INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-04-03 13:38 UTC

TeamPCP Campaign Weaponizes Security Scanning Tools to Compromise 1,000+ SaaS Environments via CI/CD Abuse

THREAT CAMPAIGN | **CRITICAL** | CVSS 9.5

SCC Item ID	SCC-CAM-2026-0142
Type	Threat Campaign
Severity	CRITICAL
CVSS Base Score	9.5
Affected Products	SaaS environments (multiple unspecified vendors), CI/CD pipeline tooling, security scanning tools, European Commission cloud infrastructure
Published	2026-04-03T09:18:01
Discovery Source	Rss

Executive Summary

The TeamPCP campaign has compromised approximately 1,000+ SaaS environments (per secondary reporting; primary source verification pending) by weaponizing organizations' own security scanning tools against their CI/CD pipelines, a supply chain intrusion method that bypasses conventional trust boundaries. CERT-EU has confirmed a breach of European Commission cloud infrastructure attributed to this campaign, which was still actively expanding as of late March 2026. Organizations running automated security scanning in CI/CD pipelines face material risk of credential theft, cloud account takeover, and lateral movement across connected SaaS environments. NOTE: Attribution to Mandiant and CERT-EU is drawn from secondary reporting; confidence in technical specifics is MEDIUM pending primary source confirmation.

Technical Analysis

TeamPCP exploits implicit trust in security scanning tools embedded within CI/CD pipelines to gain initial access and move laterally into cloud-hosted SaaS environments. No CVE has been assigned; the campaign maps to four CWEs: CWE-250 (excessive privilege granted to scanning tool service accounts), CWE-522 (credentials insufficiently protected within pipeline configurations), CWE-829 (inclusion of functionality from untrusted control sphere via compromised scanner plugins or dependencies), and CWE-913 (improper control of dynamically managed code resources executed during pipeline runs). MITRE ATT&CK coverage includes T1195.002 (compromise software supply chain), T1199 (trusted relationship abuse), T1072 (software deployment tools),

T1609 (container administration command), T1078 and T1078.004 (valid accounts, cloud accounts), T1552 and T1552.001 (unsecured credentials, credentials in files), T1650 (acquire access), T1526 (cloud service discovery), T1530 (data from cloud storage), T1190 (exploit public-facing application), and T1059 (command and scripting interpreter). The attack surface is the build and deployment pipeline itself, adversaries operate through tooling already trusted by the environment, making detection difficult with standard signature-based controls. Affected scope includes multiple unspecified SaaS vendors and European Commission cloud infrastructure. No vendor-issued patch exists; remediation is configuration and access control driven. All sources are Tier 3 secondary reporting; primary source documents from Mandiant and CERT-EU have not been directly verified in this analysis.

