05
MAY

THREAT INTEL REPORT 2024

CONVERGE
TECHNOLOGY SOLUTIONS

Prepared by: Converge Cybersecurity Practice
convergetp.com/cybersecurity | 800.747.8585

Observations for May 2024

April 2024 saw a surge in severe cyber threats, shaking the very foundations of global cybersecurity. Critical vulnerabilities, like the zero-day flaw in Palo Alto Networks' GlobalProtect and the exploitation of GitHub and GitLab's networks, underscore a distressing escalation in cyber attacks. These incidents spotlight the relentless ingenuity of cyber adversaries, forcing organizations to urgently reassess and fortify their cybersecurity strategies.

In response to these daunting challenges, this edition introduces the "Threat Hunting Hypothesis" section. This new feature delves into the threats detailed in this report, offering CISOs and security professionals deeper insights and proactive strategies. By examining the patterns and methodologies of recent attacks, this section aims to enhance organizational resilience and readiness against the unpredictable dynamics of the cyber threat landscape to strengthen digital infrastructures and safeguarding sensitive information across various sectors.

RETURN

Executive Overview

Zero-day exploitation of Palo Alto Networks GlobalProtect

Audience

- Cybersecurity professionals and analysts
- C-suite executives of affected and potentially vulnerable companies
- Network administrators

Summary

On April 10, 2024, Volexity identified a critical zero-day exploitation in the GlobalProtect feature of Palo Alto Networks PAN-OS, tracked as CVE-2024-3400, leading to unauthorized remote code execution. Threat actor UTA0218 initiated the exploitation to install a Python-based backdoor dubbed UPSTYLE that enables further malicious activities within the compromised systems. The severity of this vulnerability, combined with its high CVSS score of 10.0, underscores the urgent need for affected organizations to apply patches and conduct thorough security reviews to mitigate potential threats.

Go to tactical guidance >>

Exploitation of GitHub and GitLab CDN features for malware distribution

Audience

- Cybersecurity professionals
- Developers
- System engineers

Summary

A critical vulnerability identified in the content delivery network (CDN) features of GitHub and GitLab allows threat actors to distribute malware under the guise of legitimate repository files. This exploitation leverages the trust in URLs associated with well-known repositories to deceive users into downloading malicious content, posing significant security risks to any organization using these platforms.

Go to tactical guidance >>

DEV#POPPER cyber threat campaign fakes interviews to target developers

Audience

- C-suite executives
- IT and cybersecurity teams
- Developers

Summary

The DEV#POPPER campaign orchestrated by suspected North Korean threat actors exploits the software development hiring process through fake job interviews. These interviews are a facade for delivering a Python-based remote access Trojan (RAT) that compromises developers' systems to steal sensitive information and gain remote access. This campaign highlights the

RETURN

critical need for heightened security awareness during recruitment interactions and the implementation of robust security measures within software development and recruitment processes.

Go to tactical guidance >>

Audience

- C-suite executives
- IT and cybersecurity teams

Emerging threat profile for CoralRaider's cyber espionage campaign

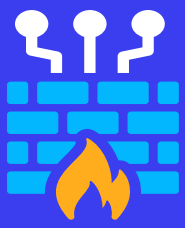
Summary

The cyber criminal group CoralRaider, believed to be based in Vietnam, has actively targeted a range of victims in Asia and Southeast Asia since at least 2023. Utilizing sophisticated malware tools like RotBot and XClient stealer, CoralRaider focuses on exfiltrating financial information, login credentials, and data from social media accounts. This summary explores the implications of CoralRaider's operations for regional security and the global cybersecurity landscape.

Go to tactical guidance >>

Tactical Guidance | Emerging Threats

UPSTYLE backdoor



on firewalls allows RCE and data exfiltration.

