along with command histories, installed applications, running processes, and system details.
- Collected information is exfiltrated through the same Telegram-based communication channel, providing attackers with continuous access to harvested data.

Recommendations

- Monitor macOS systems for unauthorized Telegram-related communications and unusual background processes.
- Review endpoint telemetry for unexpected shell execution, process termination activity, and file collection behavior.
- Implement multi-factor authentication to reduce the impact of stolen credentials from browser-based sources.
- Continuously monitor macOS endpoints for suspicious data access involving browsers, command histories, and system information.
- Ensure security teams validate findings through multiple analysis methods and do not rely solely on automated LLM-assisted triage workflows.

For a detailed breakdown of tactics, techniques, and procedures (TTPs), refer to the Annexure - DPRK-Linked macOS Gaslight Combines Credential Theft with LLM Triage Evasion

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
StealC and Amadey Infostealer Ecosystem Enables Credential Theft	HIGH	CLEAR	Campaign	Open Source

Executive Summary

Researchers at Microsoft has identified the continued use of the StealC infostealer and Amadey loader within a cybercrime ecosystem that specializes in credential theft, token theft, and malware delivery. StealC is designed to collect sensitive information from browsers, email clients, messaging applications, cryptocurrency wallets, and other applications, while Amadey acts as a delivery platform that deploys StealC and additional malware onto compromised systems.

Organizations in the financial sector should be aware that stolen credentials, session cookies, and authentication tokens could affect enterprise environments even when the initial infection occurs outside managed corporate systems. The activity may impact organizations that rely on cloud services, VPN access, and single sign-on platforms, where compromised credentials can be leveraged for unauthorized access and follow-on attacks.

Technical Details

- The attack chain often begins with social engineering techniques such as phishing emails, deceptive websites, malicious advertisements, or software downloads designed to trick users into executing malware.
- Amadey is commonly used as an initial loader that establishes a foothold on the victim's device and communicates with attacker-controlled infrastructure for further instructions.
- After gaining access, Amadey can download and execute additional payloads, including the StealC infostealer and other malicious tools.
- Once launched, StealC profiles the infected system and gathers information about the device, operating system, installed software, hardware, and running processes.
- The malware targets browser-stored data, including saved usernames, passwords, session cookies, autofill information, browsing history, and stored payment information.
- StealC also collects data from cryptocurrency wallets, email applications, messaging platforms, and file transfer tools, expanding the amount of information available to the attackers.
- The malware uses process injection techniques to access protected browser data and recover credentials before preparing the information for exfiltration.
- In addition to credential theft, StealC can collect files that match attacker-defined rules and capture screenshots from the victim's system.
- All collected information is transmitted to command-and-control infrastructure, where operators can review and monetize the stolen data.
- After completing data theft, StealC can download and execute additional malware, allowing attackers to expand access or conduct further malicious activity.

Recommendations

- Enforce multi-factor authentication and monitor for suspicious sign-in attempts involving valid credentials.
- Monitor endpoints for credential dumping, data collection activity, and unexpected browser access operations.
- Review access to cloud services, VPN platforms, and email systems for signs of unauthorized authentication.
- Restrict execution of untrusted software and monitor for loaders capable of downloading additional payloads.
- Continuously monitor for unusual outbound communications and investigate indicators of credential theft or token abuse.

For a detailed breakdown of tactics, techniques, and procedures (TTPs), refer to the Annexure - StealC and Amadey Infostealer Ecosystem Enables Credential Theft

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
New Mistic Backdoor Linked to ModeloRAT Activity and Ransomware Access	MEDIUM	CLEAR	Campaign	Open Source

Executive Summary

Researchers has identified a new backdoor named 'Mistic' that has been used in cybercrime intrusions since April 2026 and may be linked to an Initial Access Broker (IAB) associated with the ModeloRAT toolkit. The malware is deployed through DLL sideloading techniques, executes payloads directly in memory, and incorporates stealth features designed to maintain low-visibility access within victim environments.

Organizations in the financial sector should be aware that Mistic has been observed alongside remote access tooling linked to ransomware-related access operations. The activity could affect organizations through unauthorized remote access, credential theft, and follow-on intrusions, particularly where attackers are seeking to establish persistent access for later monetization or deployment of additional threats.

Technical Details

- Mistic is a newly identified backdoor that has been observed in intrusions since April 2026 and is suspected to be associated with an initial access broker activity cluster.
- In one observed intrusion, Mistic was deployed alongside ModeloRAT, a remote access trojan that has previously been linked to ransomware-related operations.

- The malware is delivered through DLL sideloading, allowing malicious components to be loaded by a legitimate executable and helping the activity blend into normal system operations.
- The attack uses a malicious DLL named to resemble legitimate endpoint security software, increasing the likelihood that the malicious activity remains unnoticed.
- During execution, the malware loads additional components, including a credential-stealing module that displays a fake login screen to capture user credentials.
- Mistic provides core backdoor functionality, including file upload and download capabilities, file management operations, and remote command execution.
- The backdoor can execute payloads directly in memory without writing files to disk, reducing forensic evidence and making detection more difficult.
- Operators can modify how frequently the malware communicates with command-and-control infrastructure, allowing flexible control over infected systems.
- The malware includes a built-in kill switch that enables it to terminate and remove itself from the compromised device when instructed.
- Observed attacks also leveraged legitimate administrative tools and scripting capabilities for reconnaissance, lateral movement, payload delivery, and credential collection activities.

Recommendations

- Monitor for DLL sideloading activity involving legitimate applications loading unexpected or untrusted libraries.
- Investigate systems exhibiting in-memory execution patterns, unusual process behavior, or unauthorized command execution.
- Deploy multi-factor authentication and monitor for signs of credential theft and suspicious login activity.
- Review endpoint logs for the use of administrative tools and scripting engines that deviate from normal operational patterns.
- Continuously monitor for persistence mechanisms, reconnaissance activity, and unauthorized communications that may indicate initial access broker activity.

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Miasma Campaign Expands Across Package and Build Workflows	MEDIUM	CLEAR	Campaign	Open Source

Executive Summary

Researchers at Socket has identified a new wave of activity associated with the Miasma, Mini Shai-Hulud, and Hades malware family that targets software supply chains through compromised npm packages, GitHub Actions workflows, and developer environments. The campaign abuses package installation processes and developer tooling to execute malicious code, steal secrets, and spread across trusted software development ecosystems.

