# Okta For Good and Bad

## Hybrid Attack Paths Crossing Okta Organizations


Michael Grafnetter
Principal Security Researcher

Lance Cain
Offensive Security Engineer - Consulting Services

# Agenda

01 Okta Security Primer

02 Okta + Devices

03 Okta + JAMF

04 Okta + Active Directory

05 Okta + GitHub

06 BloodHound OpenGraph Collectors

# Okta Security Primer

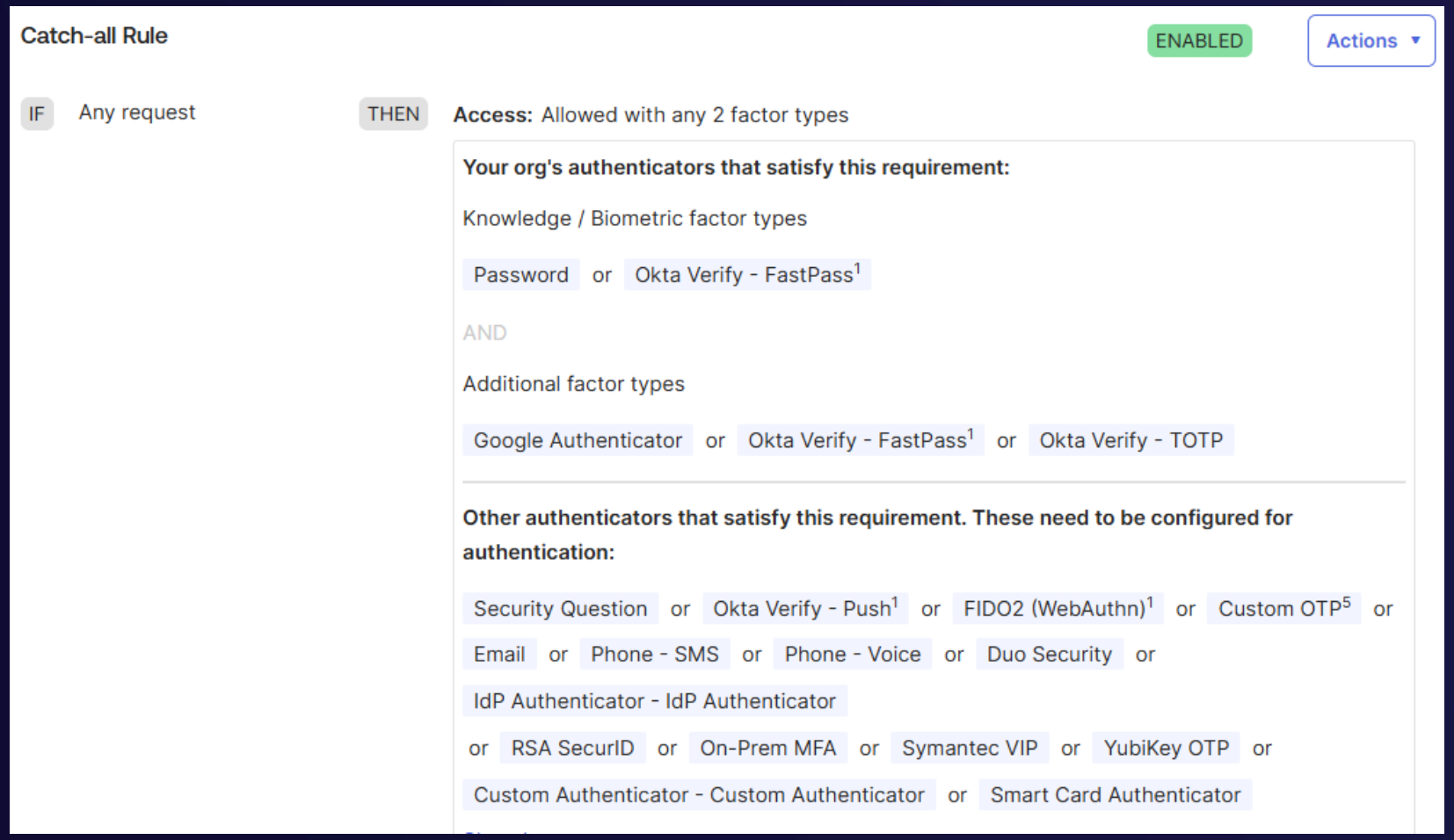
## Understanding Okta Attack Paths

# Why Okta?


ALL YOUR DEVICES
DESKTOPS · LAPTOPS
SMARTPHONES · TABLETS
ALL YOUR PEOPLE
EMPLOYEES · CUSTOMERS
PARTNERS · CONTRACTORS
Internal
External
okta
ON-PREM IDENTITY
IBM
Tivoli software
ORACLE
ca
RSA
Windows Server
Active Directory
LDAP
ALL YOUR APPLICATIONS
CLOUD
Google Apps
Office 365
salesforce
workday.
box
ON-PREM
SAP
ORACLE
Java
Microsoft
.NET
MOBILE
workday
salesforce
box

# Okta Is Secure by Default

## Everybody gets MFA!



# Okta Attack Paths Are Short

## Compared to Active Directory (AD) and Microsoft Entra ID

### No Group Nesting

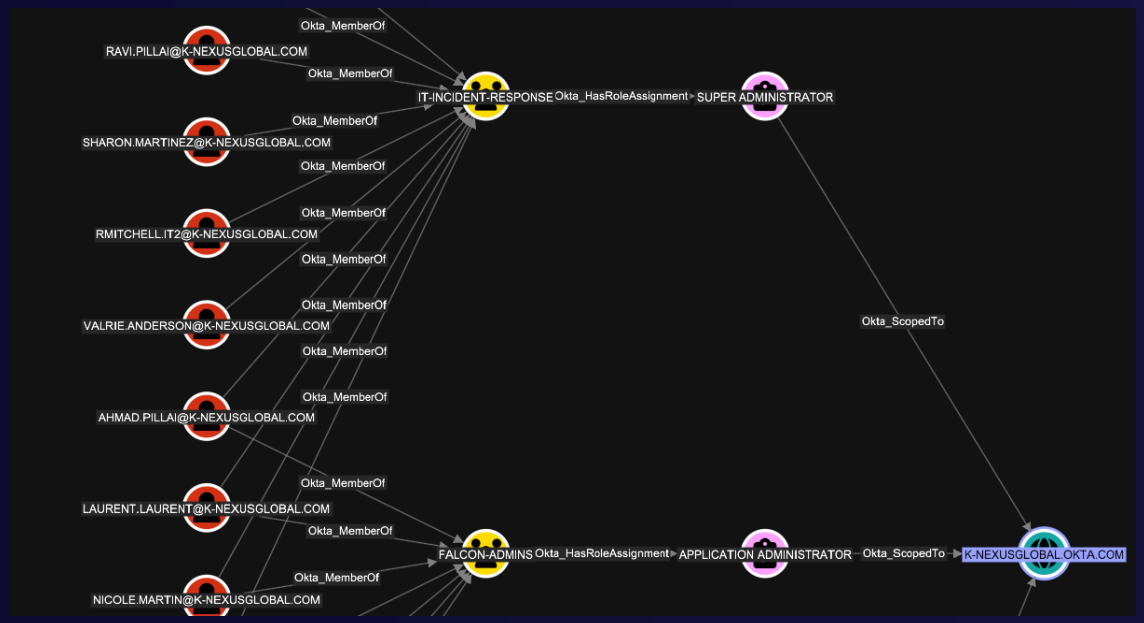
- Only users can be members of groups. Groups and applications cannot be added to groups
- Synchronized group membership gets correctly flattened

### Privilege Elevation Minimized

- Only Super Admins can modify principals with role assignment
- True for users, groups, and apps
- Valid for built-in and custom roles

### Lateral Movement Limited

- No built-in computer management
  - MDM
  - Group Policy
- No remote computer management
  - PowerShell Remoting
  - SSH
  - SMB
  - WMI
  - Windows RPC