Action Checklist

1. Step 1: Containment. Immediately audit and restrict permissions on all service accounts used by security scanning tools in CI/CD pipelines. Revoke any scanner service account with write access to production environments or broad cloud IAM permissions. Isolate pipelines showing anomalous execution behavior pending review. Prioritize environments running cloud-connected CSPM or SAST/DAST tooling with OAuth or long-lived token authentication.
2. Step 2: Detection. Query cloud access logs (AWS CloudTrail, Azure Monitor, GCP Audit Logs) for API calls originating from CI/CD runner IPs or scanner service account identities outside expected pipeline execution windows. Look for T1526 indicators: ListBuckets, DescribeInstances, or equivalent enumeration calls from scanner identities. Check pipeline logs for unexpected outbound connections or dynamic code execution (T1059) during scan phases. Review secrets managers and environment variable stores for unauthorized read events (T1552.001).
3. Step 3: Eradication. Rotate all credentials and tokens associated with security scanning tools. Replace long-lived static credentials with short-lived, scoped tokens where the CI/CD platform supports OIDC federation (e.g., GitHub Actions OIDC, GitLab CI JWT). Remove unnecessary plugin dependencies from scanner configurations and pin remaining dependencies to verified, hash-validated versions to address CWE-829. Apply least-privilege IAM policies to scanner service accounts per NIST SP 800-53 AC-6.
4. Step 4: Recovery. Validate that rotated credentials are not cached in pipeline artifacts, container images, or log outputs. Re-run pipeline jobs under monitored conditions and confirm no anomalous cloud API calls occur during scan phases. Establish a baseline of expected scanner behavior (call volume, target services, timing) and alert on deviation. Verify European Commission and any shared-tenant SaaS connections for signs of data access (T1530) during the campaign window.
5. Step 5: Post-Incident. Conduct a pipeline privilege audit across all CI/CD tooling using CIS Benchmarks for CI/CD security as a reference baseline. Map scanner service accounts against the principle of least privilege (NIST SP 800-53 AC-6, IA-5). Implement pipeline integrity controls: signed commits, artifact attestation, and dependency pinning. Review NIST SP 800-161r1 (C-SCRM) for supply chain risk management controls applicable to third-party tooling embedded in build pipelines. Document trust assumptions embedded in current scanner configurations as a recurring risk register item.

IR / Forensic Enrichment

Triage Priority	IMMEDIATE

Escalation Criteria	Escalate to CISO and legal counsel immediately if cloud audit logs confirm scanner service account API calls to data storage services (T1530: ListBuckets, GetObject, storage.objects.get) during the TeamPCP campaign window (approximately late March 2026 or earlier), as this indicates potential data exfiltration triggering breach notification obligations under GDPR Article 33 (72-hour supervisory authority notification) and applicable US state privacy laws, or if any shared-tenant SaaS connection to European Commission infrastructure is identified.
Recovery Notes	After credential rotation and pipeline isolation, re-enable pipelines incrementally — one pipeline at a time — under active cloud audit log monitoring, comparing real-time API call patterns against the clean behavioral baseline established during the monitored recovery re-run. Maintain elevated log retention (minimum 90 days, extended from default) for all cloud provider audit logs, CI/CD execution logs, and secrets manager access logs covering the campaign window, as late-discovered indicators may require retrospective analysis. Continue daily review of scanner service account API call volumes and target service sets for a minimum of 30 days post-recovery, given that CERT-EU confirmed the campaign was still actively expanding as of late March 2026 and reinfection via upstream supply chain vectors remains possible.
Forensic Artifacts	Cloud provider audit logs (AWS CloudTrail, Azure Monitor Activity Log, GCP Audit Logs) filtered on scanner service account identities for GetSecretValue, ListBuckets, DescribeInstances, AssumeRole, and storage object-read events during the campaign window — the primary evidence source for T1526 cloud enumeration and T1552.001 credential access by TeamPCP. | CI/CD pipeline execution logs (GitHub Actions workflow run logs, GitLab CI job traces, Jenkins build console output) for all jobs that invoked security scanning tools, specifically capturing the subprocess tree and network activity during scan phases — evidence of T1059 dynamic code execution injected into scanner execution context. | Scanner tool plugin and dependency manifests (requirements.txt, package-lock.json, pom.xml, .snyk files, .checkov.yml, sonar-project.properties) at the versions present during the compromise window, preserved for static analysis to identify CWE-829 (Inclusion of Functionality from Untrusted Control Sphere) as the likely TeamPCP ingress mechanism. | Secrets manager and environment variable store access logs (AWS Secrets Manager CloudTrail events, HashiCorp Vault audit log, GitHub Actions secrets audit log via GitHub Enterprise audit log streaming) showing which pipeline jobs read which secrets during the campaign window — direct evidence of T1552.001 (Credentials in Files) exploitation. | Container image layer digests and build artifact SBOMs (Software Bill of Materials) for scanner tool images used in affected pipelines during the campaign window, to establish whether a compromised scanner image or plugin version was the persistence mechanism and to support SLSA provenance gap analysis.