Organizations in the financial sector should be aware that the campaign targets software development and CI/CD environments where sensitive credentials, access tokens, and build secrets are often stored. The activity could affect organizations that develop, maintain, or deploy applications using open-source packages and automated build workflows, potentially exposing development infrastructure and trusted software supply chains.

Technical Details

- The campaign targets software supply chains by distributing malicious npm package releases and compromising developer-related workflows. The activity also expanded beyond npm into a compromised Go ecosystem project.
- Attackers publish compromised package versions that appear legitimate but contain hidden mechanisms designed to execute malicious code during installation.
- The campaign uses a technique referred to as the “Phantom Gyp” execution pattern, allowing code execution during package installation without relying on obvious installation scripts.
- Malicious packages introduce a binding[.]gyp file that causes ‘node-gyp’ to run during the build process, creating an unexpected execution path for the attackers.
- During installation, the trigger launches a modified JavaScript loader that replaces normal package functionality with malicious behavior.
- The malware uses multiple layers of obfuscation and encryption to hide its functionality and hinder analysis.
- Staged payloads are deployed through JavaScript components and developer tooling, enabling further execution within trusted development environments.
- The campaign targets GitHub Actions workflows and seeks to steal secrets stored within CI/CD environments and automated build pipelines.
- Collected credentials and secrets are used to compromise additional repositories, packages, and developer accounts, enabling the campaign to spread further across trusted ecosystems.
- The overall objective is to gain access to developer environments, harvest credentials, and leverage trusted software distribution channels to extend the reach of the campaign.

Recommendations

- Review development environments for recently installed packages and validate package integrity before deployment.
- Audit CI/CD platforms and GitHub Actions workflows for unauthorized changes, suspicious executions, and exposed secrets.
- Rotate credentials, tokens, and secrets that may have been exposed through development or build environments.
- Restrict access to software repositories and enforce strong authentication controls for developer accounts.
- Monitor package installation activity and investigate unexpected code execution during build and dependency installation processes.

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Shai Hulud Campaign Bridging CI Systems and Cloud Environment	MEDIUM	CLEAR	Campaign	Open Source

Executive Summary

Researchers at FortiGuard Labs has identified activity linked to the Shai Hulud supply chain campaign, in which compromised build dependencies and CI/CD environments were leveraged to harvest credentials and gain access to cloud infrastructure. In the observed incident, attackers obtained credentials from a Jenkins environment, abused cloud identities, escalated privileges, and accessed cloud-hosted data services.

Organizations in the financial sector should be aware that the campaign demonstrates how compromised development and build environments can become an entry point into production cloud infrastructure. The activity could affect organizations that rely on CI/CD platforms, cloud-hosted databases, and automated deployment pipelines where privileged credentials and access tokens are available.

Technical Details

- The campaign is associated with the Shai Hulud supply chain threat, which targets software development environments through malicious packages that execute during installations or CI/CD jobs.
- The malware harvests credentials from build environments, including cloud credentials, package registry tokens, source-code repository tokens, SSH keys, and other sensitive secrets.
- Stolen credentials enable attackers to access cloud environments and expand their control beyond the affected development infrastructure.

- In the investigated incident, attackers gained access to a Jenkins runner environment and obtained cloud identity credentials associated with the system.
- The compromised cloud identity was later used from external systems, allowing attackers to authenticate to the cloud environment and begin reconnaissance activities.
- After gaining access, the attackers created a new cloud account with administrator privileges and generated additional access credentials to strengthen their foothold.
- The attackers then modified cloud infrastructure settings, including security configurations and database-related controls, to expand access to targeted resources.
- Extensive enumeration of cloud services, storage resources, secrets, databases, and identity components were observed as the attackers searched for valuable information.
- Access to cloud-hosted secrets enabled the attackers to obtain database credentials and interact with targeted data warehouse resources.
- The final stage involved repeated database interactions and the use of cloud storage permissions that facilitated data collection and exfiltration from the cloud environment.

Recommendations

- Monitor CI/CD environments for unauthorized package execution, credential access, and unusual build activity.
- Restrict cloud permissions assigned to development and build systems using least-privilege principles.
- Continuously monitor for abnormal use of cloud identities, especially access originating from unexpected locations or systems.
- Audit cloud accounts, secrets management platforms, and database access for unauthorized modifications or privilege escalation attempts.
- Rotate exposed credentials and review CI/CD pipelines for indicators of compromise following any suspected supply chain incident.

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
The Gentlemen RaaS Expands Operations with Evolving Tactics and Custom Tools	MEDIUM	CLEAR	Campaign	Open Source

Executive Summary

Researchers has identified new activity associated with The Gentlemen ransomware-as-a-service (RaaS) operation, including the use of custom tools, enhanced reconnaissance methods, and a Go-based backdoor

deployed ahead of ransomware attacks. The group gains access through exposed internet-facing services and compromised credentials, then performs extensive network discovery and lateral movement before deploying ransomware.

Organizations in the financial sector should be aware that The Gentlemen's operations emphasize long-term access, internal reconnaissance, credential abuse, and security-control evasion before ransomware deployment. The activity could affect organizations with exposed remote access infrastructure, weak credential security, or insufficient monitoring of administrative tools and network discovery activity.

Technical Details

- The group commonly gains initial access by exploiting vulnerabilities in internet-facing systems and using stolen, leaked, weak, or default credentials. Evidence also suggests collaboration with initial access brokers in some intrusions.
- After gaining access, the attackers conduct extensive reconnaissance to map the environment, identify domain resources, and understand network architecture.
- The group uses Active Directory discovery and network scanning tools to identify systems, services, and potential paths for further compromise.
- Network packet capture is performed to collect information about internal communications, potentially exposing credentials and other sensitive data that can assist later stages of the attack.
- For lateral movement, the attackers distribute ransomware across connected systems using Group Policy mechanisms and administrative network shares.
- The group also leverages remote administration utilities to execute ransomware on targeted endpoints when other propagation methods are not suitable.
- Prior to ransomware deployment, the attackers attempt to disable security products using multiple techniques, including vulnerable drivers and specialized tools designed to interfere with endpoint protection solutions.
- Researchers observed a custom Go-based backdoor that establishes persistent communication with command-and-control infrastructure and supports remote command execution.
- The backdoor gathers host information executes attacker commands and can establish proxy capabilities that help operators move deeper into the environment and expand reconnaissance efforts.
- Once preparation is complete, the ransomware is deployed across the environment using automated mechanisms that disable defenses, distribute payloads, and trigger execution on multiple systems simultaneously.

