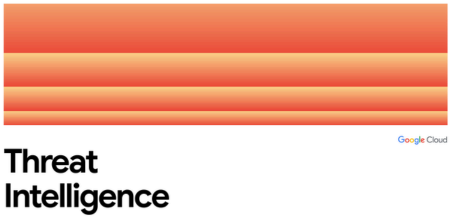
Cybercrime Observations from the Frontlines: UNC6040 Proactive Hardening Recommendations

Mandiant ⋮ ⋮ 9/29/2025



Mandiant Incident Response

Investigate, contain, and remediate security incidents.

Learn more

Written by: Omar ElAhdan, Matthew McWhirt, Michael Rudden, Aswad Robinson, Bhavesh Dhake, Laith Al

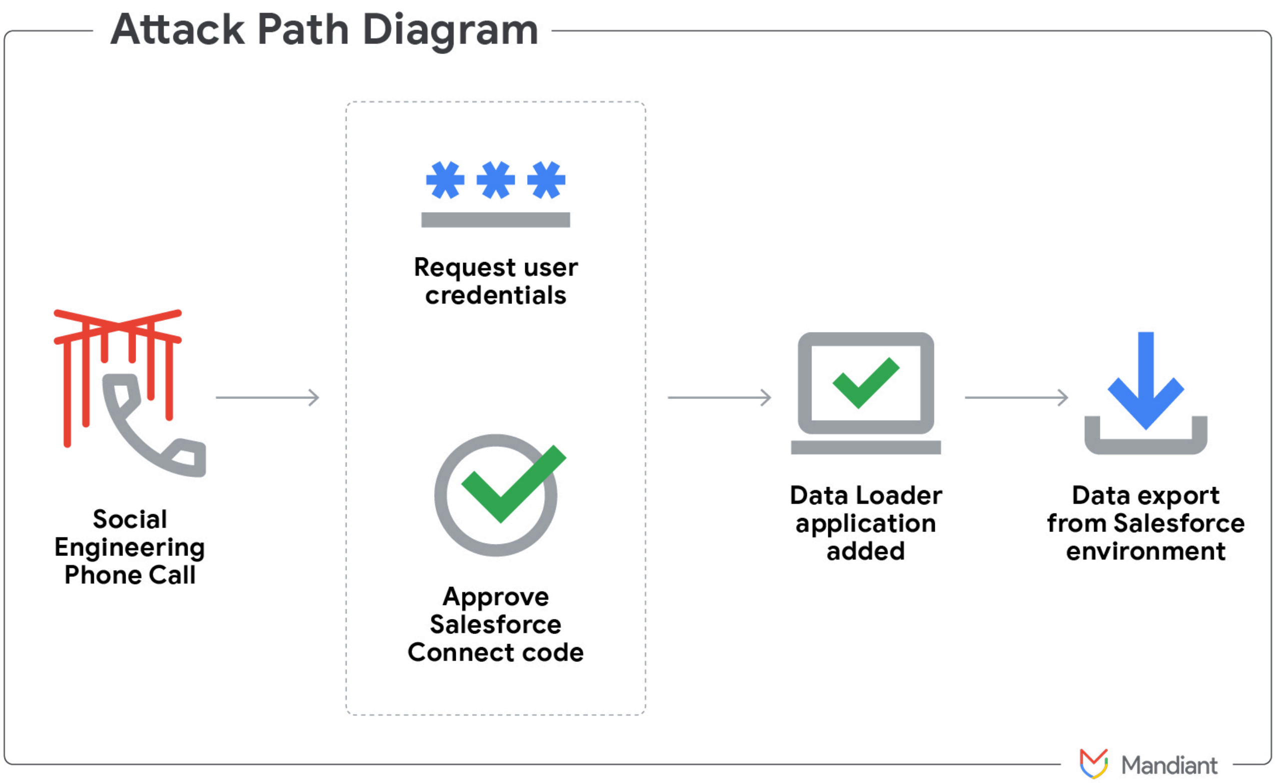
Background

Protecting software-as-a-service (SaaS) platforms and applications requires a comprehensive security strategy. Drawing from analysis of UNC6040’s specific attack methodologies, this guide presents a structured defensive framework encompassing proactive hardening measures, comprehensive logging protocols, and advanced detection capabilities. While emphasizing Salesforce-specific security recommendations, these strategies provide organizations with actionable approaches to safeguard their SaaS ecosystem against current threats.

Google Threat Intelligence Group (GTIG) is tracking UNC6040, a financially motivated threat cluster that specializes in voice phishing (vishing) campaigns specifically designed to compromise organizations' Salesforce instances for large-scale data theft and subsequent extortion. Over the past several months, UNC6040 has demonstrated repeated success in breaching networks by having its operators impersonate IT support personnel in convincing telephone-based social engineering engagements. This approach has proven particularly effective in tricking employees, often within English-speaking branches of multinational corporations, into actions that grant the attackers access or lead to the sharing of sensitive credentials, ultimately facilitating the theft of organization’s Salesforce data. In all observed cases, attackers relied on manipulating end users, not exploiting any vulnerability inherent to Salesforce.