Zero-day exploitation of Palo Alto Networks GlobalProtect

Overview & Impact

CVE-2024-3400 is an OS command injection vulnerability within Palo Alto Networks' GlobalProtect feature that allows unauthenticated remote code execution. UTA0218 exploited this vulnerability to implant the UPSTYLE backdoor on firewall devices to facilitate unauthorized command execution and data exfiltration. The exploitation activities included downloading tools, exporting configuration data, and moving laterally within the network to access sensitive organizational resources. Palo Alto Networks issued a security advisory and a patch for the vulnerability on April 14, 2024. This incident highlights significant security risks and must prompt immediate action from organizations to prevent potential breaches and data theft.

Observations

- **Enhance asset management:** Implement robust asset management protocols to rapidly identify and classify devices affected by critical CVEs and zero-day vulnerabilities.
- **Log source review:** Conduct regular reviews of log sources to ensure that the logs critical to security monitoring are captured effectively. This includes enhancing log coverage from security tools and critical infrastructure to ensure all relevant data is available for analysis.

Guidance

Strategic Intelligence

- **Enhance threat intelligence sharing:** Foster relationships with industry partners and cybersecurity forums to exchange timely and actionable threat intelligence to improve to collective defenses.
- **Emergency patch plan:** Develop and maintain an emergency patch deployment plan to expedite the remediation of critical systems when vulnerabilities are identified to minimize exposure time.

Tactical Intelligence

- **Update detection rules:** Update detection rules to include IoCs provided by threat intelligence communities. If not connected to any, Volexity provides a list of IoCs here.
- **Network traffic analysis:** Implement advanced network monitoring tools to detect unusual traffic patterns or unauthorized data transfers that could indicate a breach.

RETURN

Threat Hunting Hypotheses

Malicious command execution via CVE-2024-3400

- **Hypothesis:** Threat actors leverage CVE-2024-3400 to execute arbitrary commands on compromised Palo Alto Networks GlobalProtect firewall devices.
- **Investigation approach:** Monitor firewall logs for unusual command executions correlating with the timestamps of known exploit attempts. Analyze network traffic to and from the firewall for anomalous patterns that could indicate command-and-control communications.

Persistence establishment post-exploitation

- **Hypothesis:** Following successful exploitation of CVE-2024-3400, attackers may establish persistence mechanisms to maintain access even after the initial entry vector is closed.
- **Investigation approach:** Review scheduled tasks, cron jobs, and startup scripts for unexpected entries. Verify all changes to system and configuration files post-exploit to identify unauthorized modifications.

Lateral movement within the network post-exploitation

- **Hypothesis:** Attackers exploiting CVE-2024-3400 may attempt lateral movement within the network to gain access to higher-value targets.
- **Investigation Approach:** Check internal traffic logs for unusual SMB or RDP activities originating from affected firewalls. Use EDR systems to track process executions and file movements that originate from or involve the compromised firewall.

Data exfiltration attempts

- **Hypothesis:** Threat actors exploiting CVE-2024-3400 might attempt to exfiltrate sensitive data from the network by leveraging the compromised firewall as an exfiltration point.
- **Investigation approach:** Monitor outbound data flows from the network and focus on large or irregular data transfers during off-peak hours. Investigate any use of encrypted channels that bypass normal network monitoring tools.

Indicator of compromise (IoC) detection

- **Hypothesis:** Systems affected by CVE-2024-3400 might show identifiable IoCs that can help detect breach attempts or successful exploitations.
- **Investigation Approach:** Utilize IoCs provided by threat intelligence communities, such as those available from Volexity, to scan network and endpoint systems. Regularly update IoC lists and ensure all monitoring tools are configured for detection.

Analysis of compromised system forensics

- **Hypothesis:** Compromised firewall devices may contain forensic evidence of the attack vectors used and the scope of the compromise.
- **Investigation approach:** Collect and analyze logs and volatile memory from compromised devices to identify malicious activities and artifacts related to CVE-2024-3400 exploitation.

RETURN

Sources

- Volexity Blog: Zero-Day Exploitation of Unauthenticated Remote Code Execution Vulnerability in GlobalProtect (CVE-2024-3400)
- Kroll Cyber Risk: CVE-2024-3400 Zero-Day Remote Code Execution Vulnerability
- Unit 42 by Palo Alto Networks: CVE-2024-3400 Analysis
- SOC Prime: CVE-2024-3400 Detection and Command Injection PAN-OS Zero-Day Vulnerability in GlobalProtect Software

Exploitation of URL trust

allows camouflaged malware to linger.

