INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-07-02 14:41 UTC

Agentic AI Identity Governance Gap: Autonomous Principals Accumulating Enterprise Access Without IGA Controls

GOVERNANCE | HIGH | CVSS 7.5

SCC Item ID	SCC-GOV-2026-0089
Type	Governance
Severity	HIGH
CVSS Base Score	7.5
Affected Products	Workday, SAP SuccessFactors, ServiceNow HR, Microsoft Active Directory, Azure AD, LangChain, AutoGen, AWS Bedrock Agents, Terraform
Published	2026-07-02T07:30:00
Discovery Source	Rss

Executive Summary

Enterprise AI agents deployed through frameworks such as LangChain, AutoGen, and AWS Bedrock Agents are accumulating privileged access across identity systems including Microsoft Active Directory, Azure AD, Workday, and ServiceNow without triggering standard Identity Governance and Administration controls. Because IGA platforms assume every identity maps to a human HR record, agentic principals bypass provisioning reviews, access certifications, and offboarding workflows entirely, leaving orphaned, over-privileged service accounts and API keys that persist indefinitely. A documented exploitation case in ServiceNow, reported by AppOmni and covered by Silverfort and Dark Reading, demonstrates that this governance gap is now actively exploitable for agent hijacking and lateral movement across enterprise environments.

Technical Analysis

Enterprise IGA frameworks enforce identity lifecycle controls, joiner, mover, leaver, through HR system signals (Workday, SAP SuccessFactors, ServiceNow HR). AI agents provisioned via LangChain, AutoGen, AWS Bedrock Agents, or Terraform-managed service accounts satisfy none of the preconditions: no HR record, no manager relationship, no departure trigger. This causes three IGA control points to fail silently. First, provisioning: agents receive service accounts or API keys without entitlement reviews, accumulating excessive permissions (CWE-250: Execution with Unnecessary Privileges; CWE-269: Improper Privilege Management). Second, access certification: periodic access reviews skip agentic identities because no HR signal associates

them with a human reviewer. Third, offboarding: credentials persist after the underlying workflow is deprecated (CWE-522: Insufficiently Protected Credentials; CWE-732: Incorrect Permission Assignment for Critical Resource). The AppOmni 'BodySnatcher' research documents a concrete ServiceNow exploitation path where agentic AI access was leveraged for agent hijacking and lateral movement (MITRE T1548: Abuse Elevation Control Mechanism; T1078: Valid Accounts; T1078.004: Cloud Accounts; T1098.001: Account Manipulation, Additional Cloud Credentials; T1528: Steal Application Access Token; T1550.001: Use Alternate Authentication Material, Application Access Token; T1136.003: Create Account, Cloud Account; T1552.001: Unsecured Credentials, Credentials in Files; T1538: Cloud Service Dashboard). No CVE ID is assigned to this governance issue; the CVSS 7.5 High estimate is an editorial assessment based on privilege escalation and lateral movement potential, not an NVD-assigned score. No vendor patch exists, remediation requires IGA process and policy changes.

Action Checklist

1. Step 1: Containment, Enumerate all service accounts, API keys, and non-human identities associated with AI agent frameworks (LangChain, AutoGen, AWS Bedrock Agents, Terraform-provisioned principals) in Active Directory, Azure AD, and connected SaaS platforms. Flag any account with no associated HR record in Workday, SAP SuccessFactors, or ServiceNow HR as uncontrolled agentic identity. Suspend or scope-limit accounts that cannot be attributed to an active, reviewed workflow. Apply NIST AC-6 (Least Privilege) and NIST AC-2 (Account Management) immediately to unreviewed agentic accounts.
2. Step 2: Detection, Query Active Directory and Azure AD for service accounts created within the past 12 months with no manager attribute, no HR system correlation, and permissions exceeding read-only. In AWS, audit IAM roles and Bedrock Agent execution roles for wildcard resource policies or unused permission boundaries. In ServiceNow, review agentic integration user accounts per AppOmni 'BodySnatcher' guidance. Monitor for MITRE T1078 (Valid Accounts) and T1548 (Abuse Elevation Control Mechanism) behaviors: unexpected API calls from service account principals, cross-system lateral movement, and token theft patterns (T1528, T1550.001). Enable logging per NIST AU-2 (Event Logging) and AU-12 (Audit Record Generation) for all agentic identity authentication events. Apply local account monitoring and user account permissions analysis countermeasures.
3. Step 3: Eradication, Establish a non-human identity register distinct from HR-driven IGA workflows. Assign a human owner (accountable team or product owner) to every agentic identity. Enforce least-privilege entitlements per NIST AC-6 and CIS Control 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts). Rotate all API keys and service account credentials for agentic principals that have not been reviewed within the past 90 days, per credential rotation best practices. Apply CIS Control 6.2 (Establish an Access Revoking Process) to deprecated agent workflows. For ServiceNow environments, review AppOmni 'BodySnatcher' findings at https://appomni.com/ao-labs/bodysnatcher-agentic-ai-security-vulnerability-in-servicenow/ and apply vendor-recommended agentic integration hardening.
4. Step 4: Recovery, Validate that all agentic identities now appear in the non-human identity register with assigned owners, documented entitlement justifications, and scheduled review dates. Confirm that access certification campaigns in your IGA platform explicitly include non-human identity classes. Re-run privilege analysis on remediated accounts to verify least-privilege scope. Monitor agentic identity authentication logs for 30 days post-remediation for anomalous access patterns per NIST AU-6 (Audit Record Review, Analysis, and Reporting). Apply credential hardening controls to new agent provisioning pipelines.