A prevalent tactic in UNC6040's operations involves deceiving victims into authorizing a malicious connected app to their organization's Salesforce portal. This application is often a modified version of Salesforce’s Data Loader, not authorized by Salesforce. During a vishing call, the actor guides the victim to visit Salesforce's connected app setup page to approve a version of the Data Loader app with a name or branding that differs from the legitimate version. This step inadvertently grants UNC6040 significant capabilities to access, query, and exfiltrate sensitive information directly from the compromised Salesforce customer environments. This methodology of abusing Data Loader functionalities via malicious connected apps is consistent with recent observations detailed by Salesforce in their guidance on protecting Salesforce environments from such threats.

In some instances, extortion activities haven't been observed until several months after the initial UNC6040 intrusion activity, which could suggest that UNC6040 has partnered with a second threat actor that monetizes access to the stolen data. During these extortion attempts, the actor has claimed affiliation with the well-known hacking group ShinyHunters, likely as a method to increase pressure on their victims.


Attack Path Diagram
Social Engineering Phone Call
Request user credentials
Approve Salesforce Connect code
Data Loader application added
Data export from Salesforce environment
Mandiant

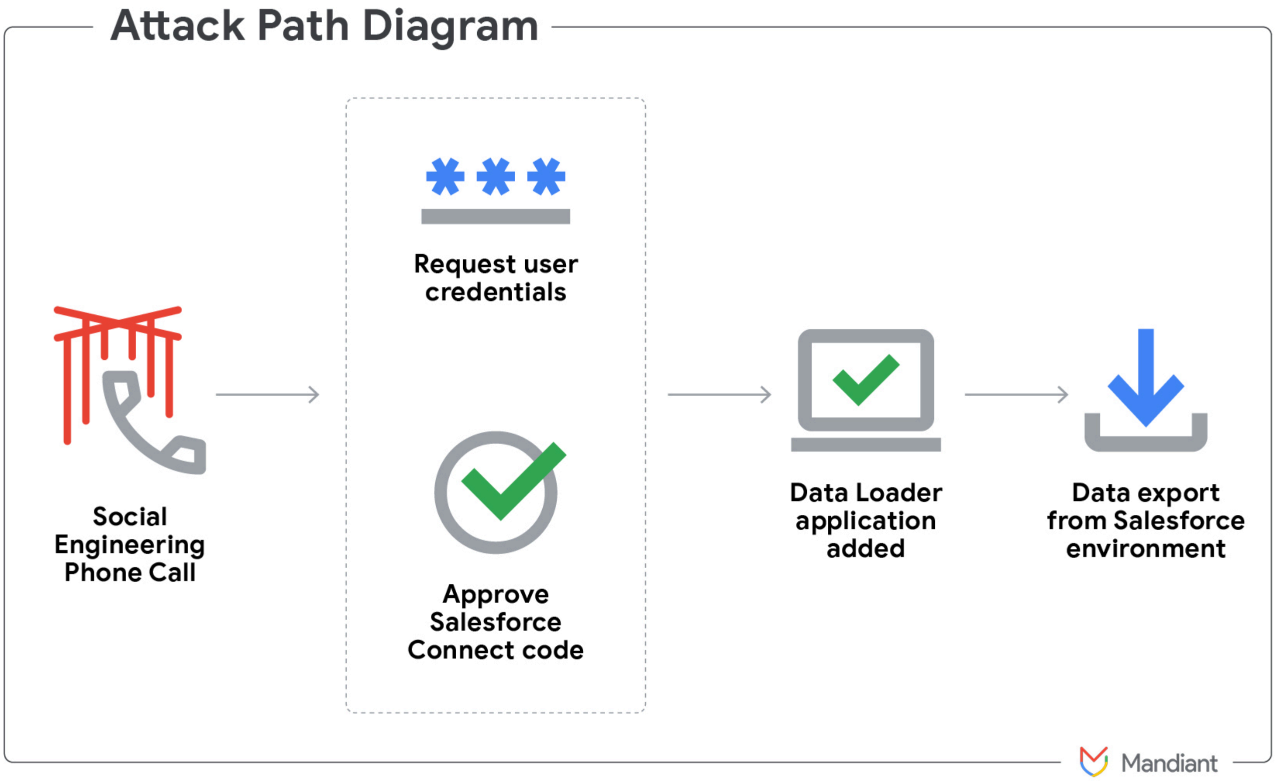

Attack Path Diagram
Social Engineering Phone Call
Request user credentials
Approve Salesforce Connect code
Data Loader application added
Data export from Salesforce environment
Mandiant

Figure 1: Data Loader attack flow

We have observed the following patterns in UNC6040 victimology:

- **Motive:** UNC6040 is a financially motivated threat cluster that accesses victim networks by vishing social engineering.
- **Focus:** Upon obtaining access, UNC6040 has been observed immediately exfiltrating data from the victim's Salesforce environment using Salesforce's Data Loader application. Following this initial data theft, UNC6040 was observed leveraging end-user credentials obtained through credential harvesting or vishing to move laterally through victim networks, accessing and exfiltrating data from the victim's accounts on other cloud platforms such as Okta and Microsoft 365.
- **Attacker infrastructure:** UNC6040 primarily used Mullvad VPN IP addresses to access and perform the data exfiltration on the victim's Salesforce environments and other services of the victim's network.

Proactive Hardening Recommendations

The following section provides prioritized recommendations to protect against tactics utilized by UNC6040. This section is broken down to the following categories:

1. Identity
    - Help Desk and End User Verification
    - Identity Validation and Protections
2. SaaS Applications

- SaaS Application (e.g., Salesforce) Hardening Measures

3. Logging and Detections

Note: While the following recommendations include strategies to protect SaaS applications, they also cover identity security controls and detections applicable at the Identity Provider (IdP) layer and security enhancements for existing processes, such as the help desk.

