

Threat Assessment: TeamPCP - CanisterWorm & Kubernetes Wiper Campaign

By Christoffer StrömbladEntusiastisk Jedi inom Cybersäkerhet cstromblad @nopatience blog@x90.se

Published: 2026-03-22 · Archived: 2026-07-09 02:00:13 UTC



2026-03-22

[Intelligenceteampcpiranthreat-profile](#)

15 minuter att läsa.

Executive Summary

TeamPCP (also tracked as DeadCatx3, PCPcat, ShellForce, and CanisterWorm) is a hybrid cybercrime threat actor that executed a multi-stage supply chain attack campaign between December 2025 and March 2026, primarily targeting the cloud-native security ecosystem. The campaign began with the compromise of Aqua Security's Trivy vulnerability scanner and cascaded into a self-propagating npm worm, the compromise of Checkmarx's KICS GitHub Action, the defacement of Aqua Security's internal GitHub organization, and the deployment of a geopolitically-targeted wiper payload against Iranian systems.

The actor demonstrates a high degree of operational sophistication in cloud-native environments, leveraging stolen credentials to chain supply chain compromises across multiple vendor ecosystems. Their use of Internet Computer Protocol (ICP) blockchain canisters for command-and-control represents a novel and takedown-resistant infrastructure choice. The campaign's scope — over 60,000 compromised servers globally, credential theft from CI/CD pipelines across thousands of repositories, and destructive payloads — places TeamPCP among the most impactful supply chain threat actors observed in 2026.

The group’s motivations appear to be a blend of financial gain (cryptomining, ransomware, extortion), data theft, and attention-seeking behavior, with the Iran-targeted wiper component introducing a geopolitical dimension that remains under assessment.^{[1](#)[2](#)[3](#)[4](#)[5](#)}

Diamond Model Analysis

Attacker

TeamPCP is a threat actor group operating under multiple aliases: **DeadCatx3**, **PCPcat**, **ShellForce**, and **CanisterWorm** (the latter referring specifically to their worm infrastructure).^{[2](#)[4](#)[5](#)}

Key characteristics:

- **Hybrid cybercrime platform:** The group combines infrastructure exploitation, credential theft, access brokering, cryptomining, ransomware/extortion, and destructive operations.^{[4](#)}
- **Operational tempo:** Extremely high. The group iterated through multiple campaign stages in under one week (19–23 March 2026), rapidly pivoting from credential harvesting to worm deployment to organizational defacement to destructive payloads.^{[2](#)[6](#)[7](#)}
- **Technical proficiency:** Deep expertise in cloud-native technologies (Docker, Kubernetes, GitHub Actions, npm), CI/CD pipeline mechanics, and decentralized infrastructure. The group writes custom tooling in Python, Node.js, and Bash.^{[3](#)[8](#)[4](#)}
- **Attention-seeking behavior:** The defacement of all 44 Aqua Security internal repositories with “TeamPCP Owns Aqua Security” messaging and the use of branded naming conventions (`tpcp-docs` , `tpcp.tar.gz`) suggest the actor values public notoriety.^{[2](#)[5](#)}
- **Communication:** The group has been observed extorting victims over Telegram and claiming large-scale data theft from major corporations.^{[5](#)}
- **Attribution confidence:** High, based on consistent reuse of RSA public keys, ICP canister C2 endpoints, encryption schemes, exfiltration archive naming (`tpcp.tar.gz`), and typosquatting patterns across all campaign stages.^{[9](#)[10](#)}

Victimology

The campaign’s victimology is broad and multi-layered, reflecting the cascading nature of supply chain attacks:

Primary targets (direct compromise):

Victim	Nature of Compromise
Aqua Security	Trivy scanner binary, trivy-action, setup-trivy GitHub Actions, aqua-bot service account, Argon-DevOps-Mgt service account, internal GitHub org (44 repos) ^{127}
Checkmarx	KICS GitHub Action (kics-github-action), cx-plugins-releases service account ^{910}