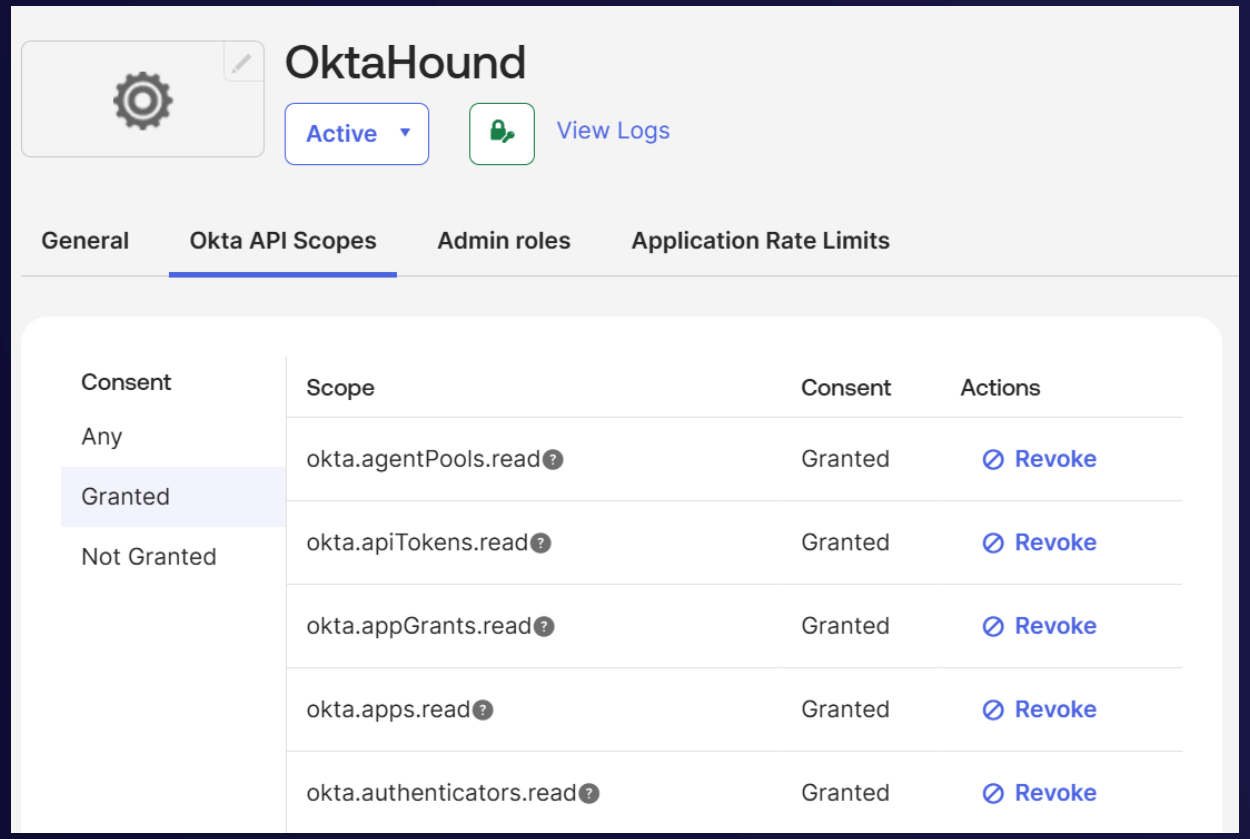


# Okta Security Is Boring

# Overprivileged Auditors

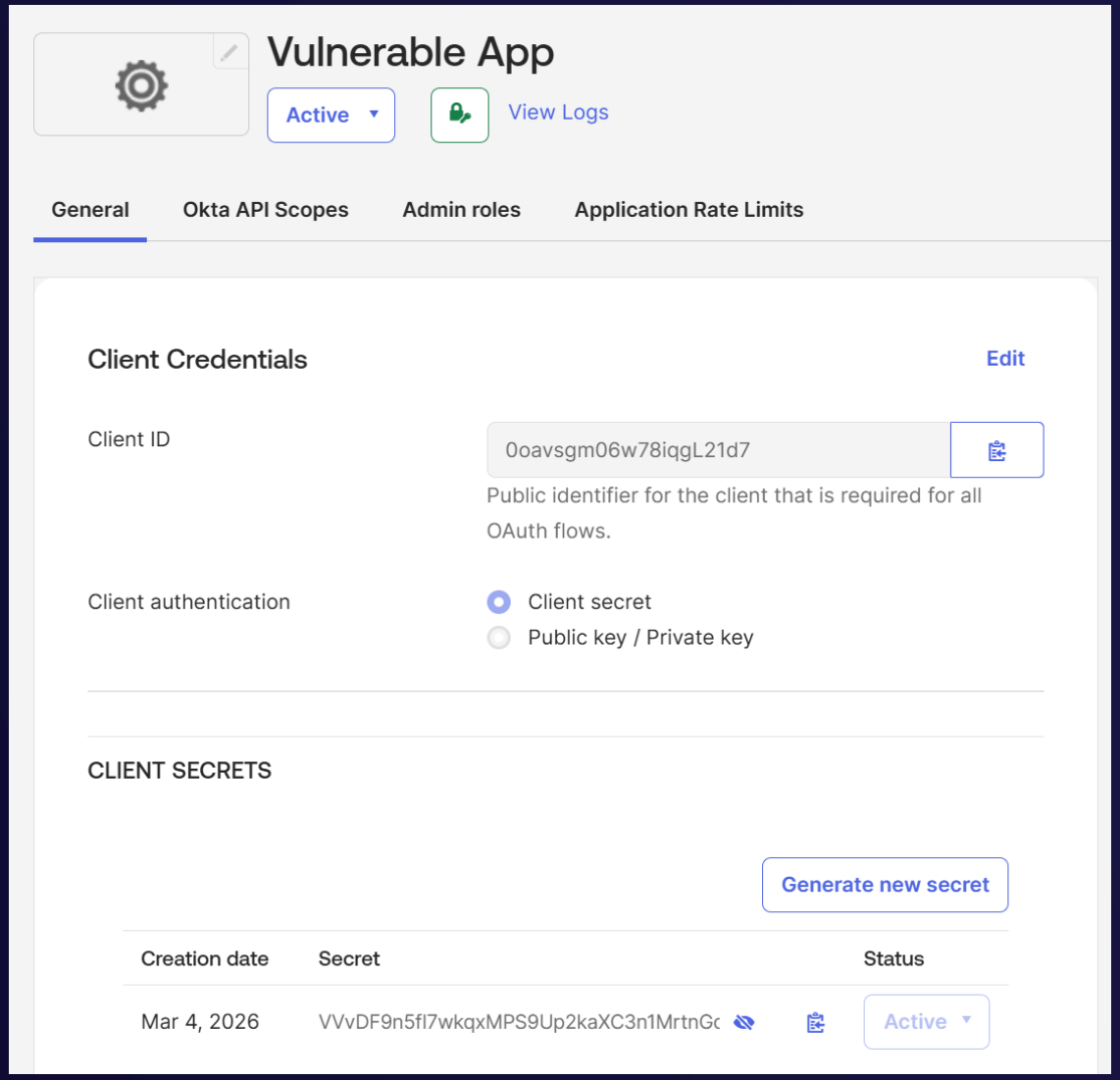
## Security Auditors Are Typically Super Admins

- Read-Only Administrators role cannot read role assignments
- Permission to read OAuth 2.0 scopes grants cannot be delegated