1. Identity

Positive Identity Verification

To protect against increasingly sophisticated social engineering and credential compromise attacks, organizations must adopt a robust, multilayered process for identity verification. This process moves beyond outdated, easily compromised methods and establishes a higher standard of assurance for all support requests, especially those involving account modifications (e.g., password resets or multi-factor authentication modifications).

Guiding Principles

- **Assume nothing:** Do not inherently trust the caller's stated identity. Verification is mandatory for all security-related requests.
- **Defense-in-depth:** Rely on a combination of verification methods. No single factor should be sufficient for high-risk actions.
- **Reject unsafe identifiers:** Avoid relying on publicly available or easily discoverable data. Information such as:
    - Date of birth
    - Last four digits of a Social Security number
    - High school names
    - Supervisor names

This data should not be used as primary verification factors, as it's often compromised through data breaches or obtainable via open source intelligence (OSINT).

Standard Verification Procedures

Live Video Identity Proofing (Primary Method)

This is the most reliable method for identifying callers. The help desk agent must:

1. Initiate a video call with the user
2. Require the user to present a valid corporate badge or government-issued photo ID (e.g., driver's license) on camera next to their face

3. Visually confirm that the person on the video call matches the photograph on the ID
4. Cross-reference the user's face with their photo in the internal corporate identity system
5. Verify that the name on the ID matches the name in the employee's corporate record

   Contingency for No Video:

   If a live video call is not possible, the user must provide a selfie showing their face, their photo ID, and a piece of paper with the current date and time written on it.

   Additionally, before proceeding with any request - help desk personnel must check the user's calendar for Out of Office (OOO) or vacation status. All requests from users who are marked as OOO should be presumptively denied until they have officially returned.

Out-of-Band (OOB) Verification (For High-Risk Requests)

For high-risk changes like multi-factor authentication (MFA) resets or password changes for privileged accounts, an additional OOB verification step is required after the initial ID proofing. This can include:

- **Call-back:** Placing a call to the user's registered phone number on file
- **Manager approval:** Sending a request for confirmation to the user's direct manager via a verified corporate communication channel

Special Handling for Third-Party Vendor Requests

Mandiant has observed incidents where attackers impersonate support personnel from third-party vendors to gain access. In these situations, the standard verification principals may not be applicable.

Under no circumstances should the Help Desk move forward with allowing access. The agent must halt the request and follow this procedure:

1. End the inbound call without providing any access or information
2. Independently contact the company's designated **account manager** for that vendor using trusted, on-file contact information
3. Require explicit verification from the account manager before proceeding with any request

Outreach to End Users

Mandiant has observed the threat actor UNC6040 targeting end-users who have elevated access to SaaS applications. Posing as vendors or support personnel, UNC6040 contacts these users and provides a malicious link. Once the user clicks the link and authenticates, the attacker gains access to the application to exfiltrate data.

To mitigate this threat, organizations should rigorously communicate to all end-users the importance of verifying any third-party requests. Verification procedures should include:

- Hanging up and calling the official account manager using a phone number on file
- Requiring the requester to submit a ticket through the official company support portal
- Asking for a valid ticket number that can be confirmed in the support console

Organizations should also provide a clear and accessible process for end-users to report suspicious communications and ensure this reporting mechanism is included in all security awareness outreach.

Salesforce has additional guidance that can be referenced.

Identity Protections

Since access to SaaS applications is typically managed by central identity providers (e.g., Entra ID, Okta), Mandiant recommends that organizations enforce unified identity security controls directly within these platforms.

Guiding Principles

Mandiant's approach focuses on the following core principles:

- **Authentication boundary** This principle establishes a foundational layer of trust based on network context. Access to sensitive resources should be confined within a defined boundary, primarily allowing connections from trusted corporate networks and VPNs to create a clear distinction between trusted and untrusted locations.
- **Defense-in-depth** This principle dictates that security cannot rely on a single control. Organizations should layer multiple security measures,such as strong authentication, device compliance checks, and session controls.
- **Identity detection and response** Organizations must continuously integrate real-time threat intelligence into access decisions. This ensures that if an identity is compromised or exhibits risky behavior, its access is automatically contained or blocked until the threat has been remediated.

Identity Security Controls

The following controls are essential for securing access to SaaS applications through a central identity provider.

Utilize Single Sign-On (SSO)

Ensure that all users accessing SaaS applications are accessing via a corporate-managed SSO provider (e.g., Microsoft Entra ID or Okta), rather than through platform-native accounts. A platform-native break glass account should be created and vaulted for use only in the case of an emergency.

In the event that SSO through a corporate-managed provider is not available, refer to the content specific to the applicable SaaS application (e.g., Salesforce) rather than Microsoft Entra ID or Okta.

Mandate Phishing-Resistant MFA

Phishing-resistant MFA must be enforced for all users accessing SaaS applications. This is a foundational requirement to defend against credential theft and account takeovers. Consider enforcing physical FIDO2 keys for accounts with privileged access. Ensure that no MFA bypasses exist in authentication policies tied to business critical applications.

