INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-06-19 13:53 UTC

AI Agents Are the New Privileged Accounts: Why Shadow AI Is Now an Identity Crisis

GOVERNANCE | HIGH | CVSS 7.5

SCC Item ID	SCC-GOV-2026-0063
Type	Governance
Severity	HIGH
CVSS Base Score	7.5
Affected Products	Enterprise environments using AI agents integrated with Salesforce, Snowflake, GitHub, Gong, Slack, and MCP servers; no specific vulnerable product version identified
Published	2026-06-19T06:30:00
Discovery Source	Rss

Executive Summary

Enterprise AI agents deployed outside IT governance are generating long-lived API keys, OAuth tokens, and service account credentials that persist indefinitely without audit or revocation. Token Security's Agentic Pulse dataset reports that 65.4% of deployed agentic chatbots have never been used since creation, yet their credentials remain active, creating a dormant privileged access surface that resembles orphaned accounts. Any attacker who obtains one of these credentials gains broad production access to systems including Salesforce, Snowflake, GitHub, and Slack, with no behavioral baseline to trigger anomaly detection.

Technical Analysis

Shadow AI deployments authenticate to production systems via service accounts, API keys, and OAuth tokens that are typically over-permissioned, never scoped to least privilege, and never enrolled in standard IAM lifecycle processes. The attack surface has three layers: (1) dormant agent credentials (CWE-522, CWE-269) that remain valid after the agent is abandoned; (2) excessive permission assignment on those credentials (CWE-284, CWE-732) allowing lateral movement into connected SaaS systems; and (3) MCP (Model Context Protocol) server integrations introducing SSRF and indirect prompt injection vectors when agents interact with external tools. Relevant MITRE ATT&CK techniques include T1078 (Valid Accounts), T1078.004 (Cloud Accounts), T1098 (Account Manipulation), T1136.003 (Cloud Account Creation), T1552 (Unsecured Credentials), T1552.001 (Credentials in Files), T1530 (Data from Cloud Storage), and T1550.001 (Application Access Token). No CVE is assigned; this is a structural governance gap, not a discrete exploited vulnerability. No public reports of active exploitation of dormant AI agent credentials have been documented as of this

publication date. Qualitative risk rating is High based on potential business impact if exploited; no CVSS score applies to structural control gaps.

Action Checklist

1. Step 1: Containment. Enumerate all AI agent service accounts, API keys, and OAuth tokens across Salesforce, Snowflake, GitHub, Gong, Slack, and any MCP server integrations. Immediately revoke credentials for agents with zero usage since creation. Revocation should be treated as an emergency IAM action, not a scheduled task.
2. Step 2: Detection. Query your IdP and SaaS admin consoles for service accounts and OAuth tokens with no login or API call activity in the past 30, 60, and 90 days. Cross-reference against known AI agent deployments. Enable audit logging (NIST AU-2, CIS 8.2) on all service accounts if not already active. Flag any token with scopes exceeding the documented agent function.
3. Step 3: Eradication. Apply least privilege scoping to all surviving agent credentials per NIST AC-6. Rotate all long-lived API keys on a defined schedule. Replace static API keys with short-lived tokens where the platform supports it. For platforms that do not support short-lived tokens (e.g., legacy Salesforce integrations), enforce mandatory rotation on a 90-day cycle and implement token revocation monitoring. For MCP server integrations, review SSRF mitigations and restrict outbound agent request destinations via allowlist.
4. Step 4: Recovery. After revocation and rotation, validate that AI agent workflows dependent on revoked credentials are accounted for, either re-issued under proper IAM controls or formally decommissioned. Establish a usage baseline for all reissued agent credentials and configure anomaly alerting. Confirm MCP server sandbox boundaries are enforced.
5. Step 5: Post-Incident. Formalize an AI agent registration and lifecycle policy requiring IT/IAM approval before deployment, mandatory credential scoping, and periodic access reviews aligned with NIST AC-2 (Account Management). Conduct a shadow AI discovery exercise quarterly. Map all agent identities into your privileged access management program alongside human accounts.

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate immediately to CISO and legal/compliance if any dormant AI agent credential shows evidence of unexpected API calls in SaaS audit logs (Salesforce Setup Audit Trail, Snowflake ACCESS_HISTORY, GitHub Audit Log) that cannot be attributed to authorized agent activity — this indicates credential compromise and may trigger breach notification obligations under GDPR, CCPA, or SOC 2 Trust Service Criteria if the accessed data includes PII or customer records in Salesforce, Snowflake, or Gong.