Secondary targets (downstream impact):

Victim Category	Scope
CI/CD pipeline users	Thousands of repositories referencing Trivy and KICS GitHub Actions by mutable version tags ^{7910}
npm ecosystem	50+ npm packages across multiple organizations, including 28 in @EmilGroup scope (~42,000 weekly downloads) and @teale.io/eslint-config (~7,000 weekly downloads) ^{86}
Cloud infrastructure operators	60,000+ servers globally (97% on Azure and AWS) compromised via exposed Docker APIs, Kubernetes clusters, Redis servers, and React2Shell ^{45}

Victim Category	Scope
Iranian systems	Geopolitically targeted for data destruction via wiper payload based on timezone/locale detection 3 5

Sector focus: Cloud-native security vendors, DevSecOps tooling providers, open-source software maintainers, and any organization consuming these tools in CI/CD pipelines. The December 2025 campaign broadly targeted corporate cloud environments without sector specificity.[4](#)

Geographic distribution: Global, with compromised infrastructure concentrated on Azure (61%) and AWS (36%). The wiper component specifically targets systems configured for Iran (Asia/Tehran timezone, fa_IR locale).[3](#) [4](#) [5](#)

Capabilities

TeamPCP demonstrates advanced capabilities across the full attack lifecycle, with particular strength in supply chain compromise and cloud-native exploitation:

Supply chain compromise expertise:

- Force-pushing malicious commits to GitHub Action version tags (75/76 trivy-action tags, 7 setup-trivy tags, 35 KICS tags), exploiting the mutability of Git tags to redirect consumers to malicious code.[7](#) [9](#) [10](#)
- Publishing compromised Docker images to Docker Hub without corresponding GitHub releases, exploiting the disconnect between source and artifact registries.[1](#)
- Self-propagating npm worm that autonomously enumerates and republishes malicious versions across all packages owned by a stolen token holder.[8](#) [6](#)

Credential harvesting:

- Process memory scraping (`/proc/*/mem`) targeting GitHub Actions Runner.Worker processes for secrets.[7](#) [9](#)
- Filesystem sweeping across 50+ sensitive paths for SSH keys, cloud credentials (AWS, GCP, Azure), Kubernetes tokens, Docker configs, cryptocurrency wallets, and npm tokens.[8](#) [7](#)
- AWS IMDS querying (169.254.169.254) for IAM credentials.[9](#)
- Environment variable harvesting from CI/CD runners.[8](#) [6](#) [7](#)

Encryption and exfiltration:

- AES-256-CBC with RSA-4096 hybrid encryption for exfiltration payloads.[7](#) [9](#)
- Consistent exfiltration archive format (`tpcp.tar.gz`) across campaign stages.[7](#) [9](#)
- Vendor-specific typosquatted C2 domains for each target (scan.aquasecurity[.]org, checkmarx[.]zone).[1](#) [7](#) [9](#) [10](#)
- Fallback exfiltration via GitHub repository creation (`tpcp-docs` / `docs-tpcp`) using stolen GITHUB_TOKENs.[7](#) [10](#)

Persistence mechanisms:

- Systemd user services disguised as legitimate processes (`pgmon.service` , `sysmon.py` , `internal-monitor.service`).[3](#) [8](#) [7](#)
- Kubernetes DaemonSets with universal tolerations for cluster-wide scheduling including control plane nodes.[3](#) [4](#)
- Privileged containers with hostPath volume mounts for container escape.[3](#) [4](#)

Destructive capabilities:

- Kubernetes cluster wiper deployed as a privileged DaemonSet that recursively deletes host filesystems and force-reboots all nodes.[3](#) [5](#)
- Standalone host wiper (`rm -rf / --no-preserve-root`) for non-Kubernetes Iranian targets.[3](#)

- Geopolitical targeting logic based on timezone, timedatectl output, and LANG environment variable.^{[3](#)[5](#)}

Evasion techniques:

- 5-minute initial sleep delay to evade sandbox analysis.^{[8](#)[7](#)}
- Spoofed Git commit author identities (impersonating legitimate maintainers).^{[7](#)}
- Mimicry of established naming conventions for reconnaissance branches.^{[2](#)}
- Self-deleting stager scripts.^{[3](#)}
- Base64-encoded payloads.^{[3](#)[8](#)}
- Low-frequency C2 polling (~50 minutes) to minimize network detection signatures.^{[3](#)[8](#)}
- YouTube.com kill-switch URL for operational control.^{[3](#)[8](#)}

Infrastructure

TeamPCP employs a diverse, layered infrastructure designed for resilience and evasion:

Command and Control:

Infrastructure Type	Details
ICP Blockchain Canister	<code>tdtqy-oyaaa-aaaae-af2dq-cai.raw.icp0.io</code> — decentralized, takedown-resistant C2 with <code>update_link</code> method for payload rotation ^{3875}
Typosquatted Domains	<code>scan.aquasecurity[.]org</code> (45.148.10.212), <code>checkmarx[.]zone</code> — newly registered, clean-reputation domains ^{17910}
Cloudflare Tunnels	<code>plug-tab-protective-relay.trycloudflare.com</code> , <code>championships-peoples-point-cassette.trycloudflare.com</code> , <code>investigation-launches-hearings-copying.trycloudflare.com</code> , <code>souls-entire-defined-routes.trycloudflare.com</code> ^{237}
Sliver C2	Used in December 2025 campaign ^{4}
Staging Servers	67.217.57.240, 44.252.85.168 ^{4}

Exfiltration channels:

- HTTPS POST to typosquatted domains with `X-Filename: tpcp.tar.gz` header.^{[9](#)}
- GitHub repository creation (`tpcp-docs` / `docs-tpcp`) with stolen data uploaded as release assets — serves as fallback when primary C2 is disrupted.^{[7](#)[10](#)}

Payload hosting:

- Cloudflare Tunnel URLs for stager and payload delivery.^{[2](#)[3](#)}
- GitHub repositories (DeadCatx3 account) for CIDR scanning lists.^{[4](#)}
- Compromised GitHub Action tags and Docker Hub image tags as distribution vectors.^{[1](#)[7](#)[10](#)}

Compromised accounts:

Account	Organization	Role
aqua-bot	Aqua Security (aquasecurity)	Service account — GPG keys, Docker Hub, Twitter, Slack credentials stolen ^{7}

Account	Organization	Role
Argon-DevOps-Mgt (ID: 139343333)	Aqua Security (aquasec-com & aquasecurity)	Service account — admin-level cross-org access via long-lived PAT ²
cx-plugins-releases (ID: 225848595)	Checkmarx	Service account — used to repoint KICS action tags ¹⁰

File system indicators:

- `/tmp/pglog` — downloaded payload execution path ^{2 3 8}
- `/tmp/.pg_state` — state tracking file ⁸
- `~/.local/share/pgmon/service.py` — npm worm backdoor ⁸
- `~/.config/systemd/user/pgmon.service` — npm worm persistence ⁸
- `~/.config/systemd/user/sysmon.py` — Trivy backdoor persistence ⁷
- `/var/lib/svc_internal/` — Kubernetes backdoor path ³
- `payload.enc` , `tpcp.tar.gz` — exfiltration artifacts ^{1 7}

MITRE ATT&CK Mapping

Reconnaissance

Technique ID	Technique Name	Evidence
T1594	Search Victim-Owned Websites	Threat actor enumerated repos in the aquasec-com org via the GitHub API and studied the Argon-DevOps-Mgt account's behavior before acting. ²
T1590	Gather Victim Network Information	The threat actor studied the service account's naming convention (<code>update-plugin-links-vX.Y.Z</code> pattern) before executing the attack, suggesting reconnaissance of the target organization's internal processes. ²
T1595	Active Scanning	The group industrializes existing vulnerabilities and misconfigurations into a cloud-native exploitation platform, scanning CIDR ranges for exposed Docker APIs, Kubernetes clusters, Redis servers, and Ray dashboards. ^{4 5}