Exploitation of GitHub and GitLab CDN features for malware distribution

Overview & Impact

Threat actors are exploiting a feature in GitHub and GitLab that allows files to be uploaded via comments on commits and pull requests. These files are stored on the platforms' CDNs and given URLs that misleadingly associate them with the repositories in question. Even if a comment is never published or is subsequently deleted, the malicious URL remains active. This design flaw, or perhaps an oversight, provides an effective camouflage for malware distribution, exploiting people's inherent trust in repository-associated URLs. Every major software company using these services is vulnerable to this attack, which can undermine security measures based on URL trustworthiness and significantly impact the integrity and security of software supply chains.

Observations

- **Misuse of trusted URLs:** Malware is distributed through URLs appearing to belong to reputable repositories, increasing the likelihood of successful deception.
- **Persistence of malicious links:** Links to the malicious files remain active indefinitely, regardless of the visibility or status of the originating comment.
- **Lack of control over hosted files:** Repository administrators currently lack the ability to manage or remove maliciously uploaded files, leaving no straightforward remediation path.
- **Exploited platforms' features:** Both GitHub and GitLab are affected and no immediate fix available. This suggests a systemic vulnerability within the platforms that allows user-generated content to link to project files.

Guidance

Strategic Intelligence

- **Enhanced code review policies:** Organizations should implement stringent code review processes that specifically address security concerns arising from using public repositories like GitHub and GitLab.

RETURN

Verifying the integrity and origin of all code before integration is essential, especially for code that interacts with external repositories.

- **Adoption of secure code practices:** Develop and enforce policies requiring signed commits and tags to ensure code and contributors' authenticity. Utilizing cryptographic signatures can significantly reduce the risk of integrating compromised code.
- **Security training for developers:** Increase training programs focused on security best practices for developers. This training should include recognizing security vulnerabilities and understanding the risks associated with third-party code.

Operational Intelligence

- **Regular audits of repository use:** Regularly auditing repository use and integrating third-party libraries can help ensure compliance with security policies and reveal potential vulnerabilities or unauthorized changes early.
- **Monitoring public repository integrity:** Establish procedures for routine review of your publicly available repositories to identify unauthorized or malicious alterations. This includes checking for unexpected commits, branches, or files that could indicate misuse.
- **Vulnerability scanning of public repositories:** Implement automated tools to scan your public repositories for known vulnerabilities. These tools alert administrators to potentially malicious activity occurring within their hosted projects.

Tactical Intelligence

- **Enhanced threat intelligence feeds:** Utilize enhanced threat intelligence feeds tailored specifically to identifying threats targeting development platforms like GitHub and GitLab. This intel should include information on new attack vectors and vulnerabilities specific to these environments.
- **Security alert integration:** Ensure that all security systems are integrated to allow centralized monitoring and management of alerts related to repository activities. This integration helps correlate data across different security tools to identify complex attack patterns more effectively.

Threat Hunting Hypotheses

Malicious file uploads in repository comments

- **Hypothesis:** Threat actors may upload malicious files to GitHub or GitLab using the comment feature and creating links that appear legitimate and trustworthy.
- **Investigation approach:** Search for files uploaded in comments across all repositories. Analyze file types, origins, and access patterns. Check for unusual file formats or sizes that do not align with typical project activities.

RETURN

Anomalous commit patterns

- **Hypothesis:** An attacker may introduce malicious code into a repository through seemingly innocuous commits to exploit the trust in existing project structures.
- **Investigation approach:** Monitor commit logs for anomalies such as commits at odd hours, large commits by new or infrequent contributors, or commits that bypass established review processes.

Persistence mechanisms in repositories

- **Hypothesis:** Once access is gained, threat actors may attempt to establish persistence within a repository to ensure continued access even after initial exploits are discovered and remediated.
- **Investigation approach:** Review recent changes to repository configurations, webhook settings, or integrations that could facilitate persistent access. Validate changes with known project updates or authorized changes.