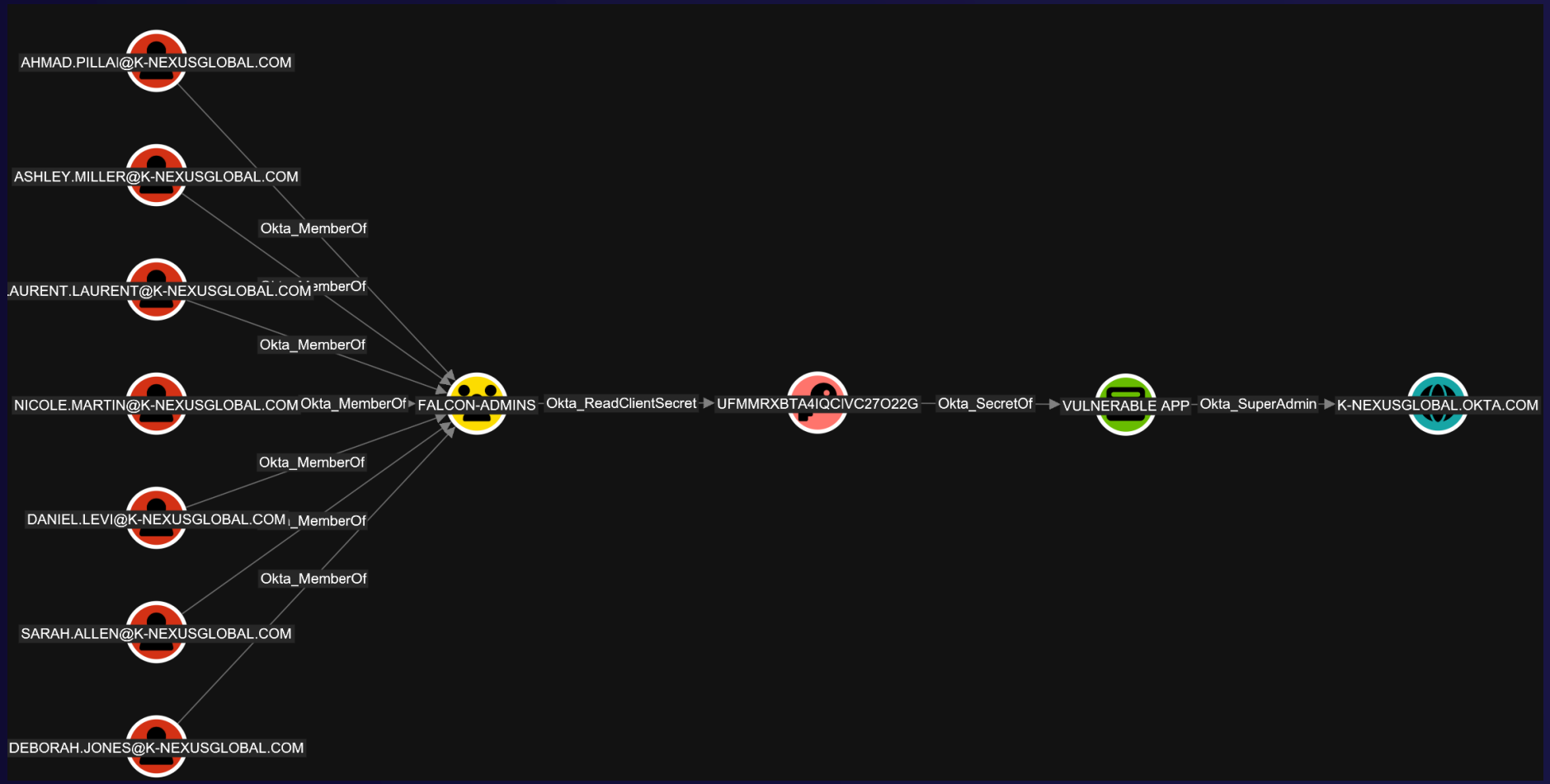
# Okta Privilege Elevation Paths

## Read-Only Administrators



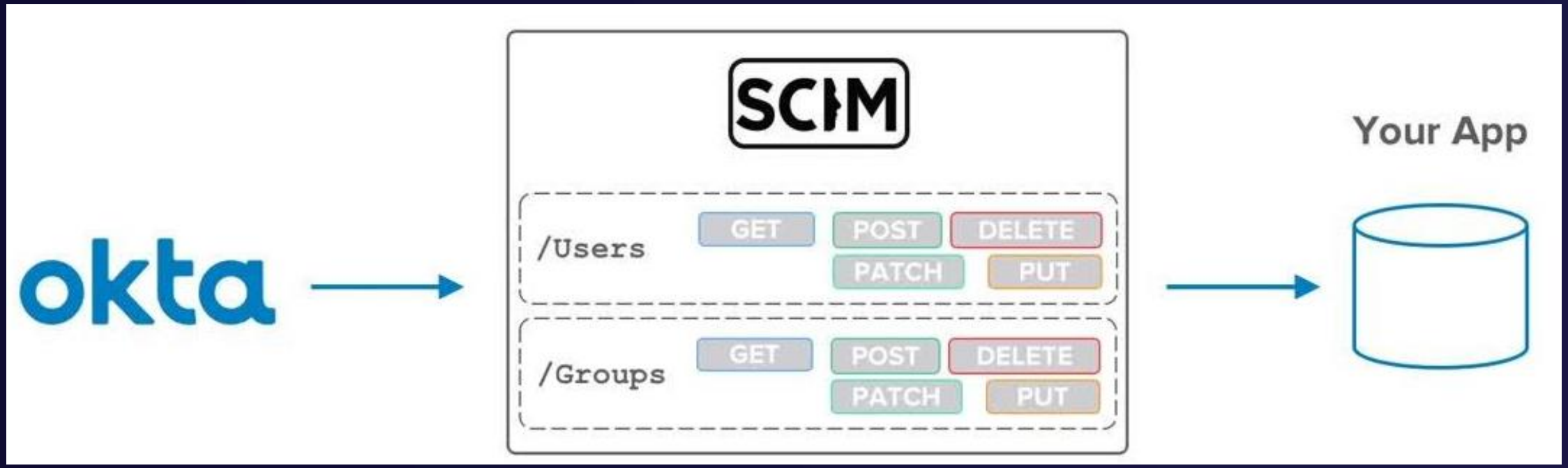
# Okta Privilege Elevation Paths

## Read-Only Administrators



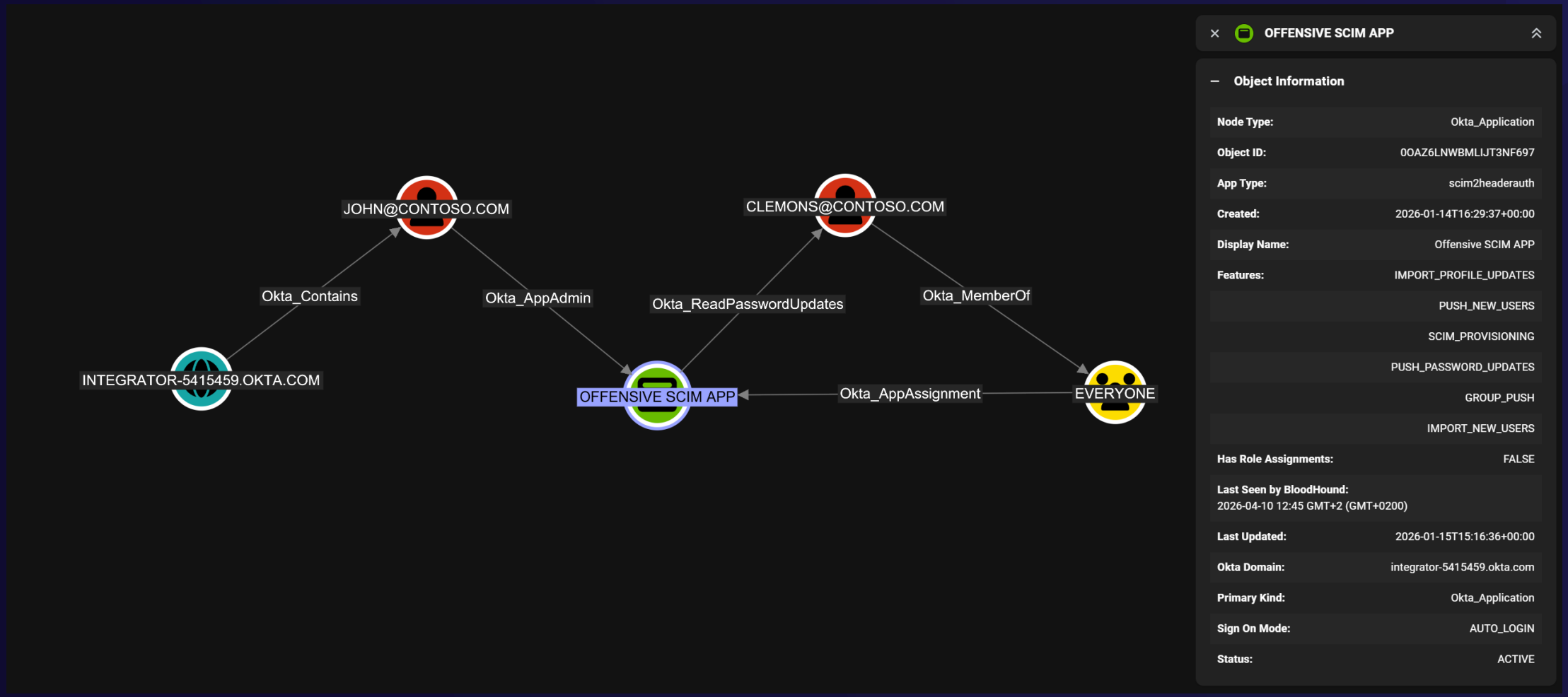
# Okta Privilege Elevation Paths

## SCIM: System for Cross-domain Identity Management



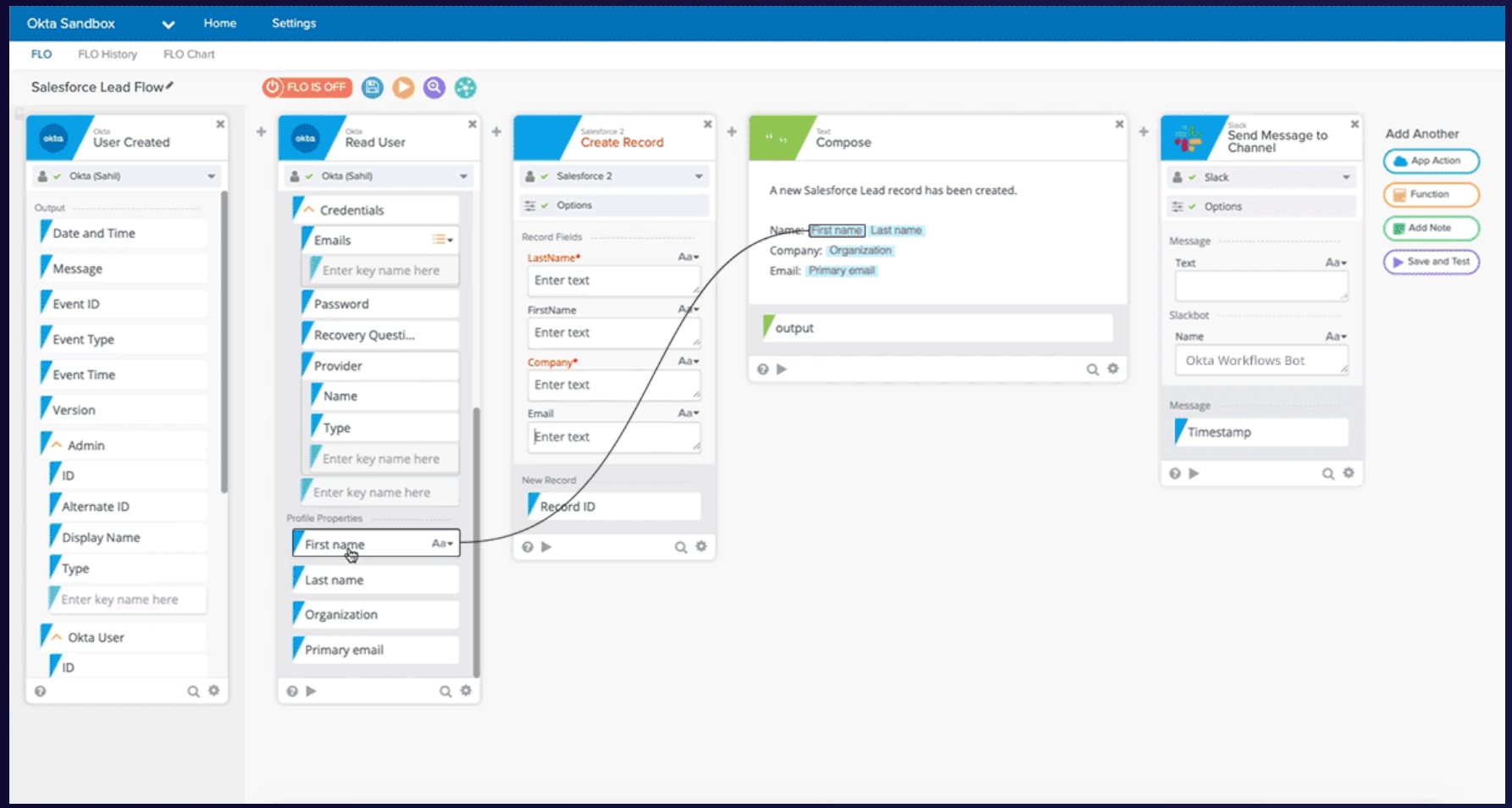