Per-Action IR Details

Step 1: Containment — Audit and immediately restrict permissions on all service accounts used by security scanning tools in CI/CD pipelines. Revoke any scanner service account with write access to production environments or broad cloud IAM permissions. Isolate pipelines showing anomalous execution behavior pending review. Prioritize environments running cloud-connected CSPM or SAST/DAST tooling with OAuth or long-lived token authentication.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy

Controls: NIST IR-4 (Incident Handling), NIST AC-6 (Least Privilege), NIST AC-2 (Account Management), CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts), CIS 6.2 (Establish an Access Revoking Process)

Compensating: Export IAM policy JSON for all scanner service accounts using `aws iam list-attached-role-policies --role-name ` and `gcloud iam service-accounts get-iam-policy @.iam.gserviceaccount.com` or `az role assignment list --assignee `. Diff current permissions against a pre-campaign baseline to identify scope creep. For GitHub Actions: run `gh api /repos///actions/secrets` to enumerate accessible secrets. Block outbound internet from CI runner hosts via host-based firewall (`ufw deny out` or `iptables -A OUTPUT -j DROP`) for any runner not in an expected execution window.

Evidence: Before revoking tokens, capture full IAM policy snapshots for each scanner service account (AWS: `aws iam get-role --role-name`, Azure: `az ad sp show`, GCP: `gcloud iam service-accounts describe`). Preserve CloudTrail, Azure Monitor, and GCP Audit Log entries showing the service account's last 30 days of API call history — these will establish the pre-revocation footprint for later attribution to TeamPCP activity. Export pipeline execution logs from GitHub Actions, GitLab CI, or Jenkins showing which jobs ran with which identities during the campaign window (approximately late March 2026 and prior).

Step 2: Detection — Query cloud access logs (AWS CloudTrail, Azure Monitor, GCP Audit Logs) for API calls originating from CI/CD runner IPs or scanner service account identities outside expected pipeline execution windows. Look for T1526 indicators: ListBuckets, DescribeInstances, or equivalent enumeration calls from scanner identities. Check pipeline logs for unexpected outbound connections or dynamic code execution (T1059) during scan phases. Review secrets managers and environment variable stores for unauthorized read events (T1552.001).

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis

Controls: NIST SI-4 (System Monitoring), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-2 (Event Logging), NIST IR-5 (Incident Monitoring), CIS 8.2 (Collect Audit Logs)

Compensating: Without a SIEM, run targeted CloudTrail queries using AWS CLI: `aws cloudtrails lookup-events --lookup-attributes AttributeKey=Username,AttributeValue= --start-time ` and filter output with `jq` for `ListBuckets`, `DescribeInstances`, `GetSecretValue`, `AssumeRole`. For GCP: `gcloud logging read 'protoPayload.authenticationInfo.principalEmail="" AND protoPayload.methodName=("storage.buckets.list" OR "compute.instances.list")' --limit=500`. For AWS Secrets Manager unauthorized reads: query CloudTrail for `GetSecretValue` events where `errorCode` is absent but the caller is a scanner identity. Install Falco (free, open source) on CI runner hosts to detect anomalous subprocess spawning (T1059) during scan phases via rules targeting unexpected child processes of scanner binaries.

Evidence: Capture the following before any log retention window expires: (1) AWS CloudTrail S3 bucket or CloudWatch Logs export covering all `GetSecretValue`, `ListBuckets`, `DescribeInstances`, `AssumeRole`, and `CreatePresignedUrl` events from scanner service account ARNs. (2) GCP Audit Logs `DATA_READ` events on Secret Manager and Cloud Storage from scanner service account emails. (3) Azure Monitor activity logs filtering on scanner service principal object IDs for `Microsoft.KeyVault/vaults/secrets/read` and `Microsoft.Authorization/roleAssignments/write` operations. (4) CI/CD pipeline execution logs (`.github/workflows` run logs, GitLab CI job traces, Jenkins build logs) for the job steps corresponding to scanner execution phases — these will show injected outbound connection attempts consistent with T1526 and T1552.001.