Use of repository credentials by unauthorized entities

- **Hypothesis:** Compromised credentials are used to perform unauthorized activities within GitHub or GitLab repositories.
- **Investigation approach:** Analyze authentication logs for access anomalies, such as logins from unusual locations or times, and validate actions performed during these sessions against expected user behaviors.

Abnormal file access or downloads from repositories

- **Hypothesis:** Malicious files within repositories are accessed or downloaded more frequently than typical project files, possibly indicating a malware distribution campaign.
- **Investigation approach:** Monitor file access logs to identify unusual download patterns or access rates, particularly for files hosted on CDN links. Compare against baseline activity for similar files.

Modification of repository files to include malicious payloads

- **Hypothesis:** Malicious actors might modify existing files in a repository to include harmful payloads, leveraging the trust in previously safe files.
- **Investigation approach:** Implement checksum or hash verification for files at regular intervals to detect unauthorized modifications. Use version control history to identify unauthorized changes.

Exploitation of third-party integrations

- **Hypothesis:** Attackers exploit third-party integrations configured in GitHub or GitLab to execute malicious activities or exfiltrate data.
- **Investigation approach:** Review all third-party integrations and OAuth tokens for anomalies. Check for unexpected external callbacks or data transmissions.

Sources

- BleepingComputer: GitHub Comments Abused To Push Malware via Microsoft Repo URLs

RETURN

- ITPro: Hackers Have Found Yet Another Way To Trick Devs Into Downloading Malware From GitHub
- Bitdefender Blog: GitHub Flaw Could Allow Threat Actors To Distribute Malware on GitLab
- Gridinsoft: GitHub and GitLab CDNs Abused for Malware

False job application tasks

use social engineering to deliver malware.

DEV#POPPER cyber threat campaign uses fake interviews to target software developers

Overview & Impact

The DEV#POPPER campaign, first identified by cybersecurity firm Securonix and further detailed by Palo Alto Networks' Unit 42 and Phylum, utilizes social engineering to target software developers. Developers are deceived into downloading malicious npm packages from GitHub as part of job application tasks. These packages contain malware, such as BeaverTail and InvisibleFerret, which enable data theft and remote system access.

This campaign reflects a sophisticated blending of social engineering and technical attack vectors that leverages developers' professional trust and engagement. The implications for organizations include potential breaches of sensitive data and the compromise of software supply chains, which can lead to broader security vulnerabilities across affected platforms.

Observations

- **Targeting technique:** Attackers pose as employers on freelance job portals to engage with software developers through fake job interviews.
- **Malware delivery:** Malicious npm packages are used as the primary delivery method, with the malware hidden in seemingly benign files downloaded during the interview process.
- **Attack complexity:** The attack involves a multi-stage infection chain that includes downloading a ZIP archive containing a malicious npm package which then downloads further payloads.
- **Operational security risk:** Once executed, the malware conducts system reconnaissance, file enumeration and exfiltration, and can perform command execution that poses significant risks to operational security.

Guidance

Strategic Intelligence

- **Enhance awareness:** Increase organizational awareness of recruitment-based cyber threats. Integrate these insights into security strategy and training.
- **Developers:** Reinforce the concept that developers are prime targets for cyber threats due to their access and capabilities. Regularly update

RETURN

development teams on new threats aimed specifically at their role to enhance their abilities to spot and avoid potential dangers.

- **Incident response plan:** Have a clear, well-communicated response plan for security incidents, including those from BYOD and organizational devices. This plan should include immediate steps for containing and mitigating damage, regardless of the device involved.

Operational Intelligence

- **Identity and access management (IAM):** Ensure that IAM tools are used and actively monitored for security purposes. Employ these tools to track and analyze developer activities so that response to suspicious behavior mitigates security threats promptly.

Tactical Intelligence

- **Implement user behavior analytics (UBA):** Use UBA solutions to monitor developer activities and detect anomalies indicative of potential security threats, such as unusual login times, unauthorized access attempts, or data access patterns deviating from the norm.