# Okta Privilege Elevation Paths

## SCIM: System for Cross-domain Identity Management



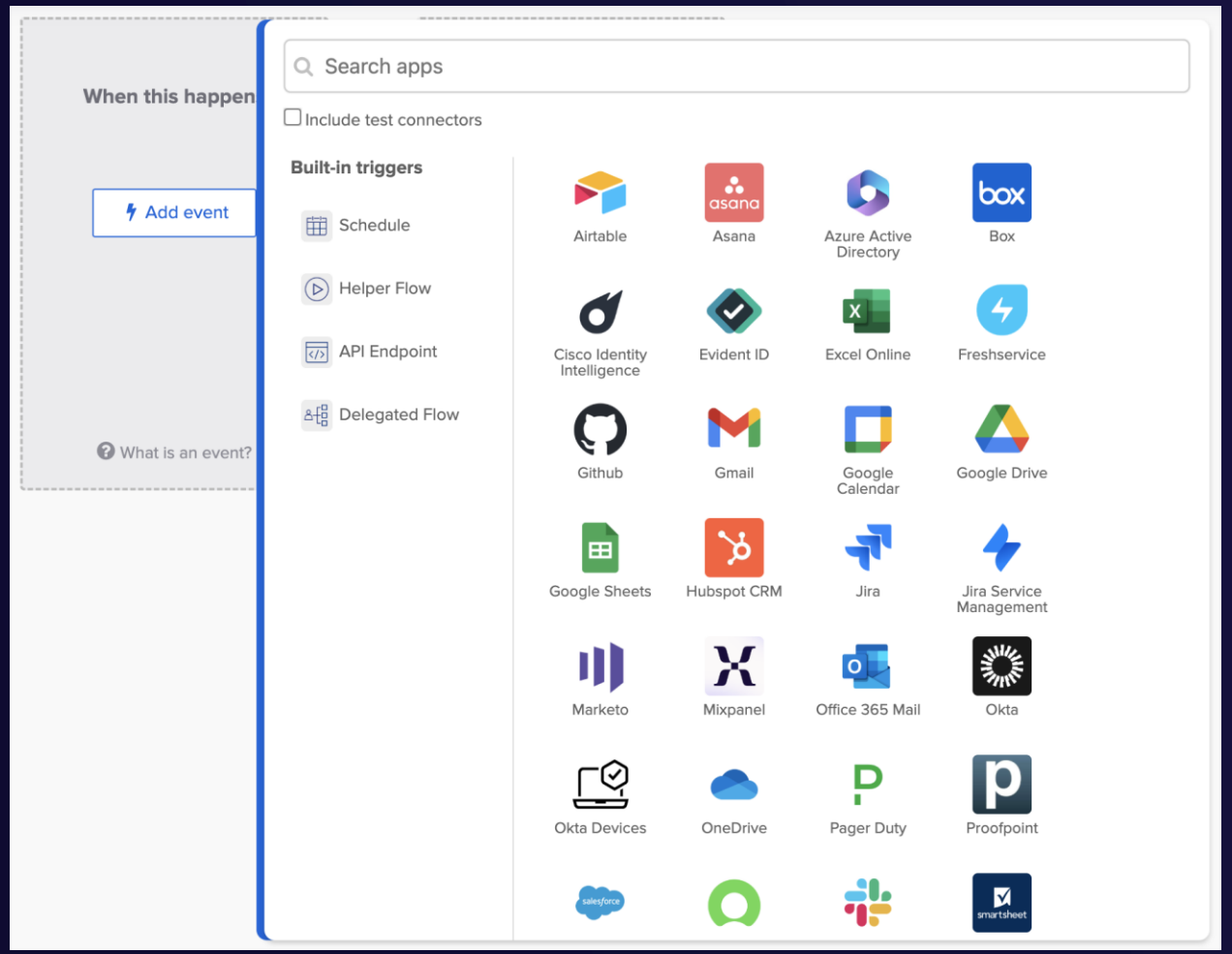
# Okta Privilege Elevation Paths

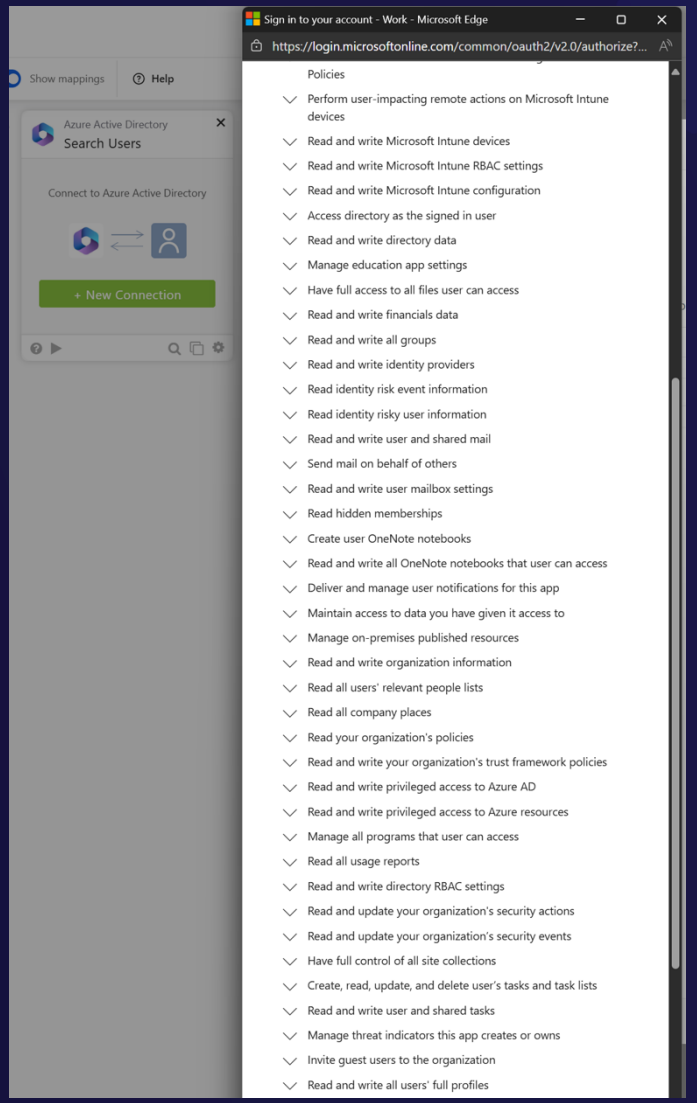
## Okta Workflows



# Okta Privilege Elevation Paths

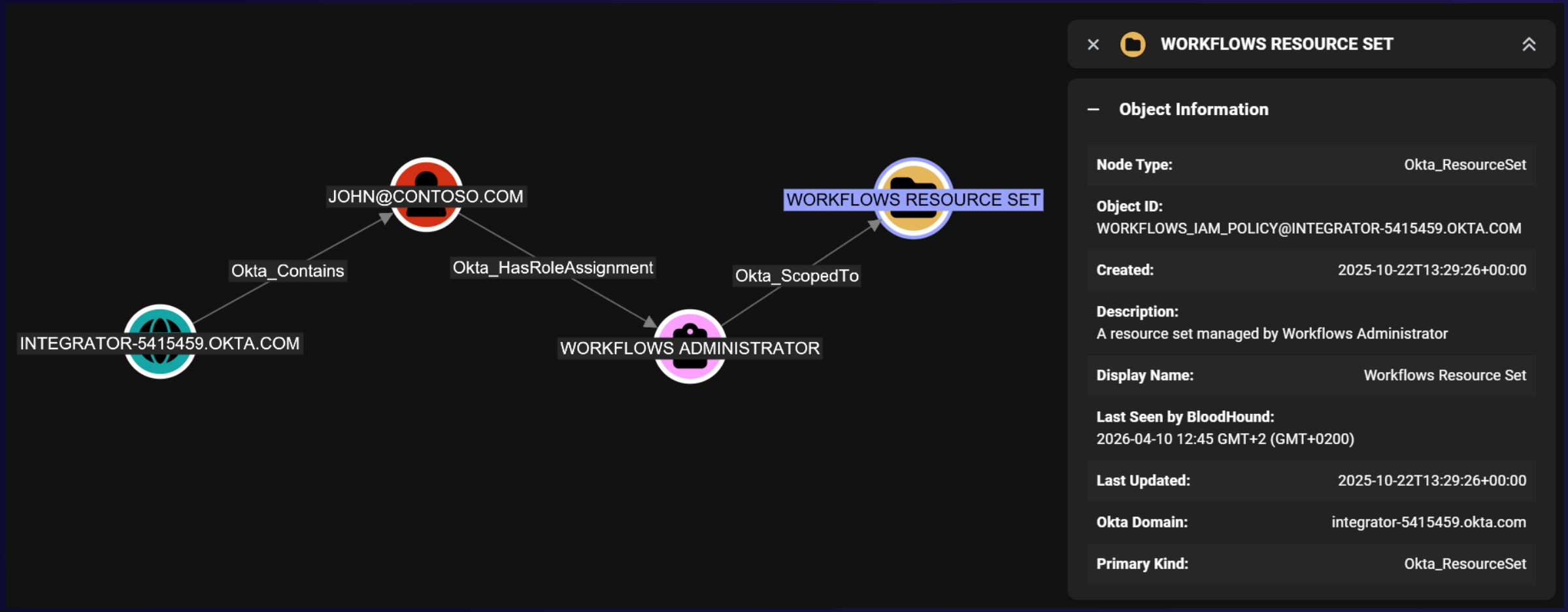
## Okta Workflows



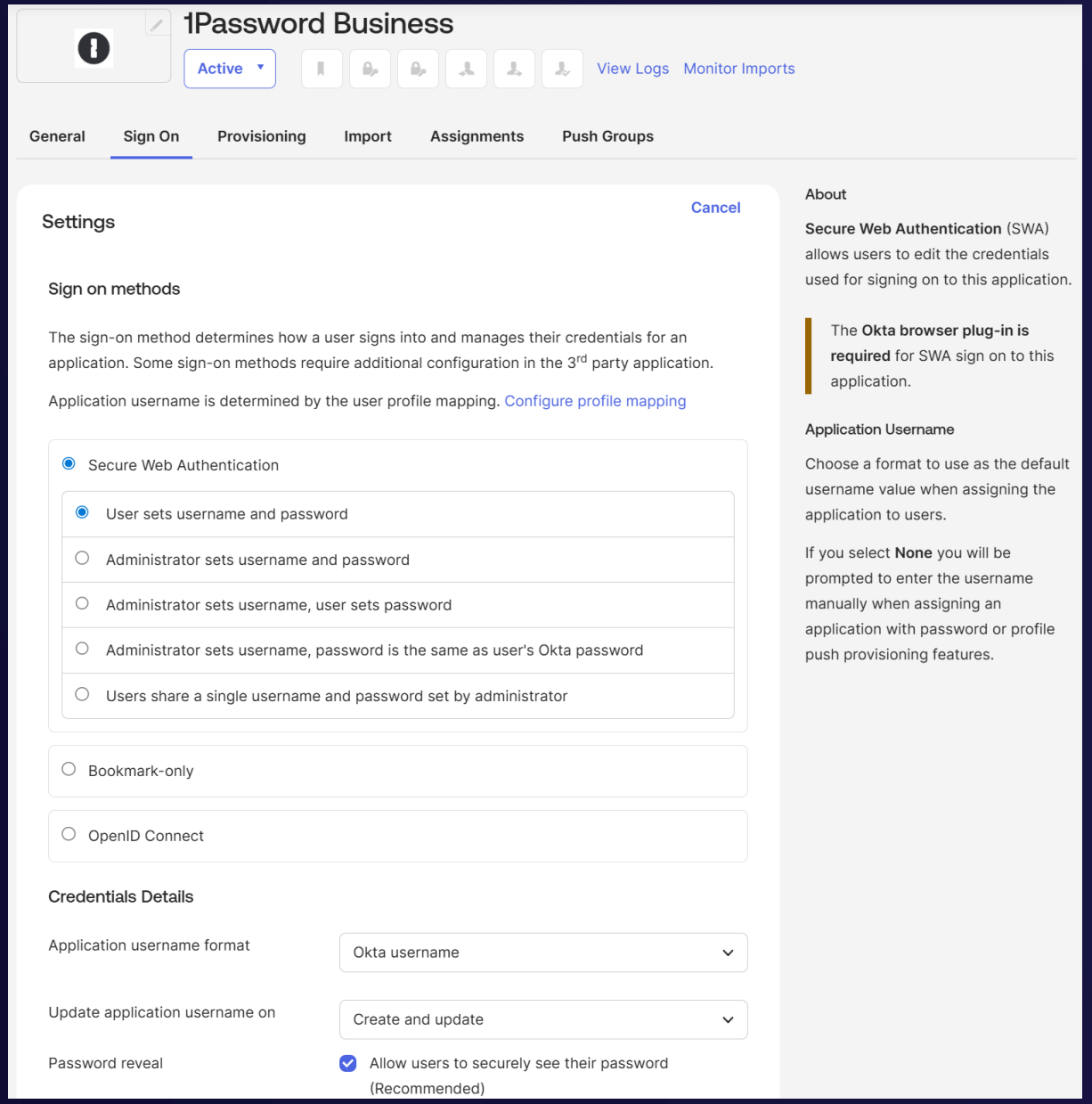