5. Step 5: Post-Incident, This gap exposes a structural IGA architecture assumption: IGA platforms were not designed to govern non-human principals at scale. Engage IGA platform vendors (identity governance tooling integrated with AD, Azure AD, and HR systems) to evaluate non-human identity lifecycle support. Define a formal AI agent identity policy covering provisioning standards, maximum permission scope, mandatory review cadence, and automated offboarding triggers tied to workflow lifecycle events rather than HR records. Map new policy to NIST AC-2, AC-6, and CIS Control 8 (Audit Log Management) for ongoing governance. Reference NIST IR controls for detection and response to future agentic identity misuse.

IR / Forensic Enrichment

Triage Priority	URGENT
Escalation Criteria	Escalate immediately to CISO and legal/privacy counsel if forensic review of Azure AD audit logs, AWS CloudTrail, or ServiceNow transaction logs confirms that any uncontrolled agentic principal accessed, modified, or exfiltrated HR data (Workday, SAP SuccessFactors, ServiceNow HR) containing PII or PHI, as this likely triggers breach notification obligations under GDPR, CCPA, or HIPAA depending on jurisdiction and data classification.
Recovery Notes	Before declaring recovery complete, verify that every agentic identity in Active Directory, Azure AD, AWS IAM, and ServiceNow appears in the non-human identity register with a named human owner, a documented least-privilege entitlement justification, and a scheduled access review date that does not depend on an HR lifecycle event. Confirm that your IGA platform's next access certification campaign explicitly includes the non-human identity OU or account population created during remediation — if the IGA platform cannot include non-human principals in campaigns, implement a manual quarterly review process as a compensating control until vendor capability is confirmed. Monitor agentic identity authentication logs across all four platforms for 30 days post-remediation, specifically alerting on any service principal authentication outside its documented workflow schedule or from an unexpected source IP, which would indicate either re-compromise or an undiscovered agentic identity not captured during enumeration.

Forensic Artifacts	Azure AD Unified Audit Log — service principal sign-in events and OAuth consent grant records (`Add service principal`, `Consent to application` operations) for LangChain, AutoGen, and Bedrock-integrated app registrations: these are the primary artifacts showing when and how agentic identities were granted enterprise access without IGA review. | AWS CloudTrail management events — `bedrock:CreateAgent`, `bedrock:InvokeAgent`, `iam:AttachRolePolicy`, and `iam:AssumeRole` records for Bedrock Agent execution roles: these establish the full privilege accumulation timeline and any cross-account lateral movement by agentic principals operating with wildcard resource policies. | Active Directory Security Event Log — Event ID 4720 (user account created), 4728 (member added to Global Security Group), 4732 (member added to Local Security Group), and 4769 (Kerberos service ticket requested) filtered on service accounts in the agentic identity population: these document privilege escalation paths that IGA certification campaigns never reviewed. | ServiceNow transaction log (`sys_log` and `sys_session` tables) filtered on integration user accounts identified as agentic principals: per the AppOmni BodySnatcher research, these logs reveal what HR, ITSM, and configuration data the agentic user queried or modified, establishing data exposure scope for regulatory assessment. | Terraform state files and CI/CD pipeline logs (e.g., GitHub Actions workflow logs, Azure DevOps pipeline audit logs) for infrastructure-as-code runs that provisioned agentic IAM roles or service principals: these are the upstream source-of-truth for how agentic identities were created outside IGA workflows and are required to identify all provisioned principals, including those that may have been destroyed and recreated without leaving a persistent AD or IAM record.