Recommendations

- Prioritize patching and monitoring of internet-facing systems, including remote access services and security appliances.
- Enforce strong credential policies and multi-factor authentication for privileged and remote-access accounts.
- Monitor for unauthorized network discovery, packet capture activity, and unusual use of administrative tools.

- Detect and investigate attempts to disable endpoint protection products, modify security settings, or install vulnerable drivers.
- Review Group Policy changes, scheduled task creation, and lateral movement activity that could indicate ransomware propagation.

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Steganographic Loader Campaign Delivers Remcos Through Multi-Stage Execution	MEDIUM	CLEAR	Campaign	Open Source

Executive Summary

Researchers has identified a phishing campaign distributing a multi-stage loader that ultimately deploys the Remcos remote access trojan. The attack uses archive attachments containing disguised executables and relies on steganographic techniques, in-memory execution, and multiple loader stages to conceal malicious payloads and evade traditional detection mechanisms.

Organizations in the financial sector should be aware that the campaign combines phishing, credential theft, persistence mechanisms, and remote access capabilities within a fileless infection chain. The activity could affect users who interact with malicious attachments, potentially enabling unauthorized access, surveillance, credential compromise, and follow-on malicious activity.

Technical Details

- The campaign begins with phishing emails delivering archive attachments that contain a malicious executable disguised as a legitimate business-related document.
- When executed, the malware launches a decoy application to reduce user suspicion while malicious activity continues in the background.
- The initial executable contains hidden resources that are used to deliver the next stages of the infection chain. The attackers conceal payloads using steganographic-style techniques.
- A concealed bitmap object is used as a container for embedded malicious content, allowing the malware to hide additional payloads inside seemingly benign resources.
- The first-stage loader extracts and loads another malicious component directly into memory without writing it to disk, reducing forensic evidence.
- Additional loaders are then executed in memory, eventually delivering the final Remcos payload through a layered execution process.
- The malware establishes persistence by creating autorun mechanisms and storing components under randomized names to make detection more difficult.

- Before proceeding, the malware checks for sandbox and virtualized environments and includes techniques designed to bypass user account controls.
- Once active, Remcos monitors user activity, tracks foreground windows, records user interactions, and supports audio and webcam surveillance functions.
- The malware steals browser-stored credentials, collects victim information, stores the harvested data locally, and exfiltrates the information to attacker-controlled infrastructure. Researchers also observed related infrastructure delivering other malware families, suggesting a broader malware distribution operation.

Recommendations

- Strengthen phishing defenses and train users to identify suspicious archive attachments and unexpected financial-themed documents.
- Monitor systems for unusual in-memory execution, reflective loading behavior, and unauthorized PowerShell activity.
- Review endpoint telemetry for persistence mechanisms involving registry autorun entries and hidden executables.
- Enable multi-factor authentication to reduce the risk associated with stolen browser credentials.
- Investigate systems exhibiting credential theft behavior, user activity monitoring, or unexpected access to audio and webcam resources.

For the indicators of compromise (IOCs), refer to the attached CSV sheet.

Reference to the Source

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Pedit COW Linux Kernel Flaw Enables Local Privilege Escalation to Root	HIGH	CLEAR	Vulnerability	CSC

Executive Summary

Linux has disclosed CVE-2026-46331, a critical local privilege escalation vulnerability known as "Pedit COW" affecting the Traffic Control (tc) subsystem. The flaw exists in the ‘tcf_pedit_act()’ function of the ‘act_pedit’ module, where an improper implementation of the Copy-on-Write mechanism allows attackers with local access to corrupt shared page-cache memory and elevate privileges to root. A public proof-of-concept exploit was released shortly after disclosure, increasing the likelihood of exploitation.

Organizations in the financial sector should be aware that successful exploitation could affect Linux-based servers, application platforms, and supporting infrastructure by enabling unauthorized root-level access. Systems that permit unprivileged user namespaces or utilize the affected component may face increased risk until vendor-supplied patches are applied and affected hosts are rebooted.

Technical Details

- The vulnerability is tracked as CVE-2026-46331 and is a local privilege escalation flaw affecting the Linux kernel Traffic Control (tc) subsystem.
- The issue resides in the 'tcf_pedit_act()' function within the 'act_pedit' kernel module, which is responsible for packet editing operations.
- The root cause is an improper implementation of the Copy-on-Write (COW) mechanism during writable memory validation.
- Runtime-calculated packet edit offsets can extend beyond the intended private memory page and corrupt shared page-cache memory.
- An attacker can abuse the vulnerable functionality through the Linux Traffic Control framework after obtaining namespace-scoped 'CAP_NET_ADMIN' privileges via unprivileged user namespaces.
- Exploitation requires only local access to a vulnerable Linux system and does not require user interaction.
- The vulnerability allows an unprivileged local user to escalate privileges from a standard account to full root access.
- Unlike many privilege escalation vulnerabilities, the attack modifies only the in-memory cached version of privileged executables while leaving the original files on disk unchanged.
- Because on-disk files remain unmodified, traditional file integrity monitoring and checksum-based detection mechanisms may not identify malicious changes.
- A publicly available proof-of-concept exploit was released within 24 hours of CVE assignment, lowering the barrier to exploitation against unpatched systems.

Recommendations

- Apply vendor-supplied kernel updates immediately on all affected Linux systems.
- Reboot systems after patch installation to ensure the updated kernel is loaded and active.

Vulnerability and affected product details can be found *here*.

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
PostgreSQL High-Severity Vulnerabilities Allow Arbitrary Code Execution	HIGH	CLEAR	Vulnerability	CSC

Executive Summary

PostgreSQL has disclosed three high-severity vulnerabilities, tracked as CVE-2026-2004, CVE-2026-2005, and CVE-2026-2006, that could allow arbitrary code execution on affected database servers. The flaws affect the intarray extension, pgcrypto extension, and PostgreSQL text processing functions, enabling attackers to execute code with the privileges of the operating system account running PostgreSQL.

Organizations in the financial sector should be aware that these vulnerabilities may impact database platforms supporting critical applications, transaction processing systems, and data repositories. Successful exploitation could affect the confidentiality, integrity, and availability of database environments and may provide an attacker with additional access to the underlying host system if vulnerable versions remain exposed.