# Okta Privilege Elevation Paths

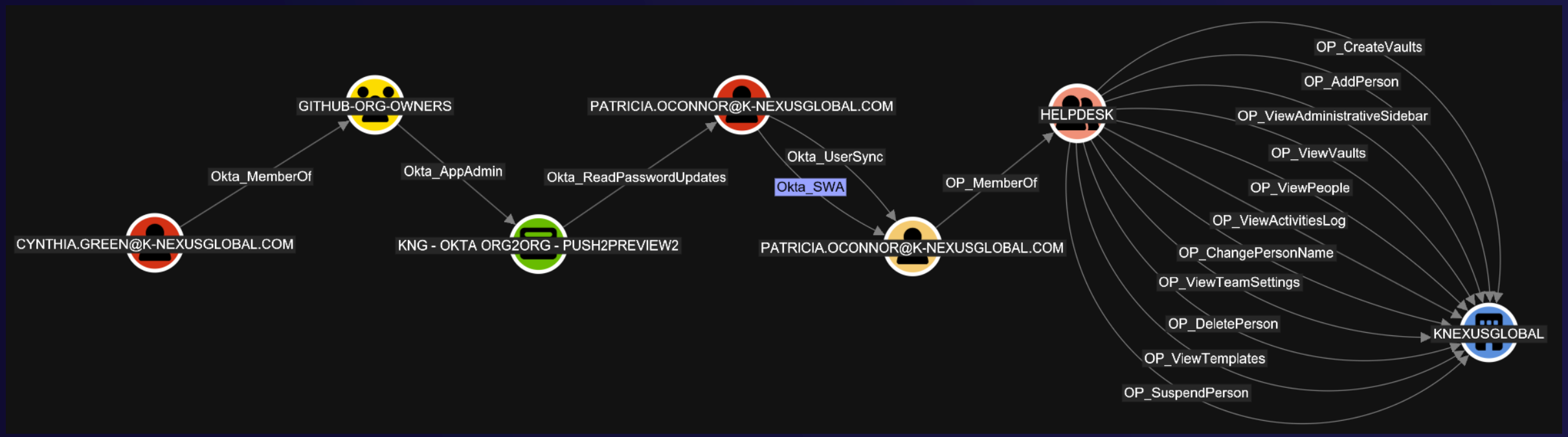
## Okta Workflows



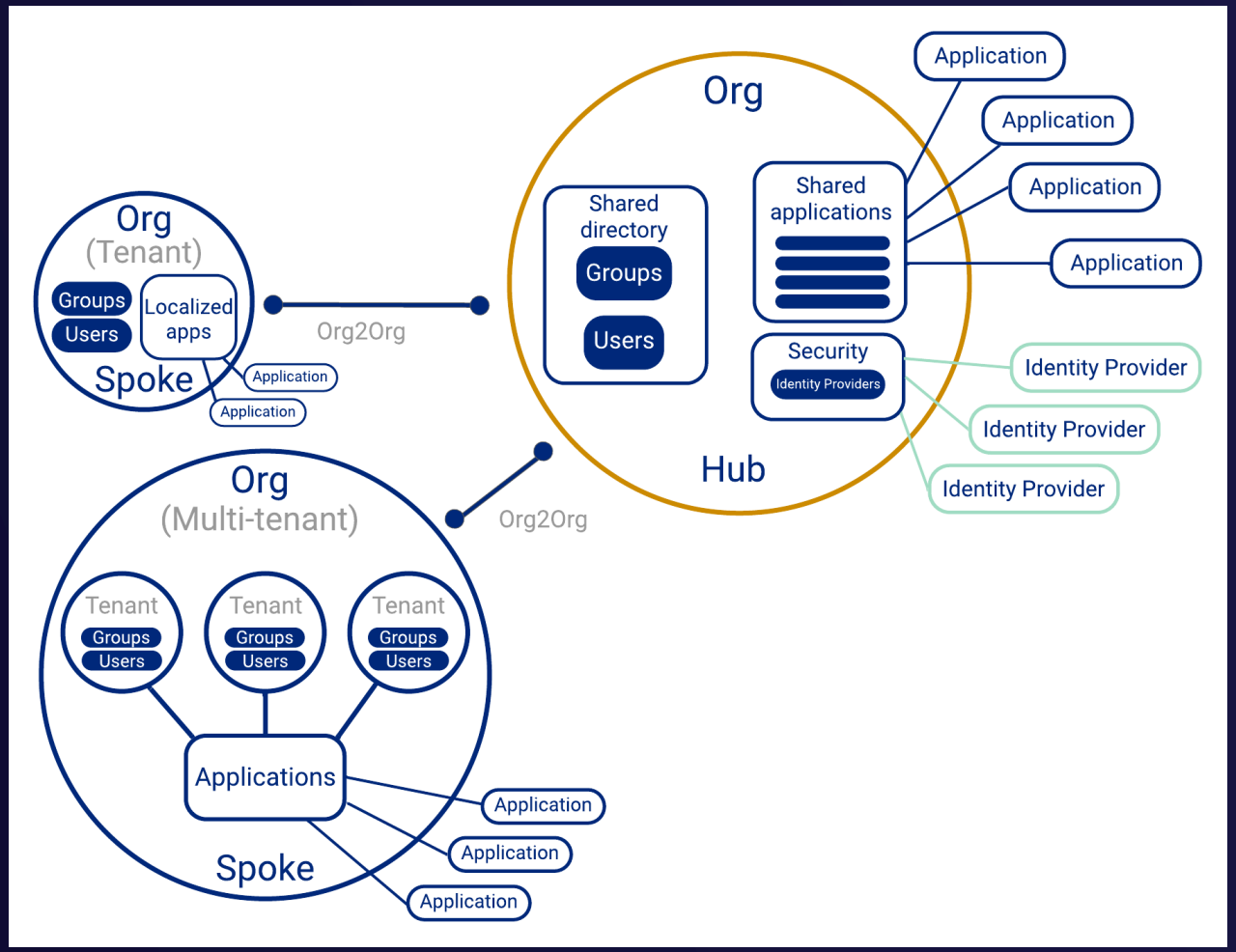
# Okta Secure Web Authentication



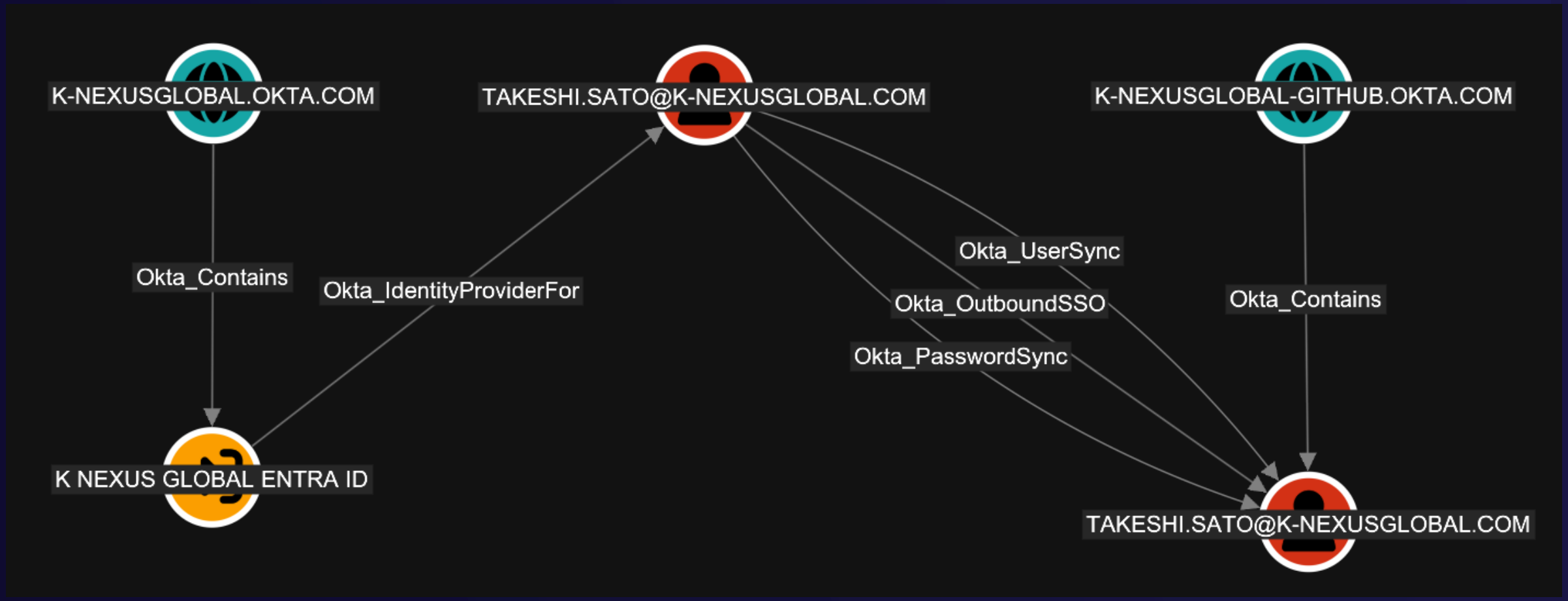
# Okta Secure Web Authentication



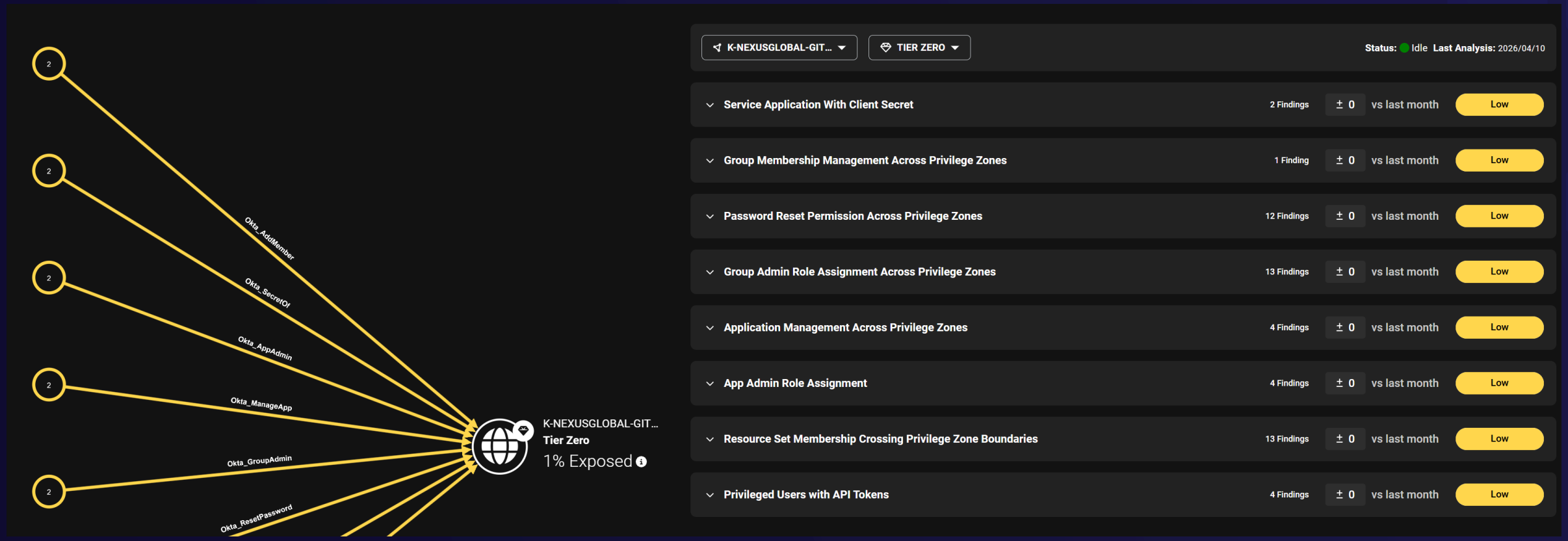
# Okta Org2Org



# Okta Org2Org



# BloodHound Enterprise Findings



# BloodHound Enterprise Findings

## Privileged Users with API Tokens

### Recommended Remediation

Prefer OAuth 2.0 access tokens over API tokens (SSWS tokens). Applications performing OAuth 2.0 authentication must be registered in Okta as service applications or API service integrations and assigned fine-grained permissions. Such applications then authenticate using public/private key pairs (preferred option) or client secrets (strong passwords).