Threat Hunting Hypotheses

Execution of malicious npm packages

- **Hypothesis:** Threat actors may disguise malicious npm packages as legitimate software components within job application tasks to infiltrate developers' systems.
- **Investigation approach:** Monitor and analyze npm package installations for unusual activity, including packages from new or unknown repositories. Check for anomalies in package content and behavior post-installation.

Anomalous download and execution patterns

- **Hypothesis:** During fake job interviews, attackers could prompt the download and execution of files that trigger the deployment of further malicious payloads.
- **Investigation approach:** Log and review file download and execution activities on systems associated with software development. Identify patterns that diverge from normal user behavior, focusing on source, file type, and execution timing.

Remote access Trojans (RAT) activities

- **Hypothesis:** RATs installed during the interview process could be used for long-term access and data exfiltration from developers' machines.
- **Investigation approach:** Employ network monitoring tools to detect outbound connections to known malicious IP addresses or unusual geographic locations. Analyze traffic for patterns consistent with data exfiltration.

Obfuscated JavaScript files in npm packages

- **Hypothesis:** JavaScript files within npm packages may be obfuscated

RETURN

to hide malicious code to enable the execution of unauthorized commands without detection.

- **Investigation approach:** Implement automated tools to de-obfuscate and analyze JavaScript files within npm packages. Monitor for scripts that execute network requests or modify system configurations.

Persistence via downloaded payloads

- **Hypothesis:** Malware downloaded as part of the DEV#POPPER campaign may attempt to establish persistence on the host system for access even after initial detection.
- **Investigation approach:** Review system startup locations and scheduled tasks for new or modified entries post-interview. Check for scripts or binaries that match known malware signatures.

Command-and-control (C2) communication

- **Hypothesis:** Infected hosts may initiate C2 communications to receive further instructions or to exfiltrate sensitive information.
- **Investigation approach:** Analyze network traffic for recurring connections to the same external IP addresses or domains, especially those using encrypted channels. Use network intrusion detection systems to flag known malicious communication patterns.

Sources

- Securonix: Analysis of DEV#POPPER: New Attack Campaign Targeting Software Developers Likely Associated With North Korean Threat Actors
- Black Hat Ethical Hacking: New DEV#POPPER Campaign Lures Developers With Job Offers, Spreads Python RAT
- BleepingComputer: Fake Job Interviews Target Developers With New Python Backdoor
- The Hacker News: Bogus npm Packages Used To Trick Developers in New Cyber Espionage Campaign

Social media attacks

use sophisticated tactics to extract valuable data.

Emerging threat profile: CoralRaider cyber espionage campaign

Overview & Impact

CoralRaider has demonstrated significant capabilities in advanced persistent threats, employing customized tools such as RotBot, a variant of QuasarRAT, and XClient stealer to infiltrate and extract valuable data. The group primarily targets social media accounts and leverages dead-drop resiliency techniques and living-off-the-land binaries to maintain stealth.

Their operations have been traced back to Vietnam, with evidence suggesting a high degree of sophistication in their attack vectors, including using Telegram bots for command-and-control activities. The implications for affected organizations are severe, as breaches can lead to

RETURN

substantial financial losses and damage to reputational integrity.

Observations

- **Geographic focus:** CoralRaider predominantly targets entities in Asia and Southeast Asia, including economic powerhouses like India and China, suggesting strategic positioning to exploit regional economic dynamics.
- **Malware deployment:** Use of RotBot and XClient indicates a high level of customization in tools designed for specific espionage tasks.
- **Command-and-control tactics:** Utilization of Telegram for C2 infrastructure showcases an adaptation to secure, less-regulated communication channels.
- **Social media exploitation:** Focused attacks on social media accounts emphasize the modern threat landscape where business and advertising accounts are lucrative targets.
- **Self-compromise indication:** Instances of the threat actor accidentally compromising their systems during tests suggest operational risks and potential avenues for counterintelligence.