For Microsoft Entra ID:

- General MFA Policy: Enforce MFA for all users with Conditional Access
- Passkey (FIDO2) Setup: Enable passkey and FIDO2 security keys

For Okta:

- Configure FIDO2 (WebAuthn): Set up the FIDO2 (WebAuthn) authenticator
- Enforce via Policy: Create an Authentication Policy to require strong authenticators

For Google Cloud Identity / Workspace:

- General MFA Policy: Deploy 2-Step Verification
- Security Key enforcement: Use a security key for 2-Step Verification

For Salesforce:

- MFA is required by default for local Salesforce accounts: Salesforce Multi-Factor Authentication FAQ
- Configure FIDO2 (WebAuthn): Register a Security Key as an Identity Verification Method for Salesforce Orgs

Enforce Device Trust and Compliance

Access to corporate applications must be limited to devices that are either domain-joined or verified as compliant with the organization's security standards. This policy ensures that a device meets a minimum security baseline before it can access sensitive data.

Key device posture checks should include:

- **Valid host certificate:** The device must present a valid, company-issued certificate
- **Approved operating system:** The endpoint must run an approved OS that meets current version and patch requirements
- **Active EDR agent:** The corporate Endpoint Detection and Response (EDR) solution must be installed, active, and reporting a healthy status

For Microsoft Entra ID:

- Device Compliance Policy: Require compliant devices for all users with Conditional Access

For Okta:

- Device Trust Overview: Configure Okta Device Trust for managed devices

For Google Cloud Identity / Workspace:

- Context-Aware Access: Overview of Context-Aware Access
- Endpoint Verification: Monitor and gather details about devices with Endpoint Verification

Automate Response to Identity Threats

Mandiant recommends that organizations implement dynamic authentication policies that respond to threats in real time. By integrating identity threat intelligence feeds—from both native platform services and third-party solutions—into the authentication process, organizations can automatically block or challenge access when an identity is compromised or exhibits risky behavior.

This approach primarily evaluates two categories of risk:

- **Risky sign-ins:** The probability that an authentication request is illegitimate due to factors like atypical travel, a malware-linked IP address, or password spray activity
- **Risky users:** The probability that a user's credential has been compromised or leaked online

Based on the detected risk level, Mandiant recommends that organizations apply a tiered approach to remediation.

Recommended Risk-Based Actions

- **For high-risk events:** Organizations should apply the most stringent security controls. This includes blocking access entirely.
- **For medium-risk events:** Access should be granted only after a significant step-up in verification. This typically means requiring proof of both the user's identity (via strong MFA) and the device's integrity (by verifying its compliance and security posture).
- **For low-risk events:** Organizations should still require a step-up authentication challenge, such as standard MFA, to ensure the legitimacy of the session and mitigate low-fidelity threats.

For Microsoft Entra ID:

- **Overview**
- Configuration

For Okta:

- Behavior detection
- Risk-based policies

For Google Cloud Identity / Workspace:

- **Access context manager overview:** Use Context-Aware Access to create granular, risk-based access policies

For Salesforce Shield:

- **Overview**
- **Event monitoring:** Provides detailed logs of user actions—such as data access, record modifications, and login origins—and allows these logs to be exported for external analysis
- **Transaction security policies:** Monitors for specific user activities, such as large data downloads, and can be configured to automatically trigger alerts or block the action when it occurs

2. SaaS Applications

Salesforce Targeted Hardening Controls

This section details specific security controls applicable for Salesforce instances. These controls are designed to protect against broad access, data exfiltration, and unauthorized access to sensitive data within Salesforce.

Network and Login Controls

Restrict logins to only originate from trusted network locations.

See Salesforce guidance on network access and profile-based IP restrictions.

Restrict Login by IP Address

This control prevents credential misuse from unauthorized networks, effectively blocking access even if an attacker has stolen valid user credentials.

- Define login IP ranges at the profile level to only permit access from corporate and trusted network addresses.

- In Session Settings, enable “Enforce login IP ranges on every request” to ensure the check is not bypassed by an existing session.

See Salesforce guidance on setting trusted IP ranges.

Application and API Access Governance

Govern Connected App and API Access

Threat actors often bypass interactive login controls by leveraging generic API clients and stolen OAuth tokens. This policy flips the model from "allow by default" to "deny by default," to ensure that only vetted applications can connect.

- **Enable a "Deny by Default" API policy:** Navigate to API Access Control and enable “For admin-approved users, limit API access to only allowed connected apps.” This blocks all unapproved clients.
- **Maintain a minimal application allowlist:** Explicitly approve only essential Connected Apps. Regularly review this allowlist to remove unused or unapproved applications.
- **Enforce strict OAuth policies per app:** For each approved app, configure granular security policies, including restricting access to trusted IP ranges, enforcing MFA, and setting appropriate session and refresh token timeouts.
- **Revoke sessions when removing apps:** When revoking an app's access, ensure all active OAuth tokens and sessions associated with it are also revoked to prevent lingering access.
- **Organizational process and policy:** Create policies governing application integrations with third parties. Perform Third-Party Risk Management reviews of all integrations with business-critical applications (e.g., Salesforce, Google Workspace, Workday).

See Salesforce guidance on managing API access.

User Privilege and Access Management

Implement the Principle of Least Privilege

Users should only be granted the absolute minimum permissions required to perform their job functions.