Recovery Notes	After credential revocation and reissuance, monitor all reissued AI agent service accounts daily for the first 30 days against their documented usage baselines — specifically watch for API call volume spikes, scope creep (new permissions requested by the agent runtime), and any calls to Snowflake schemas, Salesforce objects, GitHub repositories, or Gong recordings outside the agent's declared function. Confirm MCP server outbound allowlists remain enforced by reviewing iptables or Windows Firewall rule state weekly during the recovery window, as configuration drift or container redeployments can silently remove egress controls. Retain the full pre-remediation dormant credential inventory and SaaS audit log exports for a minimum of 12 months to support any future regulatory inquiry or forensic investigation into whether the dormant credentials were accessed prior to discovery.
Forensic Artifacts	SaaS OAuth token last-used timestamps and scope grants: Export from Salesforce Setup > Connected Apps OAuth Usage, GitHub organization audit log filtered on 'action:oauth_access', and Slack /api/apps.connections.list — these show whether any 'zero-usage' credential was actually accessed by an unauthorized party before revocation and are the primary indicator of compromise for this threat. | Snowflake ACCOUNT_USAGE.ACCESS_HISTORY and QUERY_HISTORY for AI agent service accounts: Reveals whether a dormant Snowflake service account credential was used to query sensitive tables (e.g., customer PII, financial data) without appearing in application-layer logs — critical for assessing data exfiltration scope. | MCP server outbound HTTP request logs and process environment variables: Log files at the MCP server application layer (typically JSON-formatted request/response logs) combined with a pre-containment capture of running process environment variables (`/proc//environ` on Linux) reveal whether long-lived API keys were embedded in agent runtime memory and whether SSRF was used to call internal metadata services or pivot to other SaaS endpoints. | IdP (Okta/Azure AD) service account authentication logs filtered on client_credentials grant type: Shows all token issuances for AI agent service principals with timestamps, source IPs, and requested scopes — identifies whether an attacker obtained and used a credential from an external IP or unusual ASN that would not match normal agent infrastructure. | GitHub repository event logs and deploy key audit records for agent-associated accounts: `GET /repos/{owner}/{repo}/events` and `GET /orgs/{org}/audit-log?phrase=action:repo` filtered to agent service account actor — reveals any unauthorized code commits, secrets scanning bypasses, or workflow modifications that a compromised GitHub agent token could have enabled.

Per-Action IR Details

Step 1: Containment — Enumerate all AI agent service accounts, API keys, and OAuth tokens across Salesforce, Snowflake, GitHub, Gong, Slack, and any MCP server integrations. Immediately revoke credentials for agents with zero usage since creation. Revocation should be treated as an emergency IAM action, not a scheduled task.

NIST Phase: Containment

Reference: NIST 800-61r3 §3.3 — Containment Strategy: Isolate the threat surface by terminating dormant privileged access paths before an attacker can leverage them; credential revocation for zero-activity AI agent tokens constitutes active containment of an exposed attack surface.

Controls: NIST AC-2 (Account Management) — requires identification and removal of accounts no longer needed, directly governing revocation of AI agent service accounts with zero usage, NIST AC-12 (Session Termination) — mandates termination of sessions under defined conditions; dormant AI agent OAuth tokens that have never been invoked meet the organization-defined trigger for forced termination, CIS 6.2 (Establish an Access Revoking Process) — requires a documented process for revoking access through disabling accounts and removing credentials; directly governs emergency revocation of AI agent API keys across Salesforce, Snowflake, GitHub, Gong, Slack, and MCP

integrations

Compensating: For teams without a centralized IAM platform: (1) Salesforce — run 'Setup > Connected Apps OAuth Usage' report filtered to last-used date 'never'; export via Salesforce CLI: `sfdx force:data:soql:query -q "SELECT Id, Name, LastUsedDate FROM ConnectedApplication WHERE LastUsedDate = null"`. (2) GitHub — use `gh api /orgs/{org}/installations` and cross-reference with GitHub Audit Log export filtered on 'action:token.create'. (3) Snowflake — query `SNOWFLAKE.ACCOUNT_USAGE.ACCESS_HISTORY` where `QUERY_START_TIME IS NULL` for service accounts. (4) Slack — pull app token list via `GET /api/apps.connections.list` using admin token and check last_used fields. (5) Consolidate into a shared spreadsheet; revoke via each platform's API or admin console immediately for any credential with null or pre-baseline last-used timestamp.