Step 3: Eradication — Rotate all credentials and tokens associated with security scanning tools. Replace long-lived static credentials with short-lived, scoped tokens where the CI/CD platform supports OIDC federation (e.g., GitHub Actions OIDC, GitLab CI JWT). Remove unnecessary plugin dependencies from scanner configurations and pin remaining dependencies to verified, hash-validated versions to address CWE-829. Apply least-privilege IAM policies to scanner service accounts per CWE-250 remediation guidance in NIST SP 800-53 AC-6.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: NIST SI-2 (Flaw Remediation), NIST AC-6 (Least Privilege), NIST IA-5 (Authenticator Management), NIST CM-3 (Configuration Change Control), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process)

Compensating: Automate credential rotation using native CLI: AWS `aws iam delete-access-key` + `aws iam create-access-key`; GCP `gcloud iam service-accounts keys delete` + `gcloud iam service-accounts keys create`; Azure `az ad sp credential reset`. For dependency pinning, run `pip-audit` or `npm audit` against scanner plugin manifests and enforce hash pinning in `requirements.txt` (`package==1.2.3 --hash=sha256:`) or `package-lock.json` with `integrity` fields. Validate OIDC federation configuration for GitHub Actions using the official `actions/configure-aws-credentials` action with `role-to-assume` and no static `AWS_ACCESS_KEY_ID` — confirm no static keys remain in repo secrets via `git log --all --full-history -- '**/.env' '**/*.key'` and `trufflehog git file://. --only-verified`.

Evidence: Before rotating, document the full list of credentials in scope: export GitHub Actions secrets names (`gh api /orgs//actions/secrets`), GitLab CI/CD variable names (`gitlab-rails runner 'Project.find().variables.map(&:key)'`), and AWS Secrets Manager secret ARNs accessed by scanner identities. Preserve a read-only snapshot of the scanner tool's plugin dependency tree (`pip freeze`, `npm list --all`, or equivalent) to support post-incident analysis of whether a malicious plugin was the TeamPCP ingress vector. Capture any scanner configuration files (`snyk.json`, `.sonarcloud.properties`, `.checkov.yml`, etc.) present in pipeline repos at the time of compromise for later static analysis.

Step 4: Recovery — Validate that rotated credentials are not cached in pipeline artifacts, container images, or log outputs. Re-run pipeline jobs under monitored conditions and confirm no anomalous cloud API calls occur during scan phases. Establish a baseline of expected scanner behavior (call volume, target services, timing) and alert on deviation. Verify European Commission and any shared-tenant SaaS connections for signs of data access (T1530) during the campaign window.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST IR-4 (Incident Handling), NIST SI-7 (Software, Firmware, and Information Integrity), NIST AU-6 (Audit Record Review, Analysis, and Reporting), NIST AU-9 (Protection of Audit Information), CIS 4.6 (Securely Manage Enterprise Assets and Software)

Compensating: Scan all published container images and pipeline artifacts for embedded credentials using `trufflehog image :` and `gitleaks detect --source ./artifacts`. For pipeline re-run validation, capture a Wireshark/tcpdump pcap on the CI runner's network interface during a controlled scan job execution: `tcpdump -i eth0 -w scanner-rerun-$(date +%s).pcap` and review for unexpected egress IPs. Cross-reference cloud provider audit logs during the controlled re-run to confirm scanner API calls match your documented legitimate call set. For T1530 (Data from Cloud Storage Object) validation against the European Commission campaign window, query CloudTrail `GetObject` and GCP `storage.objects.get` events for scanner identities during the late March 2026 timeframe.