Technical Details

- CVE-2026-2004 is a high-severity vulnerability (CVSS 8.8) caused by insufficient validation of input types in the PostgreSQL intarray extension selectivity estimator.
- The vulnerability affects the intarray extension and allows an object creator to trigger arbitrary code execution within the PostgreSQL environment.
- Code executed through CVE-2026-2004 runs with the privileges of the operating system account used by the PostgreSQL service.
- CVE-2026-2005 is a high-severity heap buffer overflow vulnerability (CVSS 8.8) affecting the PostgreSQL pgcrypto extension.
- A malicious ciphertext provider can exploit the vulnerability to trigger arbitrary code execution on the affected database server.
- Any code resulting from exploitation of CVE-2026-2005 executes with the privileges of the operating system account running PostgreSQL.
- CVE-2026-2006 is a high-severity vulnerability (CVSS 8.8) caused by missing validation of multibyte character lengths during PostgreSQL text manipulation operations.
- Specially crafted database queries can trigger a buffer overrun condition within affected text processing functions.
- Successful exploitation of CVE-2026-2006 may result in arbitrary code execution as the PostgreSQL operating system user.

Recommendations

- Upgrade PostgreSQL deployments to the fixed versions: 18.2, 17.8, 16.12, 15.16, or 14.21, as applicable.
- Upgrade BIG-IP Next for Kubernetes environments to version 2.2.2 or later.

Vulnerability and affected product details can be found *here*.

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Critical Google Gemini CLI Vulnerability Enables RCE in CI/CD Workflows	HIGH	CLEAR	Vulnerability	CSC

Executive Summary

Google has disclosed CVE-2026-12537, a critical remote code execution vulnerability affecting Google Gemini CLI and the 'run-gemini-cli' GitHub Action. The vulnerability stems from improper handling of malicious environment files and can allow attackers to execute arbitrary operating system commands during automated CI/CD workflow execution when a specially crafted pull request is processed.

Organizations in the financial sector should be aware that this vulnerability may impact software development and deployment environments that rely on automated CI/CD pipelines. Successful exploitation could affect the confidentiality and integrity of source code repositories, deployment processes, and sensitive credentials stored within build environments, potentially enabling broader compromise of connected systems.

Technical Details

- CVE-2026-12537 is a critical OS command injection vulnerability that can lead to remote code execution within automated CI/CD environments.
- The root issue stems from improper handling of malicious environment files during workflow execution.
- An attacker can exploit the vulnerability by submitting a specially crafted pull request containing a malicious environment file.
- The attack does not require authentication when targeting vulnerable automated workflows that process untrusted pull request content.
- Successful exploitation provides attackers with the ability to execute commands in the context of the affected workflow runner.
- Attackers may use the access to obtain sensitive information stored in the build environment, including secrets and credentials.
- The vulnerability may also enable modification of source code, manipulation of automated build processes, and compromise of software deployment workflows.
- Affected organizations could face further risk of lateral movement from compromised CI/CD infrastructure into connected development, testing, or production environments.

Recommendations

- Immediately upgrade Google Gemini CLI to version 0.39.1 or later, including supported newer releases.
- Upgrade 'run-gemini-cli' GitHub Action deployments to version 0.1.22 or later.

Vulnerability and affected product details can be found *here*.

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Multiple Jenkins Plugin Vulnerabilities Enable RCE, Security Bypass, and Credential Exposure	HIGH	CLEAR	Vulnerability	CSC

Executive Summary

Jenkins has disclosed multiple vulnerabilities affecting several widely used plugins, including high-severity flaws that could lead to remote code execution, sandbox bypass, path traversal, and XML External Entity (XXE) attacks. The vulnerabilities impact plugins used for script execution, source code management, build orchestration, security integrations, and workspace management, potentially allowing attackers to bypass security controls or execute malicious actions within Jenkins environments.

Organizations in the financial sector should be aware that these vulnerabilities may impact CI/CD infrastructure, software development pipelines, and automated deployment environments. Successful exploitation could affect the confidentiality of credentials, the integrity of build processes, and the security of Jenkins controllers and agents. The risk is elevated for organizations using affected plugins that currently have no available security fixes.

Technical Details

- Two high-severity vulnerabilities, CVE-2026-57280 and CVE-2026-57281, affect the Script Security Plugin and allow sandbox or script security bypass, potentially enabling execution of unauthorized code.
- CVE-2026-57296 affects the External Workspace Manager Plugin and allows path traversal, which could enable unauthorized access to files outside intended directories.
- CVE-2026-57301 affects the OWASP ZAP Plugin and may allow builds executed on the Jenkins controller to result in remote code execution.
- CVE-2026-57303 is an XML External Entity (XXE) vulnerability in the Assembla Plugin that may allow attackers to access unintended resources or sensitive information.
- CVE-2026-57282 affects the Git Client Plugin and allows OS command injection on Jenkins agents under specific conditions.
- Multiple plugins contain cross-site request forgery vulnerabilities, including Pipeline: Groovy Plugin, Priority Sorter Plugin, Gitee Plugin, EC2 Fleet Plugin, Contrast Continuous Application Security Plugin, Assembla Plugin, and Zowe zDevOps Plugin.
- Several vulnerabilities stem from missing or incorrect permission checks, allowing unauthorized users to enumerate SCM branches, server URLs, credentials identifiers, replay scripts, metadata, and other sensitive information.
- CVE-2026-57287 affects the Job Configuration History Plugin and may expose encrypted secret values stored in job and agent configurations.
- CVE-2026-57302 affects the FitNesse Plugin and stores passwords in plain text, increasing the risk of credential disclosure.

- CVE-2026-57288 affects the Active Directory Plugin and could allow LDAP injection through insufficient input handling.

Recommendations

- Upgrade all affected Jenkins plugins to the latest fixed versions where updates are available.
- Prioritize remediation of high-severity vulnerabilities affecting the Script Security Plugin, External Workspace Manager Plugin, Git Client Plugin, and other internet-exposed Jenkins components.
- Remove, disable, or restrict use of plugins that currently have no available fixes, including Assembla Plugin, FitNesse Plugin, OWASP ZAP Plugin, and Zowe zDevOps Plugin.

Vulnerability and affected product details can be found *here*.