Evidence: BEFORE revoking any credential: (1) Export the full current token inventory from each platform including token ID, creation date, last-used timestamp, granted scopes, and associated service account name — this establishes the dormant access baseline and is destroyed the moment revocation occurs. (2) Capture active OAuth session state from Salesforce Setup Audit Trail (last 180 days), GitHub Audit Log (streaming export), Snowflake ACCOUNT_USAGE.LOGIN_HISTORY, Gong API activity logs, and Slack audit logs via `POST /api/audit.logs.query`. (3) For MCP server integrations, capture outbound HTTP request logs and any agent-generated API call logs before revoking — these may show whether a dormant credential was silently used by an attacker already. Volatile state lost on revocation: active bearer tokens in memory on MCP server processes; grab process-level environment variables via `cat /proc//environ` on Linux MCP hosts or `(Get-Process -Id ).StartInfo.EnvironmentVariables` on Windows before killing any agent process.

Step 2: Detection — Query your IdP and SaaS admin consoles for service accounts and OAuth tokens with no login or API call activity in the past 30, 60, and 90 days. Cross-reference against known AI agent deployments. Enable audit logging (NIST AU-2, CIS 8.2) on all service accounts if not already active. Flag any token with scopes exceeding the documented agent function.

NIST Phase: Detection Analysis

Reference: NIST 800-61r3 §3.2 — Detection and Analysis: Correlate IdP and SaaS audit telemetry to identify dormant AI agent credentials and over-scoped tokens that constitute the unmonitored privileged access surface described in the Token Security Agentic Pulse dataset.

Controls: NIST AU-2 (Event Logging) — requires identification of event types the system is capable of logging; directly governs enabling service account activity logging across Salesforce, Snowflake, GitHub, Gong, Slack, and MCP integrations where it may be disabled by default, NIST AU-6 (Audit Record Review, Analysis, And Reporting) — requires periodic review of audit records for indications of inappropriate activity; governs the 30/60/90-day activity query and scope-excess flagging for AI agent tokens, NIST AC-2 (Account Management) — requires monitoring accounts for atypical usage and reviewing accounts for compliance; governs cross-referencing dormant service accounts against known AI agent deployment registry, CIS 8.2 (Collect Audit Logs) — requires enabling logging across enterprise assets; directly cited in the step and governs activation of SaaS audit logging for AI agent service accounts not yet covered

Compensating: Without a SIEM: (1) Snowflake — `SELECT USER_NAME, LAST_SUCCESS_LOGIN, HAS_PASSWORD FROM SNOWFLAKE.ACCOUNT_USAGE.USERS WHERE LAST_SUCCESS_LOGIN System Overview > OAuth Connected Apps, sort by Last Used. (4) Slack — `GET /api/apps.connections.list` admin API; flag apps with scopes including `files:read`, `channels:history`, or `admin` beyond documented function. (5) Build a simple Python script using each platform's REST API to consolidate results into a CSV flagged by days-since-last-use and scope-vs-documented-function delta.

Evidence: Capture before enabling new logging (enabling logging may alter system state and overwrite pre-existing gaps in the audit trail): (1) Export current audit log configuration state from each SaaS platform — screenshot or API export of which event types are currently enabled — so you can document the detection gap that existed before remediation. (2) Pull all existing IdP (Okta, Azure AD, Ping) logs for service account authentications filtering on client_credential and service_account grant types for the maximum available retention window. (3) From Snowflake ACCOUNT_USAGE.QUERY_HISTORY, extract all queries executed by service accounts matching AI agent naming conventions in the past 90 days — this establishes whether 'zero usage' is true or merely unreported. (4) For GitHub, export the full organization audit log before any changes: `gh api /orgs/{org}/audit-log --paginate > gh_audit_baseline.json`. Volatile artifact: any in-flight OAuth token exchange currently cached in IdP session store —

capture IdP session tables or active token cache exports before logging enablement flushes transient records.