Per-Action IR Details

Step 1: Containment — Enumerate all service accounts, API keys, and non-human identities associated with AI agent frameworks (LangChain, AutoGen, AWS Bedrock Agents, Terraform-provisioned principals) in Active Directory, Azure AD, and connected SaaS platforms. Flag any account with no associated HR record in Workday, SAP SuccessFactors, or ServiceNow HR as uncontrolled agentic identity. Suspend or scope-limit accounts that cannot be attributed to an active, reviewed workflow. Apply NIST AC-6 (Least Privilege) and NIST AC-2 (Account Management) immediately to unreviewed agentic accounts.

NIST Phase: Containment

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

Controls: NIST AC-2 (Account Management), NIST AC-6 (Least Privilege), CIS 5.1 (Establish and Maintain an Inventory of Accounts)

Compensating: Run the following PowerShell against AD to enumerate service accounts with no manager attribute and no HR-correlated description field: `Get-ADUser -Filter {ServicePrincipalName -ne '$null'} -Properties Manager, Description, ServicePrincipalName | Where-Object {$_.Manager -eq $null} | Export-Csv orphan_spns.csv`. For Azure AD, use the free Azure CLI: `az ad sp list --all --query '[?!(owners)].{displayName:displayName, appId:appId, createdDateTime:createdDateTime}' -o table`. Cross-reference output against Workday or ServiceNow HR export (CSV) manually in a spreadsheet to identify accounts with no HR record match. For AWS, run `aws iam list-roles --query 'Roles[?contains(RoleName, \`bedrock\`) || contains(RoleName, \`agent\`) || contains(RoleName, \`langchain\`)]'` to surface Bedrock Agent execution roles.

Evidence: Before suspending or scope-limiting any agentic account, capture: (1) current Azure AD sign-in logs for each flagged service principal via `az monitor activity-log list` or the Azure Portal Audit Logs blade — export last 90 days of authentication events for each principal; (2) AD last-logon timestamps and SPN registrations (`Get-ADUser -Properties LastLogonDate, ServicePrincipalName`); (3) AWS CloudTrail `AssumeRole` and `InvokeAgent` events for Bedrock Agent execution roles covering the past 90 days (`aws cloudtrail lookup-events --lookup-attributes AttributeKey=EventName,AttributeValue=AssumeRole`); (4) ServiceNow integration user session logs before disabling — these are the only record of what the agentic principal accessed prior to suspension. Suspension irreversibly ends active sessions and destroys live token state.

Step 2: Detection — Query Active Directory and Azure AD for service accounts created within the past 12 months with no manager attribute, no HR system correlation, and permissions exceeding read-only. In AWS, audit IAM roles and Bedrock Agent execution roles for wildcard resource policies or unused permission boundaries. In ServiceNow, review agentic integration user accounts per AppOmni 'BodySnatcher' guidance. Monitor for MITRE T1078 (Valid Accounts) and T1548 (Abuse Elevation Control Mechanism) behaviors: unexpected API calls from service account principals, cross-system lateral movement, and token theft patterns (T1528, T1550.001). Enable logging per NIST AU-2 (Event Logging) and AU-12 (Audit Record Generation) for all agentic identity authentication events. Apply D3-LAM (Local Account Monitoring) and D3-UAP (User Account Permissions) countermeasures.

NIST Phase: Detection Analysis

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

Controls: NIST AU-2 (Event Logging), NIST AU-12 (Audit Record Generation), NIST AC-2 (Account Management)

Compensating: Deploy Sysmon with a config that captures Event ID 10 (ProcessAccess) and Event ID 25 (ProcessTampering) on hosts running LangChain or AutoGen agent processes. For AD, use this PowerShell to find service accounts created in the past 12 months with no manager and DirectoryRole or AdminCount set: `Get-ADUser -Filter {whenCreated -gt (Get-Date).AddDays(-365) -and AdminCount -eq 1} -Properties Manager, whenCreated, AdminCount | Where-Object {$_.Manager -eq $null}`. For AWS IAM wildcard policy detection, use Prowler (free, open-source): `prowler aws -c iam_policy_no_administrative_privileges`. For ServiceNow, manually audit integration user records under User Administration for accounts with roles such as `admin`, `itil`, or `x_*` that have no linked HR employee record.