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Critical Remote Code Execution Vulnerability in libssh2	**HIGH**	**CLEAR**	**Vulnerability**	**CSC**

Executive Summary

libssh2 (a library that implements the SSH2 protocol) has disclosed CVE-2026-55200, a critical vulnerability affecting libssh2 version 1.11.1 and earlier. The flaw exists in the 'ssh2_transport_read()' function, where insufficient validation of the 'packet_length' field can allow a specially crafted SSH packet to trigger an out-of-bounds write condition, leading to heap memory corruption.

Organizations in the financial sector should be aware that successful exploitation may impact systems and applications relying on vulnerable libssh2 implementations for SSH communications. The vulnerability could affect the confidentiality, integrity, and availability of affected services, and in certain circumstances may enable remote code execution on vulnerable hosts if exploited by a remote attacker.

Technical Details

- CVE-2026-55200 is a critical vulnerability with a CVSS v4 score of 9.2 affecting libssh2 versions 1.11.1 and earlier.
- The vulnerability exists within the 'ssh2_transport_read()' function, which processes incoming SSH transport packets.
- The root cause is insufficient validation of the SSH 'packet_length' field before memory operations are performed.
- A remote attacker can send a specially crafted SSH packet containing an excessively large packet length value.
- The malformed packet can trigger an out-of-bounds write condition during packet processing.
- Successful exploitation results in heap memory corruption within the affected libssh2 process.
- Memory corruption may cause application instability, crashes, or unexpected program behavior.

- Under certain conditions, the memory corruption may be leveraged to achieve remote code execution on the affected system.
- Exploitation can be performed remotely through SSH communications and does not require local access to the target system.

Recommendations

- Apply the latest security updates for libssh2 and affected applications.
- Identify and remediate vulnerable instances across the environment.
- Monitor systems for abnormal SSH-related activity or application crashes.
- Maintain an inventory of third-party libraries and dependencies to facilitate timely patching.
- Follow vendor security advisories and update guidance.

Vulnerability and affected product details can be found *here*.

back to top

Name	Threat Severity Rating	TLP	Attribution	Originating Source
Polymarket Supply-Chain Attack Leads to $3 Million Loss Through Malicious Frontend Script	**HIGH**	**CLEAR**	**Cyber Breach**	**Open Source**

Executive Summary

Polymarket, a cryptocurrency-based prediction market platform, disclosed a supply-chain attack in which a compromised third-party vendor injected a malicious script into the platform's frontend. The attack tricked users into approving fraudulent transactions through the legitimate website, resulting in the theft of approximately $3 million from a limited number of customer accounts. Polymarket stated that its backend systems and servers were not affected, and that the malicious dependency has been removed.

Organizations in the financial sector should be aware that this incident demonstrates how compromises involving trusted third-party components can be leveraged to target users without breaching core infrastructure. The activity may impact organizations that rely on third-party web components, digital asset platforms, or browser-based transaction workflows where malicious frontend content could facilitate credential theft or fraudulent transaction approval.

Technical Details

- The incident originated from a compromised third-party vendor that supplied content used by the platform's frontend environment. The attackers used this access to inject a malicious script into the legitimate website.
- Users interacted with the official platform interface and were presented with fraudulent transaction requests generated through the injected script.

- The attack relied on phishing-style transaction approval rather than exploitation of the platform's backend infrastructure or smart contract components.
- Once users approved the malicious requests, attackers gained access to digital assets stored in affected wallets.
- Independent blockchain analysis estimated that approximately $3 million was stolen during the incident.
- Analysis indicated that the stolen funds were converted and consolidated after being removed from victim accounts.
- The attackers moved the stolen assets between blockchain environments before converting them into another cryptocurrency asset.
- The incident impacted a relatively small number of accounts, with blockchain researchers estimating fewer than 15 affected wallets.
- Polymarket reported that its backend infrastructure and servers were not compromised during the attack. The compromise was limited to the frontend dependency.
- Following discovery of the incident, the affected dependency was removed and impacted users were notified regarding reimbursement.

Recommendations

- Review and continuously assess third-party dependencies used within customer-facing applications and web services.
- Implement monitoring and integrity controls to detect unauthorized changes to frontend code and external components.
- Apply strict approval validation processes for financial transactions and high-risk user actions.
- Monitor web applications for unexpected script behavior that could indicate supply-chain compromise.
- Conduct regular third-party security reviews and maintain visibility into vendor-managed components used in production environments.

References: Link 1 Link 2

back to top

Appendix A - Tactics, Techniques & Procedures (TTPs)

Multi-Stage LokiBot Infection Chain Observed in Recent Campaign

Tactics	Techniques	Observed Activity
Initial Access	Phishing Attachment	Delivery of malware via email attachment
Execution	Command and Scripting Interpreter	Script execution through Windows scripting engine
Execution	PowerShell	Execution of decoded secondary script
Defense Evasion	Obfuscated Files or Information	Use of heavily obfuscated script with nested functions
Defense Evasion	Reflective Code Loading	In-memory loading of .NET assembly
Privilege Escalation / Execution	Process Injection	Injection of payload into a spawned system process
Defense Evasion	API Obfuscation	Use of API hashing to resolve functions dynamically
Persistence	Registry Run Keys	Attempted persistence through registry modification
Discovery	System Information Gathering	Collection of system-related information prior to communication
Credential Access	Credential Dumping	Harvesting credentials from multiple applications
Command and Control	Application Layer Protocol	Periodic communication with remote server for instructions

Outlook Groups Abuse Enables Phishing Through Trusted Collaboration Workflows

Tactics	Techniques	Observed Activity
Initial Access	T1566 – Phishing	The research describes phishing activity that abuses Outlook Groups and Microsoft 365 collaboration features to move the victim toward a malicious action.
Initial Access	T1566.002 – Spearphishing Link	The final interaction can involve the user clicking through from group, shared-resource, or calendar-based content as part of the phishing workflow.
Initial Access	T1566.001 – Spearphishing Attachment	The research states that user actions such as opening or downloading content can lead to malware delivery or related compromise outcomes.
Execution	T1204 – User Execution	The attack depends on the user accepting, opening, signing in, downloading, or replying within what appears to be a normal collaboration workflow.
Credential Access	T1056 – Input Capture (high-level outcome mapping only)	The report states that user interaction may lead to credential theft, indicating credential-harvesting behavior driven by the phishing workflow.
Credential Access	T1528 – Steal Application Access Token	The research explicitly notes that the final action may result in token capture.
Collection	T1213 – Data from Information Repositories (high-level effect mapping)	The report states that successful interaction can lead to data exposure, indicating risk to information accessible through collaboration workflows

VBScript Campaign Deploys Remote Management Software Via WhatsApp for Access

Tactics	Techniques	Observed Activity
Initial Access	T1566.001 – Phishing Attachment	The actor distributed malicious VBScript files through direct messages in WhatsApp as attachments.