- **Use a "Minimum Access" profile as a baseline:** Configure a base profile with minimal permissions and assign it to all new users by default. Limit the assignment of "View All" and "Modify All" permissions**.**
- **Grant privileges via Permission Sets:** Grant all additional access through well-defined Permission Sets based on job roles, rather than creating numerous custom profiles.
- **Disable API access for non-essential users:** The "API Enabled" permission is required for tools like Data Loader. Remove this permission from all user profiles and grant it only via a controlled Permission Set to a small number of justified users.
- **Hide the 'Setup' menu from non-admin users:** For all non-administrator profiles, remove access to the administrative "Setup" menu to prevent unauthorized configuration changes.
- **Enforce high-assurance sessions for sensitive actions:** Configure session settings to require a high-assurance session for sensitive operations such as exporting reports.

See Salesforce guidance on modifying session security settings.

See Salesforce guidance on requiring high-assurance session security.

See Salesforce guidance on "View All" and "Modify All" permissions.

Granular Data Access Policies

Enforce "Private" Organization-Wide Sharing Defaults (OWD)

- Set the internal and external **Organization-Wide Defaults (OWD)** to **"Private"** for all sensitive objects.
- Use strategic Sharing Rules or other sharing mechanisms to grant wider data access, rather than relying on broad access via the Role Hierarchy.

Leverage Restriction Rules for Row-Level Security

Restriction Rules act as a filter that is applied on top of all other sharing settings, allowing for fine-grained control over which records a user can see.

See Salesforce guidance on restriction rules.

Revoke Salesforce Support Login Access

Ensure that any users with access to sensitive data or with privileged access to the underlying Salesforce instance are setting strict timeouts on any Salesforce support access grants.

Revoke any standing requests and only re-enable with strict time limits for specific use cases. Be wary of enabling these grants from administrative accounts.

See Salesforce guidance on granting Salesforce Support login access.

Mandiant recommends running the Salesforce Security Health Check tool to identify and address misconfigurations. For additional hardening recommendations, reference the Salesforce Security Guide.

3. Logging and Detections

Salesforce Targeted Logging and Detections Controls

This section outlines key logging and detection strategies for Salesforce instances. These controls are essential for identifying and responding to advanced threats within the SaaS environment.

SaaS Applications Logging

To gain visibility into the tactics, techniques, and procedures (TTPs) used by threat actors against SaaS Applications, Mandiant recommends enabling critical log types in the organization's Salesforce environment and ingesting the logs into their Security Information and Event Management (SIEM).

What You Need in Place Before Logging

Before you turn on collection or write detections, make sure your organization is actually entitled to the logs you are planning to use - and that the right features are enabled.

1. **Entitlement check (must-have)**

    1. Most security logs/features are gated behind Event Monitoring via Salesforce Shield or the Event Monitoring Add-On. This applies to Real-Time Event Monitoring (RTEM) streaming and viewing.

2. **Pick your data model per use case**

    1. RTEM - Streams (near real-time alerting): Available in Enterprise/Unlimited/Developer subscriptions; streaming events retained ~3 days.

    2. RTEM - Storage: Many are Big Objects (native storage); some are standard objects (e.g. Threat Detection stores)

    3. Event Log Files (ELF) - CSV model (batch exports): Available in Enterprise/Performance/Unlimited editions.

    4. Event Log Objects (ELO) - SOQL model (queryable history): Shield/add-on required.

3. **Turn on what you need (and scope access)**

    1. Use Event Manager to enable/disable streaming and storing per event; viewing RTEM events.

    2. Grant access via profiles/permissions sets for RTEM and Threat Detection UI.

4. **Threat Detection & ETS**

    1. Threat Detection events are viewed in UI with Shield/add-on; stored in corresponding EventStore objects.

    2. Enhanced Transaction Security (ETS) is included with RTEM for block/MFA/notify actions on real-time events.

Recommended Log Sources to Monitor

- **Login History (LoginHistory)**: Tracks all login attempts, including username, time, IP address, status (successful/failed), and client type. This allows you to identify unusual login times, unknown locations, or repeated failures, which could indicate credential stuffing or account compromise.

- **Login Events (LoginEventStream)**: LoginEvent tracks the login activity of users who log in to Salesforce.

- **Setup Audit Trail (SetupAuditTrail)**: Records administrative and configuration changes within your Salesforce environment. This helps track changes made to permissions, security settings, and other critical configurations, facilitating auditing and compliance efforts.

- **API Calls (ApiEventStream)**: Monitors API usage and potential misuse by tracking calls made by users or connected apps.

- **Report Exports (ReportEventStream)**: Provides insights into report downloads, helping to detect potential data exfiltration attempts.

- **List View Events (ListViewEventStream)**: Tracks user interaction with list views, including access and manipulation of data within those views.
- **Bulk API Events (BulkApiResultEvent)**: Track when a user downloads the results of a Bulk API request.
- **Permission Changes (PermissionSetEvent)**: Tracks changes to permission sets and permission set groups. This event initiates when a permission is added to, or removed from a permission set.
- **API Anomaly (ApiAnomalyEvent)**: Track anomalies in how users make API calls.
- **Unique Query Event Type**: Unique Query events capture specific search queries (SOQL), filter IDs, and report IDs that are processed, along with the underlying database queries (SQL).
- **External Identity Provider Event Logs**: Track information from login attempts using SSO. **(Please follow the guidance provided by your Identity Provider for monitoring and collecting IdP event logs.)**