Evidence: Capture network flow data from CI runner hosts during the monitored recovery re-run (netflow or pcap) to establish a verified clean behavioral baseline — this will serve as the comparison artifact for future anomaly detection. Export a timestamped snapshot of all S3, GCS, and Azure Blob object-access logs for the campaign window to document any T1530 data access attributable to TeamPCP scanner identities before those log records age out. Preserve container image layer digests for scanner tool images in use during the compromise window (`docker inspect --format='{{json .RootFS.Layers}}' `) to support later integrity verification.

Step 5: Post-Incident — Conduct a pipeline privilege audit across all CI/CD tooling using CIS Benchmarks for CI/CD security as a reference baseline. Map scanner service accounts against the principle of least privilege (NIST SP 800-53 AC-6, IA-5). Implement pipeline integrity controls: signed commits, artifact attestation, and dependency pinning. Review NIST SP 800-161r1 (C-SCRM) for supply chain risk management controls applicable to third-party tooling embedded in build pipelines. Document trust assumptions embedded in current scanner configurations as a recurring risk register item.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity

Controls: NIST IR-4 (Incident Handling), NIST IR-8 (Incident Response Plan), NIST AC-6 (Least Privilege), NIST IA-5 (Authenticator Management), NIST SI-7 (Software, Firmware, and Information Integrity), NIST RA-3 (Risk Assessment), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 5.1 (Establish and Maintain

an Inventory of Accounts)

Compensating: Generate a full CI/CD service account inventory with permission mappings using `aws iam generate-service-last-accessed-details --arn ` to identify unused permissions that should be stripped — this directly addresses the TeamPCP exploitation pattern of over-permissioned scanner accounts. Implement SLSA (Supply-chain Levels for Software Artifacts) provenance at Level 1 minimum using the free `slsa-github-generator` action for GitHub Actions or equivalent. For dependency pinning enforcement, add a pre-commit hook using `detect-secrets` and a CI gate running `pip-audit` or `snyk test --file=requirements.txt` on scanner plugin manifests. Document trust assumptions using a simple YAML-based threat model file committed to the repo (`SECURITY.md` or a dedicated `threatmodel.yml`) updated as a recurring sprint task.

Evidence: Assemble the complete incident timeline from preserved artifacts: CI/CD job execution timestamps, cloud audit log API call sequences, and any scanner plugin version histories from package manager logs — this narrative will support the NIST 800-61r3 §4 lessons-learned review and any regulatory notification obligations (particularly relevant given the confirmed European Commission breach). Preserve the pre-incident scanner configuration files, IAM policy snapshots, and pipeline workflow YAML files as reference artifacts for the post-incident risk register update and C-SCRM control gap analysis against NIST SP 800-161r1.

Detection Guidance

Focus detection on behavioral anomalies from scanner and CI/CD service account identities rather than signature-based indicators, as no public IOCs have been confirmed for this campaign. Key detection approaches: (1) Cloud audit logs, alert on enumeration API calls (S3 ListBuckets, EC2 DescribeInstances, Azure Resource Graph queries) issued by scanner service account identities, especially outside pipeline execution windows. (2) CI/CD runner logs, flag outbound connections to non-expected endpoints during scan job execution phases; look for dynamic code execution or interpreter invocation (T1059) within scanner job steps. (3) Credential access, monitor secrets manager access logs for reads by scanner identities not correlated with a running pipeline job (T1552.001). (4) Lateral movement, correlate scanner service account activity with downstream SaaS API calls, particularly OAuth token issuance or cloud service discovery requests (T1526). (5) Supply chain, diff scanner plugin or dependency manifests against last-known-good state; flag additions or version changes not associated with an approved change record (T1195.002). No confirmed IOC hashes, IPs, or domains are available from verified primary sources as of this analysis. IOC list is empty pending primary source confirmation from Mandiant or CERT-EU.