Step 3: Eradication — Apply least privilege scoping to all surviving agent credentials per NIST AC-6. Rotate all long-lived API keys on a defined schedule. Replace static API keys with short-lived tokens where the platform supports it. For MCP server integrations, review SSRF mitigations and restrict outbound agent request destinations via allowlist.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication: Eliminate the conditions that allowed dormant over-privileged AI agent credentials to persist by removing excessive scopes, rotating static secrets, and closing the MCP server SSRF pathway through which an attacker could use a compromised agent credential to pivot internally.

Controls: NIST AC-6 (Least Privilege) — directly cited in the step; mandates that AI agent OAuth token scopes be reduced to only the permissions required for the documented agent function across Salesforce, Snowflake, GitHub, Gong, and Slack, NIST AC-4 (Information Flow Enforcement) — governs restricting outbound agent request destinations via allowlist on MCP servers, enforcing approved authorization for information flow between the agent runtime and external service endpoints, CIS 4.4 (Implement and Manage a Firewall on Servers) — governs implementation of host-based or network firewall rules on MCP server hosts to enforce the outbound allowlist and block SSRF-exploitable egress paths

Compensating: Without enterprise PAM: (1) Scope reduction — for GitHub, use fine-grained PATs scoped to specific repositories and permissions only; revoke classic tokens entirely via `gh auth token --delete`. For Snowflake, create role-specific service accounts with GRANT on only the required schema/table rather than ACCOUNTADMIN or SYSADMIN. (2) Short-lived tokens — Salesforce supports Connected App certificate-based OAuth 2.0 JWT bearer flow issuing tokens valid for 15 minutes; configure this in lieu of long-lived refresh tokens. (3) MCP SSRF mitigation — on Linux MCP server hosts, add iptables egress rules: `iptables -A OUTPUT -m owner --uid-owner -d -j ACCEPT; iptables -A OUTPUT -m owner --uid-owner -j DROP`; on Windows, use Windows Firewall outbound rules scoped to the agent service account SID.

Evidence: BEFORE rotating any credential or modifying MCP server network rules: (1) Capture the current scope manifest for every surviving AI agent credential — exact OAuth scopes granted in Salesforce Connected App settings, GitHub PAT permissions JSON, Snowflake GRANT history via `SHOW GRANTS TO USER ` — this is destroyed when scopes are reduced. (2) On MCP server hosts, capture current outbound network connections: `ss -tunp` or `netstat -ano` plus `lsof -i` on Linux; `Get-NetTCPConnection` on Windows — document any active or recently established connections to internal resources that would indicate SSRF exploitation already occurred. (3) Pull Snowflake ACCOUNT_USAGE.QUERY_HISTORY for the agent user filtered to the past 30 days before credential rotation invalidates the attribution chain. (4) For GitHub, export all repository events attributed to the agent token via `GET /repos/{owner}/{repo}/events` before rotation breaks the audit linkage. Volatile artifact: MCP server process memory may contain cached bearer tokens for downstream SaaS APIs — run `strings /proc//mem` or use `procdump` on Windows before restarting the MCP service post-allowlist deployment.

Step 4: Recovery — After revocation and rotation, validate that AI agent workflows dependent on revoked credentials are accounted for — either re-issued under proper IAM controls or formally decommissioned. Establish a usage baseline for all reissued agent credentials and configure anomaly alerting. Confirm MCP server sandbox boundaries are enforced.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery: Restore authorized AI agent workflows under proper IAM governance, verify that no revoked credential has been reused or re-activated outside the sanctioned reissuance process, and establish monitoring baselines to detect future drift from the governed state.

Controls: NIST AC-2 (Account Management) — governs reissuance of AI agent service accounts under formal IAM controls including documented approval, defined account type, and usage monitoring as part of the recovery workflow, NIST AU-12 (Audit Record Generation) — governs configuration of audit record generation for all reissued AI agent credentials across Salesforce, Snowflake, GitHub, Gong, Slack, and MCP integrations to support the anomaly alerting baseline, CIS 1.1 (Establish and Maintain Detailed Enterprise Asset Inventory) — governs formal registration of reissued AI agent identities into the enterprise asset inventory so they are no longer shadow assets

Compensating: Without enterprise UEBA or SIEM for anomaly alerting: (1) Snowflake — create a scheduled task that queries ACCOUNT_USAGE.QUERY_HISTORY daily and alerts (email via Snowflake notification integration) if a reissued agent service account executes queries outside its documented schema/table scope. (2) GitHub — configure repository webhooks on events `push`, `create`, and `delete` filtered to the agent service account and pipe to a Slack webhook for real-time alerting on unexpected repository writes. (3) Salesforce — enable Event Monitoring (available in Enterprise/Unlimited) and configure a Transaction Security Policy that fires on API calls from the agent Connected App exceeding defined hourly volume thresholds. (4) For MCP server sandbox verification — run a manual SSRF probe from the agent process context: `curl -v http://169.254.169.254/latest/meta-data/` (AWS IMDSv1 check) to confirm cloud metadata endpoint is blocked; if it responds, the sandbox boundary is not enforced.