These log sources will provide organizations with the logging capabilities to properly collect and monitor the common TTPs used by threat actors. The key log sources to monitor and observable Salesforce activities for each TTP are as follows:

TTP	Observable Salesforce Activities	Log Sources
Vishing	• Suspicious login attempts (rapid failures). • Logins from unusual IPs/ASNs (e.g., Mullvad/Tor). • OAuth (“Remote Access 2.0”) from unrecognized clients.	• Login History • LoginEventStream/LoginEvent • Setup Audit Trail
Malicious Connected App Authorization (e.g., Data Loader, custom scripts)	• New Connected App creation/modification (broad scopes: api, refresh_token, offline_access). • Policy relaxations (Permitted Users, IP restrictions). • Granting of API Enabled / “Manage Connected Apps” via perms.	• Setup Audit Trail • PermissionSetEvent • LoginEventStream/LoginEvent (OAuth)
Data Exfiltration (via API, Data Loader, reports)	• High-rate Query/QueryMore/QueryAll bursts. • Large RowsProcessed/RecordCount in reports & list views (chunked). • Bulk job result downloads.	• ApiEventStream/ApiEvent • ReportEventStream/ReportEvent • ListViewEventStream/ListViewEvent • BulkApiResultEvent • FileEvent/FileEventStore • ApiAnomalyEvent/ReportAnomalyEvent

	- File/attachment downloads at scale - Permissions elevated (e.g., View/Modify All Data, API Enabled).	- Unique Query Event Type
Lateral Movement/Persistence (within Salesforce or to other cloud platforms)	- New user/service accounts. - LoginAs activity. - Logins from VPN/Tor after SF OAuth. - Pivots to Okta/M365, then Graph data pulls.	- Setup Audit Trail - PermissionSetEvent - LoginAsEventStream

SaaS Applications Detections

While native SIEM threat detections provide some protection, they often lack the centralized visibility needed to connect disparate events across a complex environment. By developing custom targeted detection rules, organizations can proactively detect malicious activities.

Data Exfiltration & Cross-SaaS Lateral Movement (Post-Authorization)

MITRE Mapping: TA0010 - Exfiltration & TA0008 - Lateral Movement

Scenario & Objectives

After an user authorizes a (malicious or spoofed) Connected App, UNC6040 typically:

1. Performs data exfiltration quickly (REST pagination bursts, Bulk API downloads, lards/sensitive report exports).
2. Pivots to Okta/Microsoft 365 from the same risky egress IP to expand access and steal more data.

The objective here is to detect Salesforce OAuth → Exfil within ≤10 minutes, and Salesforce OAuth → Okta/M365 login within ≤60 minutes (same risky IP), plus single-signal, low-noise exfil patterns.

Baseline & Allowlist

Re-use the lists you already maintain for the vishing phase and add two regex helpers for content focus.

- STRING
    - ALLOWLIST_CONNECTED_APP_NAMES
    - KNOWN_INTEGRATION_USERS (user ids/emails that legitimately use OAuth)

  - VPN_TOR_ASNS (ASNs as strings)
- CIDR
  - ENTERPRISE_EGRESS_CIDRS (your corporate/VPN public egress)
- REGEX
  - SENSITIVE_REPORT_REGEX

```
(?i)\b(all|export|dump)\b.*\b(contact|lead|account|customer|pii|email|phone|ssn)\b
```

-
  - M365_SENSITIVE_GRAPH_REGEX

```
(?i)^https?://graph\.microsoft\.com/(beta|v1\.0)/(users|me)/messages
(?i)^https?://graph\.microsoft\.com/(beta|v1\.0)/drives/.*/items/.*/content
(?i)^https?://graph\.microsoft\.com/(beta|v1\.0)/reports/
(?i)^https?://graph\.microsoft\.com/(beta|v1\.0)/users(\?|$)
```

High Fidelity Detection Catalog (Pseudo-Code)

Salesforce OAuth → Data Exfil in ≤10 Minutes (Multi-Event)

Suspicious OAuth followed within 10m by Bulk result download, REST pagination burst, or sensitive/large report export by the same user.

Why high-fidelity: Matches UNC6040's "approve → drain" pattern; tight window + volume thresholds.

Key signals:

- OAuth success (unknown app OR allowlisted+risky egress), bind on user.
- Then any of:
  - BulkApiResultEvent with big RowsProcessed/RecordCount
  - ApiEventStream many query/queryMore calls
  - ReportEventStream large/sensitive report export
- Lists/knobs: ENTERPRISE_EGRESS_CIDRS, VPN_TOR_ASNS, SENSITIVE_REPORT_REGEX.

```
$oauth.metadata.product_name = "SALESFORCE"
$oauth.metadata.log_type = "SALESFORCE"
$oauth.extracted.fields["LoginType"] = "Remote Access 2.0"
```

```
($oauth.extracted.fields["Status"] = "Success" or
$oauth.security_result.action_details = "Success")
( not ($app in %ALLOWLIST_CONNECTED_APP_NAMES)
or ( ($app in %ALLOWLIST_CONNECTED_APP_NAMES)
and ( not ($ip in cidr %ENTERPRISE_EGRESS_CIDRS)
or strings.concat(ip_to_asn($ip), "") in %VPN_TOR_ASNS ) ) )
$uid = coalesce($oauth.principal.user.userid, $oauth.extracted.fields["UserId"])

$bulk.metadata.product_name = "SALESFORCE"
$bulk.metadata.log_type = "SALESFORCE"
$bulk.metadata.product_event_type = "BulkApiResultEvent"
$uid = coalesce($bulk.principal.user.userid, $bulk.extracted.fields["UserId"])

match:
$uid over 10m
```