Initial Access	T1586 / T1078 – Compromised Accounts / Valid Accounts	The researchers concluded that the actor had gained access to several WhatsApp accounts and used those accounts to distribute the attachments to contacts.
Execution	T1204.002 – User Execution Malicious File	The campaign relied on recipients downloading and executing the VBScript attachment.
Defense Evasion	T1036 – Masquerading	The file names were designed to appear as legitimate business and financial documents, and the scripts contained comments and metadata mimicking Windows Update components.
Command and Control	T1219 – Remote Access Software	The infection chain ultimately resulted in installation of legitimate RMM software, enabling remote access to the victim system.

Payouts King Ransomware IAB Affiliate Abusing Microsoft Edge to Deploy New Edgecution Malware

Tactics	Techniques	Observed Activity
Initial Access	Phishing (T1566)	Social engineering via enterprise messaging impersonating IT staff
Execution	Command and Scripting Interpreter (T1059)	Use of AutoHotKey, PowerShell, and batch scripts to deploy malware
Persistence	Scheduled Task/Job (T1053)	Creation of scheduled tasks to launch hidden browser instance
Defense Evasion	Obfuscated Files or Information (T1027)	Encrypted payloads and obfuscated scripts to evade detection
Privilege Escalation / Execution	Native API (T1106)	Abuse of browser native messaging to execute system-level commands
Command and Control	Application Layer Protocol (T1071)	Communication via browser extension using structured messaging
Discovery	System Information Discovery (T1082)	Backdoor collects system and process information
Impact / Collection	Data from Local System (T1005)	Filesystem access and data retrieval via Python backdoor

DPRK-Linked macOS Gaslight Combines Credential Theft with LLM Triage Evasion

Tactics	Techniques	Observed Activity
Persistence	Boot or Logon Autostart Execution: Launch Agent T1547.015	The implant establishes persistence through a LaunchAgent and masquerades as an Apple system service.
Credential Access	Credentials from Password Stores T1555	The Python-based stealer harvests keychain material and browser-related data from the infected macOS system.
Collection	Data from Local System T1005	The malware collects browser data and terminal history from the local device.
Command and Control	Encrypted Channel T1573	Command-and-control uses AES-GCM-encrypted payloads over certificate-pinned TLS.
Execution	Command and Scripting Interpreter: Python T1059.006	The implant uses a runtime fetched standalone CPython interpreter to support the stealing component.
Defense Evasion	Masquerading T1036	Persistence is disguised as an Apple system service to blend into the operating environment.

StealC and Amadey Infostealer Ecosystem Enables Credential Theft

Tactics	Techniques	Observed Activity
Initial Access	Valid Accounts	Stolen credentials, SSO tokens, and session cookies may allow attackers to access enterprise services using legitimate authentication material.
Execution	User Execution	The research describes initial infection occurring outside managed endpoints, after which infostealer activity can create enterprise risk.
Persistence	Use Alternate Authentication Material	Stolen session cookies and tokens may allow continued access without needing the user's password.
Defense Evasion	Use Alternate Authentication Material	Stolen tokens and session cookies could allow attackers to bypass MFA protections.
Credential Access	Credentials from Password Stores	StealC collects sensitive browser data, including passwords, cookies, and session tokens.
Collection	Data from Local System	StealC collects data from browsers, cryptocurrency wallets, messaging applications, email clients, and gaming platforms.
C2	Ingress Tool Transfer	Amadey is used as a loader to deliver StealC and other malware.
Exfiltration	Exfiltration Over Command-and-Control Channel	Stolen passwords, cookies, and tokens are exfiltrated to attacker-controlled infrastructure.
Impact	Account Access Removal or Follow-on Intrusion Enablement	Stolen data can enter the access broker ecosystem and support ransomware or other follow-on operations.

Appendix B – Threat Severity Ratings & Definitions

In this newsletter, each campaign, breach, and vulnerability has been assigned a severity level—High, Medium, or Low—based on a contextual assessment of its impact, relevance to the financial sector, regional exposure, and exploit status.

A **High** severity rating is assigned to threats that involve widely used technologies, nation-state actors, or actively exploited zero-day vulnerabilities. These incidents pose a significant risk due to their potential for widespread disruption, data compromise, and strategic targeting of core infrastructure.

A **Medium** severity rating reflects threats that are technically impactful but have limited regional exposure or affect less prominent entities. These may include ransomware attacks on mid-sized firms, blockchain-based malware delivery mechanisms, or smart contract exploits in DeFi platforms.

A **Low** severity rating is applied to threats that are peripherally related to the financial sector, have no confirmed impact in the region, or target non-critical systems. These include credential stuffing attacks on platforms outside the MENA region or supply chain breaches that have not yet affected local infrastructure.

Threat Score Ratings & Definitions

1. **Severe**: Widespread, high-impact attacks ongoing. Critical infrastructure affected by destructive malware or multiple zero-day exploits.
2. **High**: Confirmed breaches or coordinated campaigns in progress. High-impact threat actor activity and zero-day exploitation observed.

3. **Elevated**: Active exploitation of known vulnerabilities or targeted campaigns detected. Some breaches or zero-days may surface.
4. **Guarded**: Slight increase in malicious activity. Low-level campaigns or threat actor reconnaissance may be underway.
5. **Normal**: No significant cyber activity beyond baseline. No major breaches, campaigns, or zero-day activity observed.