Resource Development

Technique ID	Technique Name	Evidence
T1583.001	Acquire Infrastructure: Domains	Typosquatted C2 domains registered: <code>scan.aquasecurity[.]org</code> (45.148.10.212) and <code>checkmarx[.]zone</code> — newly registered, clean-reputation domains used for exfiltration. ^{1 7 9 10}
T1583.006	Acquire Infrastructure: Web Services	ICP canister (<code>tdtqy-oyaaa-aaaae-af2dq-cai.raw.icp0.io</code>) deployed on the Internet Computer Protocol blockchain as takedown-resistant C2 infrastructure. ^{3 8}
T1583.008	Acquire Infrastructure: Serverless	Cloudflare Tunnel endpoints used for payload staging and C2 communication. ^{2 3 7}

Technique ID	Technique Name	Evidence
T1584.001	Compromise Infrastructure: Domains	Force-pushed 75/76 trivy-action version tags and 35 KICS action tags to malicious commits, hijacking legitimate repository infrastructure. 7 9 10
T1586.003	Compromise Accounts: Cloud Accounts	Compromised service accounts: aqua-bot, Argon-DevOps-Mgt (Aqua Security), and cx-plugins-releases (Checkmarx). 2 7 10
T1608.001	Stage Capabilities: Upload Malware	Compromised Docker images (Trivy 0.69.4, 0.69.5, 0.69.6) pushed to Docker Hub without corresponding GitHub releases. 1
T1608.002	Stage Capabilities: Upload Tool	References to fallback <code>tpcp-docs</code> GitHub repository found within compromised binaries; malicious versions published to npm registry. 1 8 6

Initial Access

Technique ID	Technique Name	Evidence
T1195.001	Supply Chain Compromise: Compromise Software Dependencies and Development Tools	Trivy GitHub Actions tags force-pushed to malicious commits; KICS GitHub Action tags redirected to imposter commits; 50+ npm packages compromised via self-propagating worm. 8 6 7 10
T1195.002	Supply Chain Compromise: Compromise Software Supply Chain	CanisterWorm self-propagated across npm packages by republishing malicious versions under stolen maintainer identities. 8 6 5
T1195.003	Supply Chain Compromise: Compromise Hardware Supply Chain / Software Dependencies	Backdoored Trivy binaries published to GitHub Releases, Docker Hub, GHCR, and ECR — compromising the core scanner artifact itself. 7
T1078.004	Valid Accounts: Cloud Accounts	Stolen PATs and npm tokens used to authenticate as legitimate service accounts and package maintainers across GitHub and npm registries. 2 8 6 10
T1190	Exploit Public-Facing Application	React2Shell (CVE-2025-29927) exploitation against Next.js applications; exploitation of exposed Docker APIs, Kubernetes clusters, and Redis servers. 4 5
T1133	External Remote Services	Unauthenticated Docker, Kubernetes, Redis, and Ray management APIs used for initial access in the December 2025 campaign. 4

Execution

Technique ID	Technique Name	Evidence
T1059.004	Command and Scripting Interpreter: Unix Shell	Stager scripts (<code>kamikaze.sh</code> , <code>entrypoint.sh</code> , <code>setup.sh</code>) execute shell commands for payload delivery, credential harvesting, and filesystem destruction. 3 7 9

Technique ID	Technique Name	Evidence
T1059.006	Command and Scripting Interpreter: Python	Python payloads (kube.py , service.py , sysmon.py , runner.py) used for environment fingerprinting, backdoor persistence, and C2 polling. ^{3 8 7}
T1610	Deploy Container	Malicious DaemonSets deployed via kubectl apply , scheduling privileged containers on every node in Kubernetes clusters. ^{3 4}
T1609	Container Administration Command	Docker API and Ray job submission used for remote workload execution in the December 2025 campaign. ⁴
T1648	Serverless Execution	Automated GitHub API calls (PATCH /repos/{org}/{repo}) executed in a scripted 2-minute burst to rename all 44 repositories. ²