Evidence: This is a detection step that does not itself alter live state, but queries that enumerate active sessions or running agent processes should be completed before any containment action on the same host. Capture before proceeding to eradication: (1) Azure AD Unified Audit Log entries for service principal OAuth token issuances (event operation `Add service principal`) and consent grants (`Consent to application`) — these record the initial access grants that IGA missed; (2) AWS CloudTrail records for `bedrock:InvokeAgent`, `bedrock:CreateAgent`, and IAM `AttachRolePolicy` events tied to agent execution roles; (3) ServiceNow transaction logs (`sys_log` table) filtered on integration user account names to establish what tables and records the agentic principal queried or modified; (4) Active Directory Security Event Log Event ID 4720 (account created) and 4728/4732 (member added to privileged group) for service accounts matching the enumeration query — these establish when the orphaned identity was created and what privileges were granted without IGA review.

Step 3: Eradication — Establish a non-human identity register distinct from HR-driven IGA workflows. Assign a human owner (accountable team or product owner) to every agentic identity. Enforce least-privilege entitlements per NIST AC-6 and CIS 5.4 (Restrict Administrator Privileges to Dedicated Administrator Accounts). Rotate all API keys and service account credentials for agentic principals that have not been reviewed within the past 90 days, per D3-CRO (Credential Rotation). Apply CIS 6.2 (Establish an Access Revoking Process) to deprecated agent workflows. For ServiceNow environments, review AppOmni 'BodySnatcher' findings at https://appomni.com/ao-labs/bodysnatcher-agentic-ai-security-vulnerability-in-servicenow/ and apply vendor-recommended agentic integration hardening.

NIST Phase: Eradication

Reference: NIST 800-61r3 §3.4 — Eradication

Controls: 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: Build the non-human identity register as a CSV or shared spreadsheet with columns: AccountName, Platform (AD/AzureAD/AWS/ServiceNow), CreatedDate, LastAuthenticated, AssignedOwner, EntitlementJustification, ReviewDate. For credential rotation of AWS Bedrock Agent execution role keys, use `aws iam create-access-key` followed by `aws iam delete-access-key` after confirming the new key is operational. For Azure AD app registration secrets, use `az ad app credential reset --id --years 1`. For AD service account passwords, use

`Set-ADAccountPassword` with a 25+ character random value from a password manager. Document every rotation in the register with a timestamp. For deprecated LangChain or AutoGen workflows, revoke the associated Azure AD application registration entirely using `az ad app delete --id ` rather than just disabling — this removes the principal from the directory entirely.

Evidence: Credential rotation and account revocation are irreversible actions that destroy live token state. Before rotating any agentic credential: (1) capture all active OAuth refresh tokens and access tokens issued to the agentic service principal from Azure AD sign-in logs — these cannot be recovered after rotation; (2) export AWS CloudTrail records for the specific IAM role or access key being rotated covering the full 90-day unreviewed window, confirming no active in-flight agent job depends on the credential; (3) for ServiceNow, export the full session history (`sys_session` table) of the integration user before disabling — the AppOmni BodySnatcher scenario involves the agentic user having accessed HR and ITSM data, and this log is the only artifact establishing data exposure scope; (4) snapshot the current IAM policy document attached to each AWS Bedrock Agent execution role before modifying (`aws iam get-role-policy` and `aws iam list-attached-role-policies`) — this is the authoritative record of what permissions existed during the unreviewed period and may be needed for regulatory reporting. Note: The AppOmni URL cited in the step is a search-retrieved reference — recommend human validation that the URL resolves to the current BodySnatcher advisory before relying on it for vendor hardening guidance.

Step 4: Recovery — Validate that all agentic identities now appear in the non-human identity register with assigned owners, documented entitlement justifications, and scheduled review dates. Confirm that access certification campaigns in your IGA platform explicitly include non-human identity classes. Re-run privilege analysis on remediated accounts to verify least-privilege scope. Monitor agentic identity authentication logs for 30 days post-remediation for anomalous access patterns per NIST AU-6 (Audit Record Review, Analysis, and Reporting). Apply D3-CH (Credential Hardening) controls to new agent provisioning pipelines.