Or

```
$oauth.metadata.product_name = "SALESFORCE"
$oauth.metadata.log_type = "SALESFORCE"
$oauth.extracted.fields["LoginType"] = "Remote Access 2.0"
($oauth.extracted.fields["Status"] = "Success" or
$oauth.security_result.action_details = "Success")
( not ($app in %ALLOWLIST_CONNECTED_APP_NAMES)
or ( ($app in %ALLOWLIST_CONNECTED_APP_NAMES)
and ( not ($ip in cidr %ENTERPRISE_EGRESS_CIDRS)
or strings.concat(ip_to_asn($ip), "") in %VPN_TOR_ASNS ) ) )
$uid = coalesce($oauth.principal.user.userid, $oauth.extracted.fields["UserId"])

$api.metadata.product_name = "SALESFORCE"
$api.metadata.log_type = "SALESFORCE"
$api.metadata.product_event_type = "ApiEventStream"
$uid = coalesce($api.principal.user.userid, $api.extracted.fields["UserId"])

match:
$uid over 10m
```

Or

```
$oauth.metadata.product_name = "SALESFORCE"
$oauth.metadata.log_type = "SALESFORCE"
$oauth.extracted.fields["LoginType"] = "Remote Access 2.0"
($oauth.extracted.fields["Status"] = "Success" or
$oauth.security_result.action_details = "Success")
```

```
( not ($app in %ALLOWLIST_CONNECTED_APP_NAMES)
or ( ($app in %ALLOWLIST_CONNECTED_APP_NAMES)
and ( not ($ip in cidr %ENTERPRISE_EGRESS_CIDRS)
or strings.concat(ip_to_asn($ip), "") in %VPN_TOR_ASNS ) ) )
$uid = coalesce($oauth.principal.user.userid, $oauth.extracted.fields["UserId"])

$report.metadata.product_name = "SALESFORCE"
$report.metadata.log_type = "SALESFORCE"
$report.metadata.product_event_type = "ReportEventStream"
strings.to_lower(coalesce($report.extracted.fields["ReportName"], "")) in regex
SENSITIVE_REPORT_REGEX
$uid = coalesce($report.principal.user.userid,
$report.extracted.fields["UserId"])

match:
$uid over 10m
```

Note: Single event rule can also be used instead of multi-event rules in this case where only the Product Event Types like ApiEventStream, BulkApiResultEvent, ReportEventStream can be used as a single event rule to be monitored. But, care has to be taken if a single event rule is established as these can be very noisy, and thus the reference lists should be actively monitored.

Bulk API Large Result Download (Non-Integration User)

Bulk API/Bulk v2 result download above threshold by a human user.

Why high-fidelity: Clear exfil artifact.

Key signals: BulkApiResultEvent, user not in KNOWN_INTEGRATION_USERS.

Lists/knobs: KNOWN_INTEGRATION_USERS, size threshold.

```
$e.metadata.product_name = "SALESFORCE"
$e.metadata.log_type = "SALESFORCE"
$e.metadata.product_event_type = "BulkApiResultEvent"
not (coalesce($e.principal.user.userid, $e.extracted.fields["UserId"]) in
%KNOWN_INTEGRATION_USERS)
```

REST Query Pagination Burst (query/queryMore)

High-rate query*/queryMore calls over a short window.

Why high-fidelity: Mimics scripted drains; steady human usage won’t hit burst thresholds.

Key signals: ApiEventStream, Operation in query, queryMore, query_all, queryall, count ≥ threshold in 10m, user not in KNOWN_INTEGRATION_USERS.

Lists/knobs: burst threshold, KNOWN_INTEGRATION_USERS.

```
$api.metadata.product_name = "SALESFORCE"
$api.metadata.log_type = "SALESFORCE"
$api.metadata.product_event_type = "ApiEventStream"
not (coalesce($api.principal.user.userid, $api.extracted.fields["UserId"]) in
%KNOWN_INTEGRATION_USERS)
strings.to_lower(coalesce($api.extracted.fields["Operation"], "")) in regex `(?
i)^(query|querymore|query_all|queryall)
```

%hostname% %pathname%

Cybercrime Observations from the Frontlines: UNC6040 Proactive Hardening Recommendations

%byline% ⋮
⋮ %published-time%

```
$uid = coalesce($api.principal.user.userid, $api.extracted.fields["UserId"])
```

Sensitive Report Export by Non-Integration User

Exports of large or sensitive-named reports by a human.

Why high-fidelity: Report extracts are a common, noisy-to-attackers but high-signal vector.

Key signals: ReportEventStream, high RowsProcessed or ReportName matches SENSITIVE_REPORT_REGEX, user not in KNOWN_INTEGRATION_USERS.