Persistence

Technique ID	Technique Name	Evidence
T1543.002	Create or Modify System Process: Systemd Service	Systemd user services created for persistence: pgmon.service (npm worm), sysmon.py (Trivy backdoor), internal-monitor.service (Kubernetes backdoor) — all without requiring root access. ^{3 8 7}
T1078	Valid Accounts	Long-lived PATs for service accounts (Argon-DevOps-Mgt, cx-plugins-releases) provided persistent cross-organization access without MFA. ^{2 10}
T1525	Implant Internal Image	Malicious containers deployed with auto-restart policies; compromised Docker images published to official registries. ^{4 5}
T1053.003	Scheduled Task/Job: Cron	Systemd services and scheduled scanners used for persistent execution in the December 2025 campaign. ⁴

Privilege Escalation

Technique ID	Technique Name	Evidence
T1611	Escape to Host	Both DaemonSets mount the host root filesystem (hostPath: path: /) and run with privileged: true , allowing container escape to the underlying host. ^{3 4}
T1548.003	Abuse Elevation Control Mechanism: Sudo and Sudo Caching	The poison_pill function attempts sudo -n rm -rf / --no-preserve-root to escalate privileges when not already root. ³

Defense Evasion

Technique ID	Technique Name	Evidence
T1036.003	Masquerading: Rename System Utilities	Imposter commits pushed to actions/checkout (spoofing user rauchg) and aquasecurity/trivy (spoofing user DmitryLewen). ⁷

Technique ID	Technique Name	Evidence
T1036.004	Masquerading: Masquerade Task or Service	Service names designed to mimic legitimate processes: <code>pgmon</code> (PostgreSQL monitor), <code>internal-monitor</code> , <code>host-provisioner-std</code> . 38
T1036.005	Masquerading: Match Legitimate Name or Location	Typosquatted C2 domains designed to appear as legitimate vendor infrastructure in CI/CD logs; reconnaissance branch mimicked established naming conventions. 29
T1027	Obfuscated Files or Information	Backdoor Python code stored as base64-encoded strings; exfiltration payloads encrypted with AES-256-CBC + RSA-4096 hybrid encryption. 3879
T1070.004	Indicator Removal: File Deletion	Stager scripts self-delete after execution (<code>rm -- "\$0"</code>); reconnaissance branches created and immediately deleted; original <code>package.json</code> restored after malicious npm publish. 236
T1497.003	Virtualization/Sandbox Evasion: Time Based Evasion	5-minute initial sleep delay before backdoor activation to evade sandbox analysis timeouts. 87
T1562.001	Impair Defenses: Disable or Modify Tools	C2 polling interval of ~50 minutes (3,000 seconds) minimizes network traffic signatures, reducing detection probability. 8

Credential Access

Technique ID	Technique Name	Evidence
T1552.001	Unsecured Credentials: Credentials in Files	Harvesting of <code>.npmrc</code> files, <code>.env</code> files, SSH keys, cloud credentials, Git credentials from filesystem paths across CI/CD runners and developer machines. 8674
T1552.004	Unsecured Credentials: Private Keys	Filesystem sweeping for SSH keys, GPG keys, and cloud provider credentials across 50+ sensitive file paths. 79
T1552.005	Unsecured Credentials: Cloud Instance Metadata API	AWS IMDS queried at <code>169.254.169.254</code> for IAM security credentials. 9
T1528	Steal Application Access Token	GitHub PATs, npm tokens, Kubernetes service account tokens, and Docker configs systematically harvested from CI runner environments. 287910
T1003	OS Credential Dumping	Runner.Worker process memory scraped via <code>/proc/<pid>/mem</code> searching for <code>{"value": "<secret>", "isSecret": true}</code> patterns. 29