NIST Phase: Recovery

Reference: NIST 800-61r3 §3.5 — Recovery

Controls: NIST AU-6 (Audit Record Review, Analysis, And Reporting), NIST AC-2 (Account Management), NIST AC-6 (Least Privilege), CIS 5.1 (Establish and Maintain an Inventory of Accounts)

Compensating: For the 30-day post-remediation monitoring window without a SIEM, configure Azure AD diagnostic settings to stream sign-in logs to a Log Analytics workspace (free tier covers this volume for most orgs) and set an alert rule on service principal sign-ins outside business hours or from unexpected IP ranges. For AWS, create a CloudWatch metric filter on CloudTrail for `bedrock:InvokeAgent` events from the remediated execution roles and alert on any invocation — legitimate agent workflows should only fire during known automation windows. For AD, schedule a weekly PowerShell job: `Get-ADUser -Filter {ServicePrincipalName -ne '$null'} -Properties LastLogonDate | Where-Object {$_.LastLogonDate -gt (Get-Date).AddDays(-1)} | Export-Csv weekly_svc_logins.csv` and diff against the non-human identity register to catch any new unregistered principals. For IGA campaign inclusion, produce a query against your IGA platform's identity store filtered on accounts where the `employeeType` or equivalent attribute is null or set to `service` — submit this as a custom population for the next certification cycle.

Evidence: Recovery validation does not alter live state in a way that destroys forensic artifacts, but the privilege re-analysis step (re-running access reviews on remediated accounts) should be preceded by a final export of the post-remediation entitlement state: (1) export the current Azure AD app role assignments for all remediated service principals (`az ad app show --id `) to establish a clean baseline for the 30-day monitoring window; (2) capture AWS IAM Access Analyzer findings on the remediated Bedrock Agent execution roles (`aws accessanalyzer list-findings`) immediately after policy changes — this documents that least-privilege was achieved and provides a comparison point if privilege creep re-emerges; (3) export the ServiceNow user role assignments for all remediated integration accounts (`sys_user_has_role` table) as the post-remediation entitlement baseline. These exports are the evidentiary record that eradication was complete before recovery was declared.

Step 5: Post-Incident — This gap exposes a structural IGA architecture assumption: IGA platforms were not designed to govern non-human principals at scale. Engage IGA platform vendors (identity governance tooling integrated with AD, Azure AD, and HR systems) to evaluate non-human identity lifecycle support. Define a formal AI agent identity policy covering provisioning standards, maximum permission scope, mandatory

review cadence, and automated offboarding triggers tied to workflow lifecycle events rather than HR records. Map new policy to NIST AC-2, AC-6, and CIS 7.1 (Establish and Maintain a Vulnerability Management Process) for ongoing governance. Reference NIST IR controls for detection and response to future agentic identity misuse.

NIST Phase: Post Incident

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

Controls: NIST AC-2 (Account Management), NIST AC-6 (Least Privilege), CIS 7.1 (Establish and Maintain a Vulnerability Management Process), CIS 7.2 (Establish and Maintain a Remediation Process)

Compensating: For teams that cannot engage enterprise IGA vendors immediately, implement a lightweight non-human identity lifecycle process using native tooling: (1) create a dedicated AD OU (e.g., `OU=AgenticIdentities,DC=corp,DC=com`) and apply a GPO that prevents agentic service accounts from being used for interactive logon; (2) enforce a Terraform naming convention and tagging standard for all agent-provisioned IAM roles (e.g., tag `ManagedBy=AgentFramework`, `ReviewDate=YYYY-MM-DD`) and use `aws tag:get-resources` in a monthly audit script to find roles missing the tag; (3) draft the AI agent identity policy using the NIST SP 800-53r5 AC-2 control parameters as a template — the `account types`, `account managers`, and `account review` parameters map directly to the policy elements required (provisioning standards, owner assignment, review cadence); (4) submit a feature request to your IGA vendor with a documented use case referencing this incident — vendor roadmap responses are discoverable in regulatory inquiries and demonstrate due diligence.

Evidence: Post-incident activity does not alter live system state, but the lessons-learned process should preserve: (1) the complete non-human identity enumeration artifacts from Steps 1–2 (AD export, Azure AD service principal list, AWS IAM role inventory, ServiceNow integration user list) as the authoritative record of the pre-remediation exposure surface — retain per your organization's incident record retention policy; (2) the before/after IAM policy snapshots from Step 3 to document what over-privileged entitlements existed during the unreviewed period; (3) the 30-day post-remediation authentication logs from Step 4 as evidence that no agentic re-compromise occurred during recovery; (4) a written timeline of when each agentic identity was created, when it was flagged, and when it was remediated — this timeline is the primary artifact for regulatory inquiry if agentic principals had access to PII stored in Workday, SAP SuccessFactors, or ServiceNow HR during the unreviewed window.