Evidence: BEFORE re-issuing any credential under new IAM controls: (1) Confirm the revocation of all prior credentials is complete and auditable — pull a final revocation confirmation receipt from each platform (Salesforce revocation timestamp from Setup Audit Trail, GitHub token deletion confirmation from audit log, Snowflake `SHOW USERS` confirming account is DISABLED). (2) Document the workflow inventory — a complete list of AI agent functions that were disrupted by revocation — to ensure re-issuance is scoped only to validated, business-justified workflows and no shadow agent is inadvertently re-authorized. (3) Capture the MCP server sandbox enforcement state before validating: current iptables/Windows Firewall ruleset export, and a network capture (`tcpdump -i any -w mcp_baseline.pcap` for 10 minutes) of normal MCP server egress to establish what legitimate outbound traffic looks like post-hardening. This pcap baseline is volatile and will not reflect the hardened state if captured after further configuration changes.

Step 5: Post-Incident — Formalize an AI agent registration and lifecycle policy requiring IT/IAM approval before deployment, mandatory credential scoping, and periodic access reviews aligned with NIST AC-2 (Account Management). Conduct a shadow AI discovery exercise quarterly. Map all agent identities into your privileged access management program alongside human accounts.

NIST Phase: Post Incident

Reference: NIST 800-61r3 §4 — Post-Incident Activity: Translate lessons learned from the dormant AI agent credential exposure into durable policy, detection capability improvements, and a recurring shadow AI discovery process that prevents the 65.4% zero-usage agent population from re-accumulating outside IAM governance.

Controls: NIST AC-2 (Account Management) — directly cited in the step; mandates the formal account lifecycle policy for AI agent identities including approval, scoping, periodic review, and deprovisioning that this post-incident action is designed to implement, NIST AC-1 (Policy And Procedures) — governs development and dissemination of the AI agent registration and lifecycle policy as a formal access control policy document, CIS 5.1 (Establish and Maintain an Inventory of Accounts) — requires maintaining an inventory of all accounts including service accounts; governs mapping AI agent identities into the PAM program so they appear in the same account inventory as human privileged accounts, CIS 6.1 (Establish an Access Granting Process) — requires a documented, preferably automated process for granting access; governs the IT/IAM approval gate before new AI agent deployment that this policy must encode, CIS 7.1 (Establish and Maintain a Vulnerability Management Process) — governs the quarterly shadow AI discovery exercise as a recurring governance process to identify unregistered agent deployments before they accumulate dormant privileged credentials

Compensating: For organizations without a commercial PAM tool: (1) Shadow AI discovery — quarterly, run a cross-platform service account enumeration script hitting Salesforce Connected Apps API, GitHub `/orgs/{org}/installations`, Snowflake `SHOW USERS`, Slack `/api/apps.connections.list`, and Gong API client list; diff the output against the registered AI agent inventory in your CMDB or a version-controlled YAML file in a private GitHub repo. (2) Policy enforcement gate — create a GitHub repository PR-based approval workflow: all new AI agent deployments require a `agent-registration.yaml` PR approved by the IAM team before credentials are issued; use GitHub branch protection rules to enforce this. (3) PAM integration — if no PAM tool exists, add AI agent service accounts to CyberArk Community Edition or HashiCorp Vault (free tier) for secret lifecycle management; Vault's dynamic secrets engine for AWS, GitHub, and database backends directly addresses the static API key problem by issuing short-lived credentials per-request.