Discovery

Technique ID	Technique Name	Evidence
T1082	System Information Discovery	Environment fingerprinting: checking <code>GITHUB_ACTIONS</code> variable, <code>/etc/timezone</code> , <code>timedatectl</code> , and <code>LANG</code> for geopolitical targeting and execution context. 375

Technique ID	Technique Name	Evidence
T1083	File and Directory Discovery	Filesystem sweeping across 50+ paths for credentials; <code>.npmrc</code> enumeration in home, current, and <code>/etc/</code> directories. 8 7
T1613	Container and Resource Discovery	<code>is_k8s()</code> function checks for Kubernetes service account paths and <code>KUBERNETES_SERVICE_HOST</code> environment variable; pod and namespace enumeration. 3 4
T1526	Cloud Service Discovery	npm search API queried to enumerate all packages owned by compromised token holders; GitHub org repositories enumerated via API. 2 8 6
T1614	System Location Discovery	Timezone and locale checks (<code>Asia/Tehran</code> , <code>fa_IR</code>) used to determine if victim systems are located in Iran for wiper targeting. 3 5
T1046	Network Service Scanning	CIDR-based scanning for exposed Docker APIs, Kubernetes clusters, Redis servers, and Ray dashboards. 4

Lateral Movement

Technique ID	Technique Name	Evidence
T1550.001	Use Alternate Authentication Material: Application Access Tokens	Stolen PAT for Argon-DevOps-Mgt provided write/admin access across both aquasec-com and aquasecurity GitHub organizations; stolen GitHub tokens from Trivy compromise enabled KICS action compromise. 2 9
T1570	Lateral Tool Transfer	Kubernetes DaemonSets with universal tolerations (<code>operator: Exists</code>) schedule on every node including control plane, propagating backdoors cluster-wide. 3
T1080	Taint Shared Content	npm worm publishes malicious versions to every package owned by each compromised token holder, propagating via shared registry infrastructure. 8 6
T1072	Software Deployment Tools	<code>deploy.js</code> uses legitimate npm CLI tooling to authenticate, enumerate, and republish malicious packages across victim portfolios. 8 6

Collection

Technique ID	Technique Name	Evidence
T1005	Data from Local System	Credential harvester enumerates secrets from environment variables and filesystem, including SSH keys, cloud credentials, Kubernetes tokens, cryptocurrency wallets, and npm tokens. 8 7 9
T1213	Data from Information Repositories	44 internal Aqua Security repositories exposed publicly — including source code for Tracee, internal Trivy forks, CI/CD pipelines, Kubernetes operators, and team knowledge bases. 2

Technique ID	Technique Name	Evidence
T1530	Data from Cloud Storage	Claims of stealing vast amounts of sensitive data from major companies, including a large multinational pharmaceutical firm. ⁵

Command and Control

Technique ID	Technique Name	Evidence
T1102	Web Service	ICP canister (<code>tdtqy-oyaaa-aaaae-af2dq-cai.raw.icp0.io</code>) used as decentralized, blockchain-based C2 — first observed use of ICP canisters as C2 infrastructure. ^{2 3 8 7 5}
T1071.001	Application Layer Protocol: Web Protocols	HTTPS POST with <code>X-Filename</code> header used for exfiltration to typosquatted domains; HTTPS polling of ICP canister for payload URLs. ^{1 9}
T1572	Protocol Tunneling	Cloudflare Tunnels used as C2 infrastructure for payload delivery and exfiltration. ^{2 3 7}
T1008	Fallback Channels	ICP canister <code>update_link</code> method allows payload rotation without touching infected packages; GitHub repository creation (<code>tcp-docs / docs-tcp</code>) serves as fallback when primary C2 is disrupted. ^{8 7 10}
T1090	Proxy	FRPS, gost, and P2P relays used in the December 2025 campaign for proxy infrastructure. ⁴
T1573	Encrypted Channel	Sliver C2 framework used for encrypted communications in the December 2025 campaign. ⁴