Detection Guidance

Detection requires extending standard identity monitoring to non-human principals explicitly. In Azure AD and Entra ID, query for service principals with no owner assignment, created without a corresponding Terraform or IaC change record, or holding roles beyond Reader/Contributor scope without documented justification. In AWS IAM, identify Bedrock Agent execution roles with wildcard resource policies (Resource: '*') or missing permission boundaries. In Active Directory, query for service accounts (userAccountControl flags indicating service account type) with no manager attribute and last password set more than 90 days ago. In ServiceNow, per AppOmni 'BodySnatcher' research, review integration user accounts granted elevated agentic workflow permissions and audit flow designer execution logs for unexpected cross-scope data access. Behavioral indicators aligned to MITRE techniques: service account principals authenticating outside defined working hours or from unexpected source IPs (T1078, T1078.004); API calls using application access tokens to access resources outside the agent's documented scope (T1528, T1550.001); new cloud account creation events from non-human principals (T1136.003); credential file access events from agent runtime processes (T1552.001). NIST AU-2 and AU-12 require that logging be enabled for these event types across all covered systems. Apply local account monitoring for on-premises AD service accounts and user account permissions analysis for cloud IAM roles. CIS Control 8.2 (Collect Audit Logs) should be verified as covering service principal and API key authentication events, not only human user sessions.

Framework Mappings

MITRE-ATTACK

- **T1136.003** — Cloud Account
- **T1078** — Valid Accounts
- **T1528** — Steal Application Access Token
- **T1098.001** — Additional Cloud Credentials
- **T1098** — Account Manipulation
- **T1552.001** — Credentials In Files
- **T1078.004** — Cloud Accounts
- **T1538** — Cloud Service Dashboard
- **T1550.001** — Application Access Token
- **T1548** — Abuse Elevation Control Mechanism

NIST-800-53R5

- **AC-2** — Account Management
- **AC-6** — Least Privilege
- **IA-2** — Identification and Authentication (Organizational Users)
- **IA-5** — Authenticator Management
- **CM-6** — Configuration Settings
- **AC-3** — Access Enforcement
- **AT-2** — Literacy Training and Awareness

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
- **3.3** — Configure Data Access Control Lists
- **6.3** — Require MFA for Externally-Exposed Applications
- **14.2** — Train Workforce Members to Recognize Social Engineering Attacks

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
- **CC6.3** — Authorizes, modifies, or removes access

ISO-27001-2022

- **A.8.8** — Management of technical vulnerabilities
- **A.5.23** — Information security for use of cloud services

NIST-CSF-2

- **DE.AE-08** — Incidents are declared when adverse events meet the defined incident criteria

MITRE ATT&CK Mapping

Technique ID	Technique Name	Tactic
T1136.003	Cloud Account	Persistence
T1078	Valid Accounts	Defense-Evasion
T1528	Steal Application Access Token	Credential-Access
T1098.001	Additional Cloud Credentials	Persistence
T1098	Account Manipulation	Persistence
T1552.001	Credentials In Files	Credential-Access
T1078.004	Cloud Accounts	Defense-Evasion
T1538	Cloud Service Dashboard	Discovery
T1550.001	Application Access Token	Defense-Evasion
T1548	Abuse Elevation Control Mechanism	Privilege-Escalation

Sources

Source	URL	Tier
Security News	https://thehackernews.com/2026/07/identity-lifecycle-management.html	T2
Agentic AI Security Vulnerability in ServiceNow Exposed - AppOmni	https://appomni.com/ao-labs/bodysnatcher-agentic-ai-security-vulner...	T3
The ServiceNow AI Vulnerability: What Went Wrong and ... - OpenA2A	https://opena2a.org/blogs/servicenow-ai-vulnerability	T3

Source	URL	Tier
Identity Security lessons from the ServiceNow vulnerability - Silverfort	https://www.silverfort.com/blog/agent-hijacking-lateral-movement-le...	T3
'Most Severe AI Vulnerability to Date' Hits ServiceNow - Dark Reading	https://www.darkreading.com/remote-workforce/ai-vulnerability-servi...	T2

Generated 2026-07-02 14:41 UTC by TJS Security Command Center