Appendix C – Traffic Light Protocol (TLP) Definitions and Usage

TLP	When should it be used?	How should it be shared?
TLP:Red	Sources may use TLP:RED when information cannot be effectively acted upon without significant risk for the privacy, reputation, or operations of the organizations involved. For the eyes and ears of individual recipients only, no further.	Recipients may not share TLP:RED information with any parties outside of the specific exchange, meeting, or conversation in which it was originally disclosed. In the context of a meeting, for example, TLP:RED information is limited to those present at the meeting. In most circumstances, TLP:RED should be exchanged verbally or in person.
TLP:Amber+Strict	Sources may use TLP:AMBER+STRICT when information requires support to be effectively acted upon, yet carries risk to privacy, reputation, or operations if shared outside of the organization.	Recipients may share TLP:AMBER+STRICT information only with members of their own organization on a need-to-know basis to protect their organization and prevent further harm.
TLP:Amber	Sources may use TLP:AMBER when information requires support to be effectively acted upon, yet carries risk to privacy, reputation, or operations if shared outside of the organizations involved. Note that TLP:AMBER+STRICT should be used to restrict sharing to the recipient organization only.	Recipients may share TLP:AMBER information with members of their own organization and its clients on a need-to-know basis to protect their organization and its clients and prevent further harm.
TLP:Green	Sources may use TLP:GREEN when information is useful to increase awareness within their wider community.	Recipients may share TLP:GREEN information with peers and partner organizations within their community, but not via publicly accessible channels. Unless otherwise specified, TLP:GREEN information may not be shared outside of the cybersecurity or cyber defense community.

TLP:Clear	Sources may use TLP:CLEAR when information carries minimal or no foreseeable risk of misuse, in accordance with applicable rules and procedures for public release.	Recipients may share this information without restriction. Information is subject to standard copyright rules.

Appendix D - Acronyms & Technical Terms

Term / Acronym	Meaning / Description
Access Broker	Criminal actor that obtains access to organizations and sells that access to other threat actors.
Active Directory	Microsoft's directory service used to manage users, devices, privileges, and access across enterprise networks.
Affiliate	A third party participating in a ransomware-as-a-service program using malware provided by the operator.
Amadey	Malware used to deliver other malicious payloads. It often serves as the initial stage in a larger attack.
Amazon Redshift	A cloud-based data warehouse platform used for large-scale analytics and reporting.
Arbitrary Code Execution	The ability for an attacker to run unauthorized code on a targeted system.
Aurora Database	A cloud-hosted relational database service referenced in the cloud intrusion activity.
AWS	Amazon Web Services, a cloud computing platform used for hosting applications and infrastructure.
Backdoor.Mistic	The specific Mistic malware family used to maintain stealthy persistent access.
Blockchain	A distributed ledger technology used to record cryptocurrency and digital asset transactions.
Bonzai	Malware family associated with the Gaslight activity cluster discussed in the newsletter.
Browser Cookie	Data stored by websites that can contain session information and authentication data.
Browser Credential Store	Location where browsers save usernames, passwords, cookies, and authentication data.
Browser Sandbox	A security mechanism that restricts browser processes from accessing sensitive system resources.
Build Artifact	A file or package generated during software development and deployment processes.
Build Environment	Systems and infrastructure used to compile, test, and deploy software.
Build Runner	A system that executes automated CI/CD jobs and workflows.
Business Email Compromise (BEC) Style Lure	Social engineering content designed to resemble legitimate business communications.
Certificate Pinning	A security technique that restricts trust to specific certificates during communications.
Cloud Account Abuse	Unauthorized use of legitimate cloud identities to access resources and services.
Cloud Identity	User, service, workload, or system account used to access cloud environments.
Cloud Infrastructure	Computing, storage, networking, and services hosted in cloud environments.
Cloud Misconfiguration	Security weakness caused by incorrect cloud settings, permissions, or controls.
Cloud Privilege Escalation	The process of gaining elevated permissions within a cloud environment.
Cloud Storage	Services used to store data in cloud environments.
Cobra/KongTuke	Publicly reported activity cluster associated with ModeloRAT and access-broker operations.
Collaboration Workflow Abuse	The misuse of trusted business communication and productivity tools for malicious purposes.

Compromised Dependency	A trusted software component that has been modified or abused by attackers.
Credential Exposure	The accidental or malicious disclosure of usernames, passwords, tokens, or authentication secrets.
Credential Harvesting	The collection of usernames, passwords, tokens, or other authentication information from victims.
Credential Reuse	Use of the same credentials across multiple systems, increasing attack risk if one account is compromised.
Cryptocurrency Wallet	Software or hardware used to store and manage digital assets.
Cryptojacking	Unauthorized use of systems to mine cryptocurrency.
CSC	UAE Cyber Security Council
Data Exfiltration	The unauthorized transfer of sensitive information outside an organization's environment.
Data Warehouse	A centralized environment used for analytics and large-scale data processing.
Dead-Drop Infrastructure	A technique where attackers use legitimate services or hidden locations to exchange instructions or payloads.
Dependency	A third-party software component or library relied upon by another application.
Developer Workflow	The sequence of tools and processes used during software development.
Development Pipeline	The tools and processes used to build, test, and release software.
Digital Asset	Electronic assets such as cryptocurrencies, tokens, or blockchain-based funds.
Domain Controller	A server responsible for authentication and access management within a Windows domain.
Edge Monitoring Agent	The disguise used by the Edgecution browser extension to appear legitimate.
Edgecution	A malware framework that abuses a Microsoft Edge extension and browser functionality to gain access to victim systems.
Ethereum (ETH)	A cryptocurrency used within blockchain ecosystems and observed in several incidents covered in the newsletter.
Exfiltration Staging	The preparation or collection of data before transmitting it outside an organization.
FileFix	Social engineering technique that tricks users into executing commands through Windows File Explorer.
Fileless Malware	Malware that operates primarily in memory to avoid traditional detection methods.
Frontend	The portion of a website or application that users directly interact with.
Frontend Dependency	Third-party code used by website interfaces that can expose users if compromised.
Frontend Supply-Chain Attack	A supply-chain attack where malicious code is introduced through website components presented to users.
Gaslight	A macOS malware family that combines credential theft with techniques intended to interfere with automated security analysis.
Git Client	Software used to interact with source code repositories managed by Git.
GitHub	A source code hosting and collaboration platform widely used for software development.
GitHub Repository	A source-code storage location used to manage software projects.
GitHub Token	An authentication credential used to access GitHub repositories and services.
Group Policy (GPO)	Microsoft mechanism used to centrally manage system settings and configurations across multiple devices.
Hades	Malware activity associated with broader software supply-chain compromise campaigns.
Helpdesk Impersonation	A tactic where attackers pretend to be technical support personnel to gain trust.
Helpdesk Lure	A social engineering technique where attackers impersonate support personnel to gain trust.