In situations where the use of API tokens is justified, limit their validity to the minimum required duration and restrict the use of tokens to specific network zones.

Regularly audit API token usage to identify and revoke unnecessary tokens. Rotate tokens immediately if there is any suspicion of compromise.

Issue tokens only from dedicated low-privilege service identities rather than human administrator accounts. Alert on token usage from unusual network locations, devices, or access patterns.

### Principals

Accepted

| Severity | Exposure | Non-Tier Zero Source | Tier Zero Target | Impact |
|---|---|---|---|---|
| | 0 | OKTAHOUND | MGRAFN | 10.8K |
| | 0 | KNEXUS-ANSIBLE-API2 | JSYKOR | 10.8K |
| | 0 | OPENFETCH | JDREIJE | 10.8K |
| | 0 | POSTMAN | MGRAFN | 10.8K |

### Description

Okta API tokens (SSWS tokens) are long-lived secrets used by applications and scripts to access the Okta API. Each token is tied to a specific user, and its permissions are inherited from that user's role assignments. For example, a token created by a Super Administrator has full access to all API endpoints. Because these tokens are often stored in plaintext in app configuration or environment variables, they are high-value targets. If exposed, attackers can perform administrative actions without interactive sign-in and bypass controls like MFA.

### References

References not available.

Export (.csv)

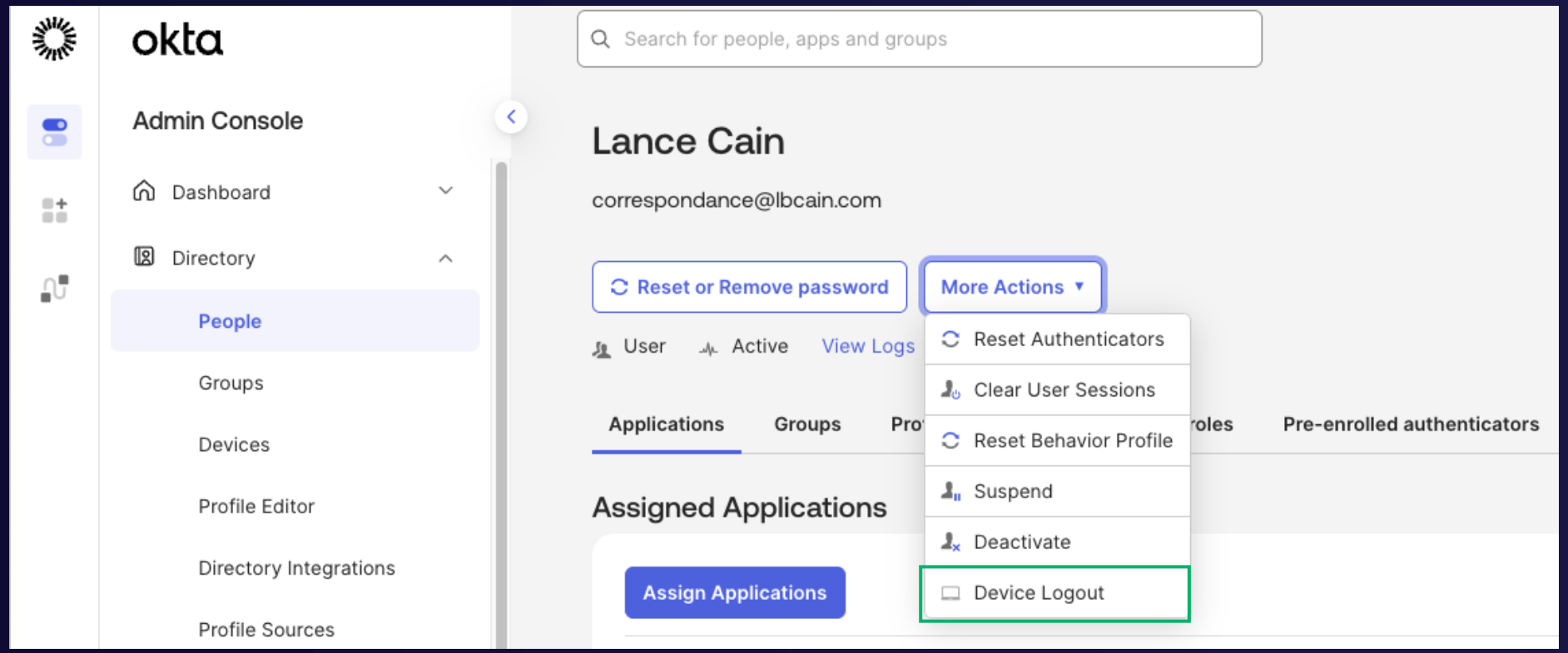
# Okta and Devices

## Targeting Okta Integrated Endpoints

# Common Okta Integrations

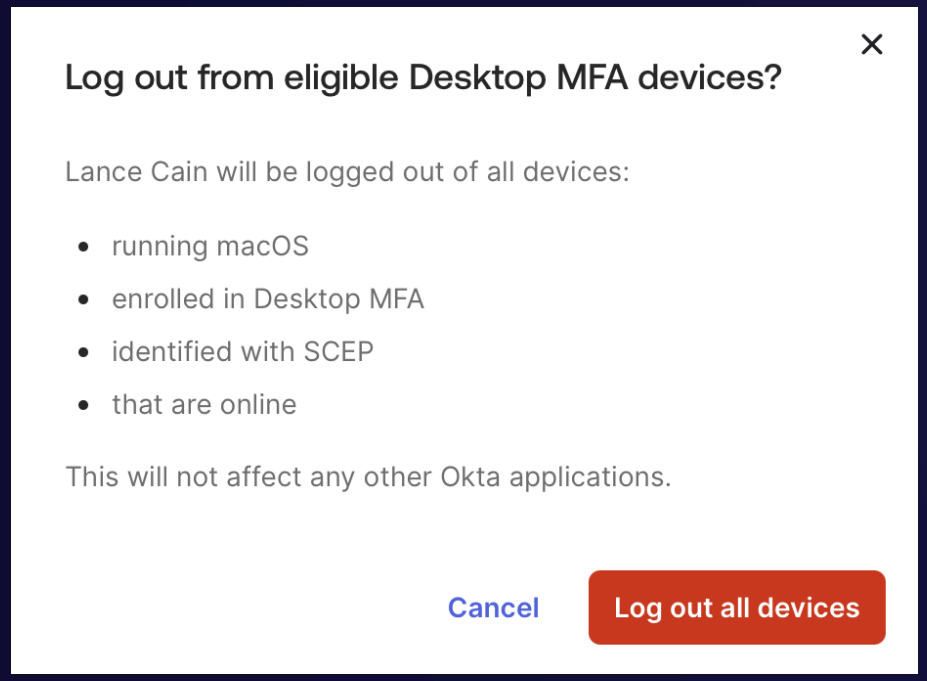
- Okta Desktop MFA
  - Okta Verify device prompts or TOTP during local logon
  - Recommended in administrator forums as an addition to Platform SSO
- Okta Application Authentication
  - Okta Verify on macOS can satisfy MFA requirements
  - Admins configure compliance checks using Device Assurance policies
    - Advanced Posture Checks expand checks beyond Okta default queries