Framework Mappings

MITRE-ATTACK

- **T1530** — Data from Cloud Storage
- **T1078** — Valid Accounts
- **T1609** — Container Administration Command
- **T1552** — Unsecured Credentials
- **T1078.004** — Cloud Accounts
- **T1650** — Acquire Access
- **T1190** — Exploit Public-Facing Application
- **T1199** — Trusted Relationship
- **T1552.001** — Credentials In Files

- **T1526** — Cloud Service Discovery
- **T1195.002** — Compromise Software Supply Chain
- **T1059** — Command and Scripting Interpreter

NIST-800-53R5

- **AC-2** — Account Management
- **AC-6** — Least Privilege
- **IA-2** — Identification and Authentication (Organizational Users)
- **IA-5** — Authenticator Management
- **CA-8** — Penetration Testing
- **RA-5** — Vulnerability Monitoring and Scanning
- **SC-7** — Boundary Protection
- **SI-2** — Flaw Remediation
- **SI-7** — Software, Firmware, and Information Integrity
- **CM-7** — Least Functionality
- **SA-9** — External System Services
- **SR-3** — Supply Chain Controls and Processes
- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **AT-2** — Literacy Training and Awareness
- **SR-2** — Supply Chain Risk Management Plan

OWASP-TOP10-2021

- **A04:2021** — Insecure Design
- **A07:2021** — Identification and Authentication Failures
- **A01:2021** — Broken Access Control

CIS-V8

- **5.2** — Use Unique Passwords
- **5.4** — Restrict Administrator Privileges to Dedicated Administrator Accounts
- **6.8** — Define and Maintain Role-Based Access Control
- **6.3** — Require MFA for Externally-Exposed Applications
- **14.2** — Train Workforce Members to Recognize Social Engineering Attacks
- **15.1** — Establish and Maintain an Inventory of Service Providers
- **8.2** — Collect Audit Logs

HIPAA-SECURITY

- **164.308(a)(5)(ii)(D)** — Password Management
- **164.312(d)** — Person or Entity Authentication

SOC2-TSC

- **CC6.1** — Logical access security software, infrastructure, and architectures

- **CC9.2** — Manages risks associated with vendors and business partners

ISO-27001-2022

- **A.5.34** — Privacy and protection of personal information
- **A.5.21** — Managing information security in the ICT supply chain
- **A.5.23** — Information security for use of cloud services

NIST-CSF-2

- **GV.SC-01** — Cybersecurity supply chain risk management program
- **DE.CM-01** — Networks and network services are monitored

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1530	Data from Cloud Storage	Collection
T1078	Valid Accounts	Defense-Evasion
T1609	Container Administration Command	Execution
T1552	Unsecured Credentials	Credential-Access
T1078.004	Cloud Accounts	Defense-Evasion
T1650	Acquire Access	Resource-Development
T1190	Exploit Public-Facing Application	Initial-Access
T1199	Trusted Relationship	Initial-Access
T1552.001	Credentials In Files	Credential-Access
T1526	Cloud Service Discovery	Discovery
T1195.002	Compromise Software Supply Chain	Initial-Access
T1059	Command and Scripting Interpreter	Execution

Sources

Source	URL	Tier
Security News	https://www.sans.org/white-papers/when-security-scanner-became-weapon	T3
Second data breach at European Commission this year leaves open ...	https://www.helpnetsecurity.com/2026/03/30/european-commission-cybe...	T3

Source	URL	Tier
Cloud Misconfiguration Risks in Focus After European Commission ...	https://coesecurity.com/cloud-misconfiguration-risks-in-focus-after...	T3
Your Cloud(s). Adversaries' Chance At Control | Group-IB Blog	https://www.group-ib.com/blog/multicloud-cspm-security/	T3
[PDF] Cloud Sovereignty Framework - European Commission	https://commission.europa.eu/document/download/09579818-64a6-4dd5-9...	T1

Generated 2026-04-03 13:38 UTC by TJS Security Command Center