Guidance

Strategic Intelligence

- **CDN security policy:** Establish a dedicated policy governing the use of CDNs to ensure they are secure and efficiently managed to prevent data breaches.
- **Social media security policy:** Create specific policies to manage and secure the use of social media platforms to mitigate risks from malware dissemination and unauthorized data access.

Operational Intelligence

- **Enhanced monitoring:** Implement advanced monitoring tools to detect unusual network traffic and potential data exfiltration to unauthorized Telegram channels or other suspect endpoints.
- **Incident response and remediation:** Establish or update incident response protocols to address breaches involving advanced persistent threats swiftly to ensure minimal operational disruption and data loss.
- **Employee cybersecurity training:** Conduct regular employee training sessions on recognizing phishing attempts and other common tactics used by cyber criminals like CoralRaider.
- **Implementing proxy and full traffic decryption:** Utilize proxy servers and full traffic decryption measures to inspect and manage web traffic for enhanced visibility into data flows and potential threats.
- **CDN security enhancements:** Enhance the security of CDN usage by implementing strict access controls, regular audits, and encryption of data in transit to prevent unauthorized access and data leaks.

Tactical Intelligence

- **Enable system logging for DPAPI events:** Ensure comprehensive

RETURN

logging of DPAPI-related activities to monitor and detect unauthorized access to browser data. Enable Audit DPAPI Activity in system security settings to log event 4693 in the security log and ensure DPAPI/Debug logs are configured to capture events, particularly event ID 16385 which indicates data decryption attempts.

- **Integrate event log monitoring into SOC operations:** Equip SOC teams with the tools and knowledge to analyze DPAPI event logs effectively. Implement log management solutions to aggregate and analyze logs from security and DPAPI/Debug logs, and train SOC analysts to identify suspicious activities in DPAPI logs, especially unusual process IDs or data access patterns.
- **Develop custom detection logic for anomalous DPAPI access:** Automate detection of anomalies in DPAPI access that may signify malicious intent. Establish detection rules in SIEM systems to alert on discrepancies between expected and actual process IDs accessing secured browser data, and correlate DPAPI access logs with process creation logs (event ID 4688) to verify the legitimacy of processes attempting data decryption.

Threat Hunting Hypotheses

Anomaly detection in network traffic

- **Hypothesis:** Increased command-and-control traffic to unrecognized external IPs or domains could indicate a breach or data exfiltration attempt.
- **Investigation approach:** Employ network anomaly detection systems to monitor and alert on unusual outbound traffic patterns, particularly encrypted traffic to previously unrecognized IPs or domains.

Advanced persistent threat (APT) hunting

- **Hypothesis:** CoralRaider could deploy sleeper agents within networks to maintain long-term access.
- **Investigation approach:** Use threat hunting techniques to search for indicators of compromises (IoCs) associated with CoralRaider, focusing on persistent scheduled tasks, unusual registry modifications, and anomalous database access patterns.

Decryption and analysis of network communications

- **Hypothesis:** Encrypted communication channels might be used by CoralRaider to hide data exfiltration activities.
- **Investigation approach:** Implement SSL/TLS inspection at the network perimeter to decrypt and inspect encrypted traffic for signs of exfiltration or C2 communication.

Behavioral analysis of endpoint security

- **Hypothesis:** Changes in user behavior or system settings might indicate endpoint compromise.
- **Investigation approach:** Use behavior analysis tools to track and alert on changes to system settings, file access patterns, and execution of scripts that deviate from the norm, which may be indicative of an APT presence.

RETURN

Sources

- GBHackers: CoralRaider Hackers Steal Data
- GBHackers: CoralRaider LNK Antivirus Evasion
- Cyber Unfolded: Safeguarding Against Coral Raider
- Cert CBU: Coral Raider Zararli Kampaniyasi
- Talos Intelligence Blog: CoralRaider Targets Social Media Accounts
- Google Security Blog: Detecting Browser Data Theft Using Windows Event Logs

RETURN

Contact the Converge Threat Intel Group at cybersecurity@convergetp.com

convergetp.com/cybersecurity