Identity Protection	Security controls focused on protecting user accounts, credentials, and access privileges.
Infrastructure Manipulation	Unauthorized modification of cloud, network, or system configurations.
Initial Access Broker (IAB)	A threat actor that specializes in obtaining and selling access to compromised organizations.
In-Memory Execution	Running malicious code directly in system memory without storing executable files on disk.
Jenkins	An automation platform commonly used for software builds, testing, and deployment.
Kernel	The core component of an operating system that manages hardware and system resources.
Keylogger	Malware that records keystrokes to capture credentials and sensitive information.
Lateral Movement	Techniques used by attackers to move between systems after initial compromise.
Living-off-the-Land (LotL)	The abuse of legitimate operating system tools to perform malicious activity while avoiding detection.
LokiBot	An information-stealing malware family focused on collecting credentials and sensitive user data.
Malicious Script	Unauthorized code inserted into a website or application to perform harmful actions.
Malvertising	The use of malicious advertisements to deliver malware or direct users to harmful content.
Malware-as-a-Service (MaaS)	A criminal model where malware is rented or sold to other attackers.
Managed Service Account	A specialized account used by services or applications to perform automated tasks.
Miasma	A software supply-chain malware campaign targeting developers, package repositories, and build environments.
Microsoft Teams	A collaboration and communication platform frequently targeted in social engineering attacks.
Mini Shai-Hulud	A malware campaign that spreads through compromised development ecosystems and software packages.
Mistic	A stealth-focused backdoor used to maintain unauthorized access and support follow-on attacks.
ModeloRAT	A remote access trojan used by attackers to gain persistent access to enterprise environments.
Monitoring Agent	Software used to collect operational or security information from systems.
Network Reconnaissance	The process of collecting information about systems, services, users, and infrastructure.
Open-Source Ecosystem	The community and repositories that distribute publicly available software components.
Open-Source Package	Reusable software code made publicly available and commonly incorporated into applications.
Outlook Groups	A Microsoft 365 collaboration feature that can be abused to deliver phishing content.
Package Poisoning	Malicious modification of software packages distributed through trusted repositories.
Package Repository	A centralized service used to store and distribute software packages and updates.
Payouts King	A ransomware-linked threat activity cluster associated with initial access operations.
Pedit COW	Name assigned to the Linux kernel privilege-escalation vulnerability CVE-2026-46331.
Persistence Mechanism	A technique used to ensure attacker access survives system reboots or user logouts.
Phantom Gyp	A package-installation execution method used in the Miasma campaign through binding.gyp files.
Pipeline Replay Script	A stored Jenkins workflow script that can be rerun or reviewed within CI/CD environments.
Plugin	A software component that adds features or functionality to an existing application.
Polymarket	Cryptocurrency-based prediction market platform impacted by a third-party supply-chain compromise.

PostgreSQL	An open-source relational database platform used in enterprise environments.
Prediction Market	A platform where users trade based on the expected outcomes of future events.
Prediction Market Platform	A service where users place trades based on the expected outcome of future events.
Privilege Escalation	The act of obtaining higher permissions than originally granted on a system.
Privileged Access	Elevated permissions that provide extensive control over systems or data.
Privileged Account	An account with administrative or elevated permissions.
Process Enumeration	The collection of information about running programs on a system.
Process Injection	A technique used to run malicious code within a legitimate process.
Proof-of-Concept (PoC) Exploit	Demonstration code showing that a vulnerability can be successfully exploited.
pUSD	A blockchain-based digital asset used within the affected prediction market platform.
Ransomware	Malware that encrypts data or disrupts systems and demands payment for recovery.
Remcos	Commercially available remote administration software that is frequently abused by threat actors for malicious purposes.
Remote Administration Tool (RAT)	Software that enables remote control of systems; often abused by attackers.
Remote Monitoring Tool	Software designed to manage systems remotely that can be misused for unauthorized access.
Rogue Extension	A malicious browser extension used to perform unauthorized actions.
Root Access	The highest level of privilege on Linux and Unix-based systems.
Secrets Enumeration	The process of identifying stored passwords, keys, tokens, or credentials.
Secrets Manager	A service used to securely store passwords, API keys, tokens, and other sensitive information.
Security Group	Cloud-based network access control mechanism that restricts communications between resources.
Service Account	A non-human account used by applications, services, or automated processes.
Session Hijacking	Unauthorized takeover of an authenticated user session.
Session Token	A digital credential used to maintain authenticated access to a service.
Shai Hulud	Malware campaign targeting software-development ecosystems and CI/CD environments.
Side Loading	Loading malicious code through trusted software execution paths.
Single Sign-On (SSO)	An authentication mechanism allowing one login to access multiple applications or services.
Social Engineering	The manipulation of users into performing actions that benefit attackers, such as opening files or revealing information.
Software Builder	A system or service responsible for compiling and packaging software.
Software Maintainer	Individual or organization responsible for updating and publishing software packages.
Software Supply Chain	The ecosystem of software developers, vendors, components, libraries, and deployment processes used to build applications.
Source Code Repository	Centralized storage for application source code and project files.
StealC	A malware-as-a-service infostealer designed to collect credentials, cookies, browser data, and sensitive information.
Steganography	The concealment of data inside another file or object, such as an image.
Supply-Chain Compromise	A breach affecting a trusted supplier, software dependency, or service provider rather than the primary target directly.
The Gentlemen	A ransomware-as-a-service operation observed using custom tools, backdoors, and ransomware deployment techniques.

Third-Party Vendor	An external company or service provider that supplies software, services, or infrastructure.
Threat Infrastructure	Systems, servers, domains, and services used to support malicious operations.
Token Capture	Theft of authentication tokens used to access systems or services.
Traffic Control (tc)	A Linux networking subsystem used for managing and modifying network traffic.
Trusted Workflow Abuse	Use of legitimate business processes to conceal malicious activity.
User Enumeration	Discovery of valid user accounts within an application, system, or organization.
VPN	Virtual Private Network. A technology that enables secure remote access to organizational networks.
Vulnerable Driver	Legitimate driver software containing weaknesses that can be abused by attackers.
Warehouse Credentials	Credentials used to access data warehouse environments.
WhatsApp Desktop	A desktop version of WhatsApp targeted in malware delivery campaigns.
Woodgnat	An initial access broker believed to sell or provide unauthorized access to victim organizations.
Workspace	A software development or execution environment used by developers and automated workflows.
Workspace Trust	A security mechanism that restricts execution of content from untrusted projects or repositories.
Worm	Malware capable of self-propagation across systems or environments.
Zero-Touch Deployment	Automated deployment process requiring little or no human interaction, often targeted by supply-chain attacks.