Evidence: Post-incident documentation artifacts to preserve for lessons learned and future audit: (1) The full dormant credential inventory captured during Step 1 — this is the baseline evidence of the shadow AI access surface that existed before containment; retain for at least 12 months or per your AU-11 retention policy. (2) The scope-excess

findings from Step 2 — all tokens flagged with permissions exceeding documented agent function — to inform the mandatory scoping requirements in the new lifecycle policy. (3) Timeline reconstruction of when each AI agent credential was created versus when IT/IAM was notified (likely never for shadow agents) — this gap quantifies the governance failure and justifies the approval gate. (4) MCP server SSRF probe results from Step 4 recovery validation — document as evidence that sandbox boundaries were verified post-hardening. These artifacts collectively constitute the incident record required under NIST IR-4 (Incident Handling) for after-action review and should be stored in your case management system with access restricted to IR team and IAM leadership.

Detection Guidance

Query identity provider logs and SaaS admin audit trails for OAuth tokens and service accounts with zero activity since issuance, filter for accounts with 'created' timestamps older than 30 days and last_used = null or never. In GitHub: audit Settings > Developer Settings > OAuth Apps and GitHub Apps for unused installations. In Salesforce: review Connected Apps and Named Credentials for inactive OAuth grants. In Snowflake: query SNOWFLAKE.ACCOUNT_USAGE.LOGIN_HISTORY and QUERY_HISTORY for service accounts with no recent activity. In Slack: admin audit API event type 'app_installed' cross-referenced against app activity logs. Alert on T1078.004 indicators: API authentication from a service account with no prior usage baseline, token use outside business hours from an account with no human owner on record, and permission escalation requests (T1098) originating from an agent identity. Flag any AI agent credential accessing data stores (T1530) that is not documented in the AI agent registry or does not match a known approved workflow. For MCP integrations, monitor outbound HTTP requests from agent processes for SSRF patterns: requests to internal RFC-1918 addresses, cloud metadata endpoints (169.254.169.254), or unexpected external domains.

Framework Mappings

MITRE-ATTACK

- **T1078.004** — Cloud Accounts
- **T1098** — Account Manipulation
- **T1136.003** — Cloud Account
- **T1552** — Unsecured Credentials
- **T1552.001** — Credentials In Files
- **T1078** — Valid Accounts
- **T1530** — Data from Cloud Storage
- **T1550.001** — Application Access Token

NIST-800-53R5

- **AC-2** — Account Management
- **AC-6** — Least Privilege
- **IA-2** — Identification and Authentication (Organizational Users)
- **IA-5** — Authenticator Management
- **AC-3** — Access Enforcement
- **SI-4** — System Monitoring
- **IR-5** — Incident Monitoring

OWASP-TOP10-2021

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

CIS-V8

- **3.3** — Configure Data Access Control Lists
- **6.1** — Establish an Access Granting Process
- **6.2** — Establish an Access Revoking Process
- **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
- **8.2** — Collect Audit Logs

SOC2-TSC

- **CC6.1** — The entity implements logical access security software, infrastructure, and architectures over protected information assets

HIPAA-SECURITY

- **164.312(a)(1)** — Access Control
- **164.308(a)(5)(ii)(D)** — Password Management
- **164.312(d)** — Person or Entity Authentication
- **164.308(a)(6)(ii)** — Response and Reporting

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities

NIST-CSF-2

- **RS.CO-03** — Recovery activities and progress communicated
- **DE.CM-01** — Networks and network services are monitored
- **DE.AE-08** — Incidents are declared when adverse events meet the defined incident criteria

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1078.004	Cloud Accounts	Defense-Evasion
T1098	Account Manipulation	Persistence
T1136.003	Cloud Account	Persistence
T1552	Unsecured Credentials	Credential-Access

Technique ID	Technique Name	Tactic
T1552.001	Credentials In Files	Credential-Access
T1078	Valid Accounts	Defense-Evasion
T1530	Data from Cloud Storage	Collection
T1550.001	Application Access Token	Defense-Evasion

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/06/forget-data-leakage-shadow-ais-re...	T3
Overview · forcedotcom/salesforcedx-vscode - GitHub	https://github.com/forcedotcom/salesforcedx-vscode/security	T3
Security Advisory: SSRF, Indirect Prompt Injection, and sandbox ...	https://github.com/modelcontextprotocol/servers/issues/3662	T3
Vulnerability report for spelak-salesforce/codecoverage - Snyk	https://snyk.io/test/github/spelak-salesforce/codecoverage	T3
Security Advisories - Salesforce Help	https://help.salesforce.com/s/articleView?id=001119935&language...	T3

Generated 2026-06-19 13:53 UTC by TJS Security Command Center