Lists/knobs: SENSITIVE_REPORT_REGEX, KNOWN_INTEGRATION_USERS.

```
$e.metadata.product_name = "SALESFORCE"
$e.metadata.log_type = "SALESFORCE"
$e.metadata.product_event_type = "ReportEventStream"
not (coalesce($e.principal.user.userid, $e.extracted.fields["UserId"]) in
%KNOWN_INTEGRATION_USERS)
strings.to_lower(coalesce($e.extracted.fields["ReportName"], "")) in regex
%SENSITIVE_REPORT_REGEX
```

Salesforce OAuth → Okta/M365 Login From Same Risky IP in ≤60 Minutes (Multi-Event)

Suspicious Salesforce OAuth followed within 60m by Okta or Entra ID login from the same public IP, where the IP is off-corp or VPN/Tor ASN.

Why high-fidelity: Ties the attacker’s egress IP across SaaS within a tight window.

Key signals:

- Salesforce OAuth posture (unknown app OR allowlisted+risky egress)
- OKTA* or OFFICE_365 USER_LOGIN from the same IP

Lists/knobs: ENTERPRISE_EGRESS_CIDRS, VPN_TOR_ASNS. (Optional sibling rule binding by user email if identities are normalized.)

```
$oauth.metadata.product_name = "SALESFORCE"
$oauth.metadata.log_type = "SALESFORCE"
$oauth.extracted.fields["LoginType"] = "Remote Access 2.0"
($oauth.extracted.fields["Status"] = "Success" or
$oauth.security_result.action_details = "Success")
( not ($app in %ALLOWLIST_CONNECTED_APP_NAMES)
or ( ($app in %ALLOWLIST_CONNECTED_APP_NAMES)
and ( not ($ip in cidr %ENTERPRISE_EGRESS_CIDRS)
or strings.concat(ip_to_asn($ip), "") in %VPN_TOR_ASNS )
```

```
$ip = coalesce($oauth.principal.asset.ip, $oauth.principal.ip)

$okta.metadata.log_type in "OKTA"
$okta.metadata.event_type = "USER_LOGIN"
$ip = coalesce($okta.principal.asset.ip, $okta.principal.ip) = $ip

$o365.metadata.log_type = "OFFICE_365"
$o365.metadata.event_type = "USER_LOGIN"
$ip = coalesce($o365.principal.asset.ip, $o365.principal.ip)

match:
$ip over 10m
```

M365 Graph Data-Pull After Risky Login

Entra ID login from risky egress followed by Microsoft Graph endpoints that pull mail/files/reports.

Why high-fidelity: Captures post-login data access typical in account takeovers.

Key signals: OFFICE_365 USER_LOGIN with off-corp IP or VPN/Tor ASN, then HTTP to URLs matching M365_SENSITIVE_GRAPH_REGEX by the same account within hours.

Lists/knobs: ENTERPRISE_EGRESS_CIDRS, VPN_TOR_ASNS, M365_SENSITIVE_GRAPH_REGEX.

```
$login.metadata.log_type = "OFFICE_365"
$login.metadata.event_type = "USER_LOGIN"
$ip  = coalesce($login.principal.asset.ip, $login.principal.ip)
( not ($ip in cidr %ENTERPRISE_EGRESS_CIDRS)
 or strings.concat(ip_to_asn($ip), "") in %VPN_TOR_ASNS )
$acct = coalesce($login.principal.user.userid,
$login.principal.user.email_addresses)

$http.metadata.product_name in ("Entra ID","Microsoft")
($http.metadata.event_type = "NETWORK_HTTP" or $http.target.url != "")
$acct = coalesce($http.principal.user.userid,
$http.principal.user.email_addresses)
strings.to_lower(coalesce($http.target.url, "")) in regex
%M365_SENSITIVE_GRAPH_REGEX

match:
$acct over 30m
```

Tuning & Exceptions

- Identity joins - The lateral rule groups by IP for robustness. If you have strong identity normalization (Salesforce <-> Okta <-> M365), clone it and match on user email instead of IP.
- Change windows - Suppress time-bound rules during approved data migrations/Connected App onboarding (temporarily add vendor app to ALLOWLIST_CONNECTED_APP_NAMES)
- Integration accounts - Keep KNOWN_INTEGRATION_USERS current; most noise in exfil rules comes from scheduled ETL.
- Egress hygiene - Keep ENTERPRISE_EGRESS_CIDRS current; stale NAT/VPN ranges inflate VPN/Tor findings.
- Streaming vs stored - The aforementioned rules assume Real-Time Event Monitoring Stream objects (e.g., ApiEventStream, ReportEventStream, ListViewEventStream, BulkApiResultEvent). For historical hunts, query the stored equivalents (e.g., ApiEvent, ReportEvent, ListViewEvent) with the same logic.

IOC-Based Detections

Scenario & Objectives

A malicious threat actor has either successfully accessed or attempted to access an organization's network.

The objective is to detect the presence of known UNC6040 IOCs in the environment based on all of the available logs.

Reference Lists

Reference lists organizations should maintain:

- STRING
    - UNC6040_IOC_LIST (IP addresses from threat intel sources eg. VirusTotal)

List of indicators of compromise (IOCs).

High Fidelity Detection Catalog (Pseudo-Code)

UNC6040 IP_IoC Detected

A known IOC associated with UNC6040 was detected in the organization's environment either from a source or destination connection.

- High-fidelity when conditioned on source or destination IP address matches a known UNC6040 IOC.

```
($e.principal.ip in %unc6040_IoC_list) or ($e.target.ip in %unc6040_IoC_list)
```

Acknowledgements

We would like to thank Salesforce for their collaboration and assistance in building this guide.

Posted in

- Threat Intelligence