Exfiltration

Technique ID	Technique Name	Evidence
T1041	Exfiltration Over C2 Channel	Encrypted <code>tcp.tar.gz</code> archives exfiltrated via HTTPS POST to typosquatted C2 domains (<code>scan.aquasecurtiy[.]org</code> , <code>checkmarx[.]zone</code>). ^{7 9}
T1567.001	Exfiltration Over Web Service: Exfiltration to Code Repository	Fallback exfiltration creates <code>tcp-docs / docs-tcp</code> repositories in victim GitHub organizations and uploads stolen credentials as release assets. ^{7 10}
T1048.003	Exfiltration Over Alternative Protocol: Exfiltration Over Unencrypted Non-C2 Protocol	curl uploads of <code>tcp.tar.gz</code> using <code>--data-binary</code> to external domains over HTTPS with consistent <code>X-Filename</code> header pattern. ⁹

Impact

Technique ID	Technique Name	Evidence
T1485	Data Destruction	Kubernetes wiper DaemonSet (<code>host-provisioner-iran</code>) recursively deletes host filesystems on all cluster nodes; standalone wiper executes <code>rm -rf / --no-preserve-root</code> on Iranian systems. ^{3 5}
T1529	System Shutdown/Reboot	After wiping host filesystems, the kamikaze container executes <code>chroot /mnt/host reboot -f</code> to force reboot every affected node. ³
T1491.001	Defacement: Internal Defacement	All 44 Aqua Security internal repositories renamed with <code>tpcp-docs-</code> prefix and descriptions changed to “TeamPCP Owns Aqua Security” in a scripted 2-minute burst. ²
T1496	Resource Hijacking	XMRig cryptomining deployed across compromised cloud infrastructure in the December 2025 campaign. ⁴
T1486	Data Encrypted for Impact	The threat actor claims ransomware/extortion capabilities; victims extorted over Telegram. ^{4 5}
T1657	Financial Theft	Credential theft enabling extortion of victims over Telegram, with claims of stealing sensitive corporate data. ⁵

Campaign Timeline

Date	Event
December 2025	TeamPCP launches mass campaign targeting exposed Docker APIs, Kubernetes clusters, Redis servers, and React2Shell (CVE-2025-29927). Over 60,000 servers compromised globally. ⁴
~19 March 2026	TeamPCP compromises Aqua Security’s Trivy scanner: malicious binary v0.69.4, force-pushed <code>trivy-action</code> (75/76 tags) and <code>setup-trivy</code> (7 tags) GitHub Actions. Credential harvester deployed across CI/CD pipelines. ⁷
~20 March 2026	Self-propagating npm worm (CanisterWorm) deployed using npm tokens stolen from Trivy compromise. 50+ packages compromised including <code>@EmilGroup</code> scope (28 packages, ~42K weekly downloads). ^{8 6}
~22 March 2026	Geopolitically-targeted wiper/backdoor payload deployed via same infrastructure. Iranian systems targeted for destruction; non-Iranian systems receive CanisterWorm backdoor. ^{3 5}
22 March 2026	Compromised Trivy Docker images 0.69.4, 0.69.5, 0.69.6 pushed to Docker Hub without GitHub releases. ¹
22 March 2026 (20:31–20:32 UTC)	Aqua Security internal GitHub org (<code>aquasec-com</code>) defaced: all 44 repositories renamed with <code>tpcp-docs-</code> prefix via compromised Argon-DevOps-Mgt service account. ²
~23 March 2026	Checkmarx KICS GitHub Action compromised: all 35 version tags redirected to malicious commits via compromised <code>cx-plugins-releases</code> service account. New C2 domain <code>checkmarx[.]zone</code> deployed. ^{9 10}

Key Indicators of Compromise

Christoffer Strömblad

Entusiastisk Jedi inom Cybersäkerhet

Page 13 of 13