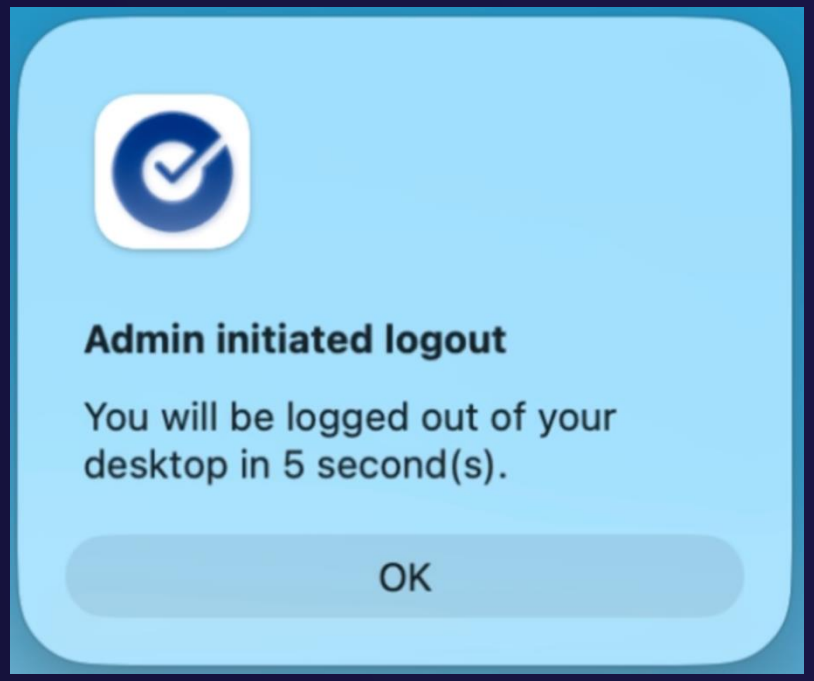
# Desktop MFA - DoS




⚠ Devices perform a best-effort poll every 15 minutes to receive the sign-out command from Okta. This action requires an active connection between the device and your Okta org.

# Desktop MFA - DoS



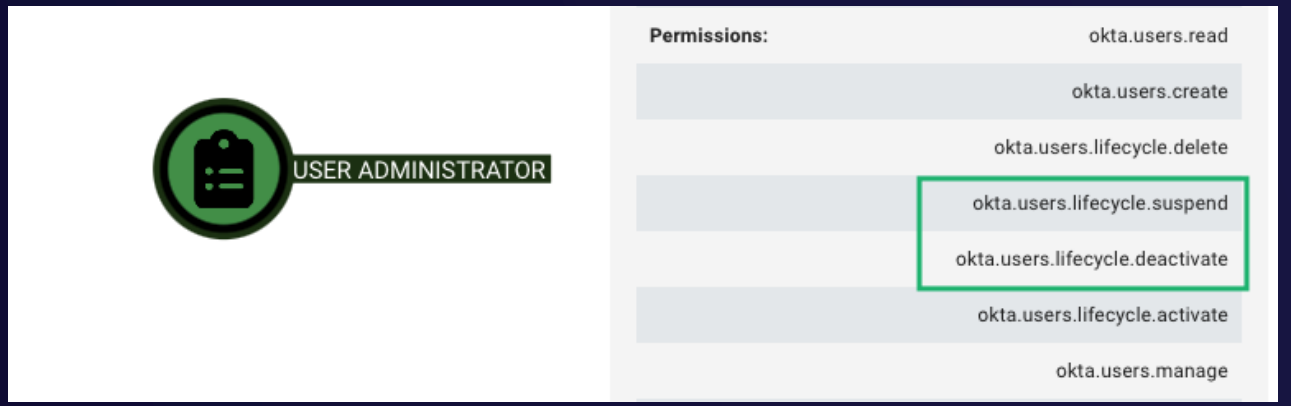



# Desktop MFA - DoS

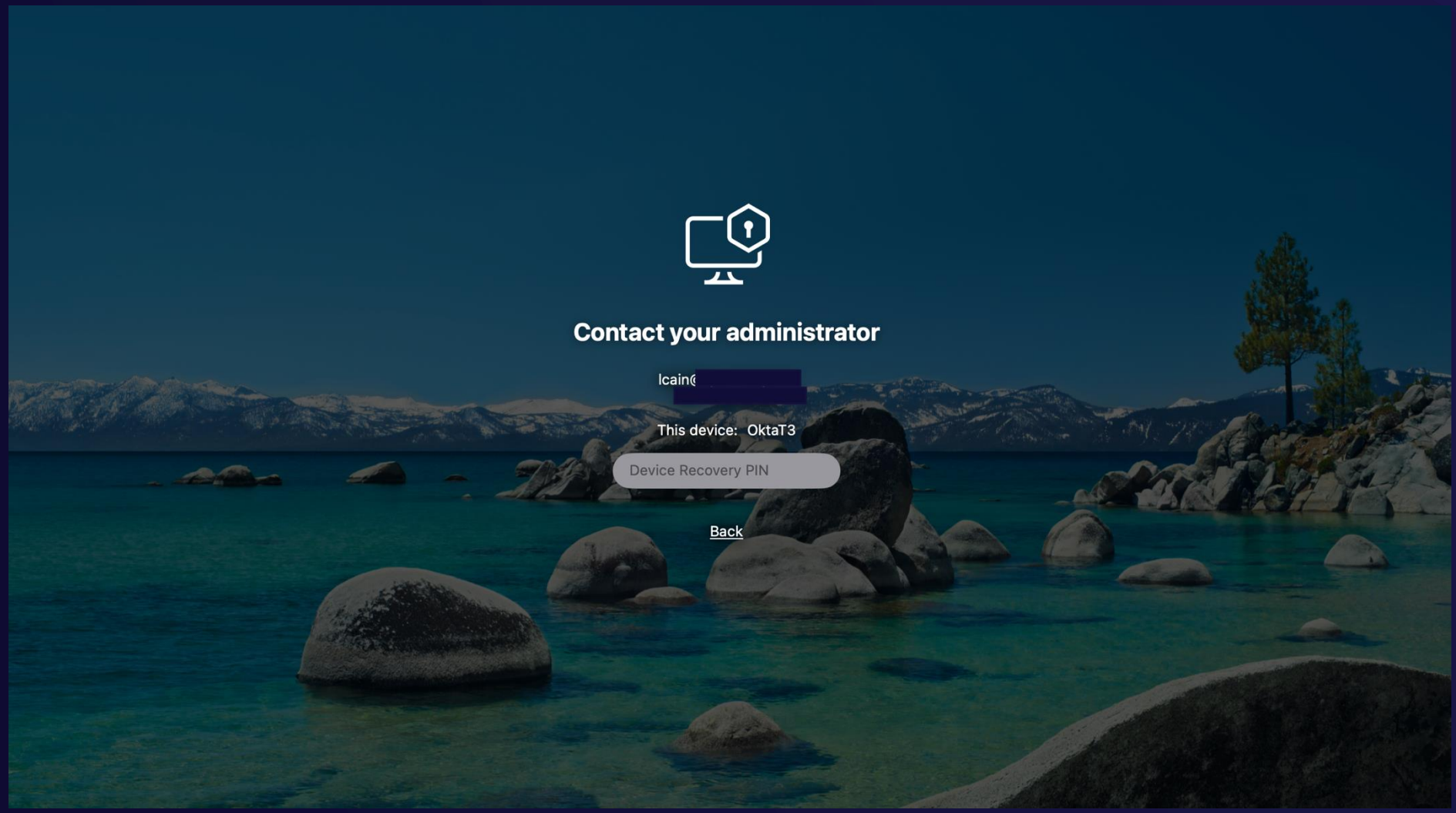
- Besides the manual method, how else can Device Logout be triggered?
  - Universal Logout
    - Okta Identity Threat Protection, if enabled, when risk status is elevated
    - User sessions and device factors are reset
  - When users are suspended or deactivated
    - Okta automates device logout on macOS

use this feature to handle employee lifecycle changes such as leaves of absence or offboarding. If a user is deactivated or suspended, Okta automatically signs the user out from all devices.

# Desktop MFA - DoS

# Desktop MFA - DoS

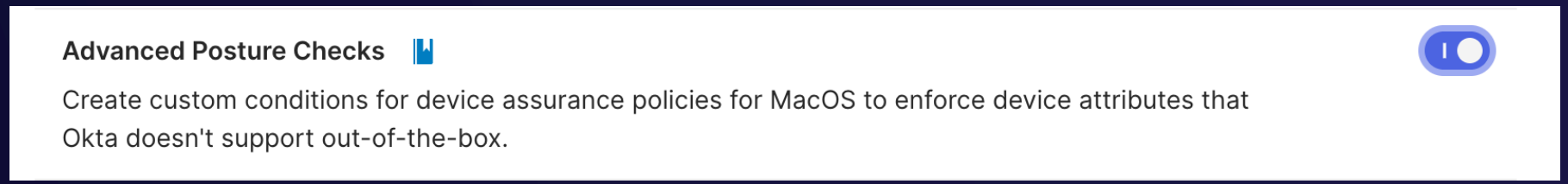


# Desktop MFA - DoS

- Recovery Options
  - Previously configured Device Recovery Pins
    - Admin re-enables and/or delivers recovery pins to users
    - *If a Super Admin was compromised, then only below two options*
  - MDM management removes Okta Verify with break-glass accounts
  - Contact Okta Support

# Advanced Posture Checks - Leaking Information

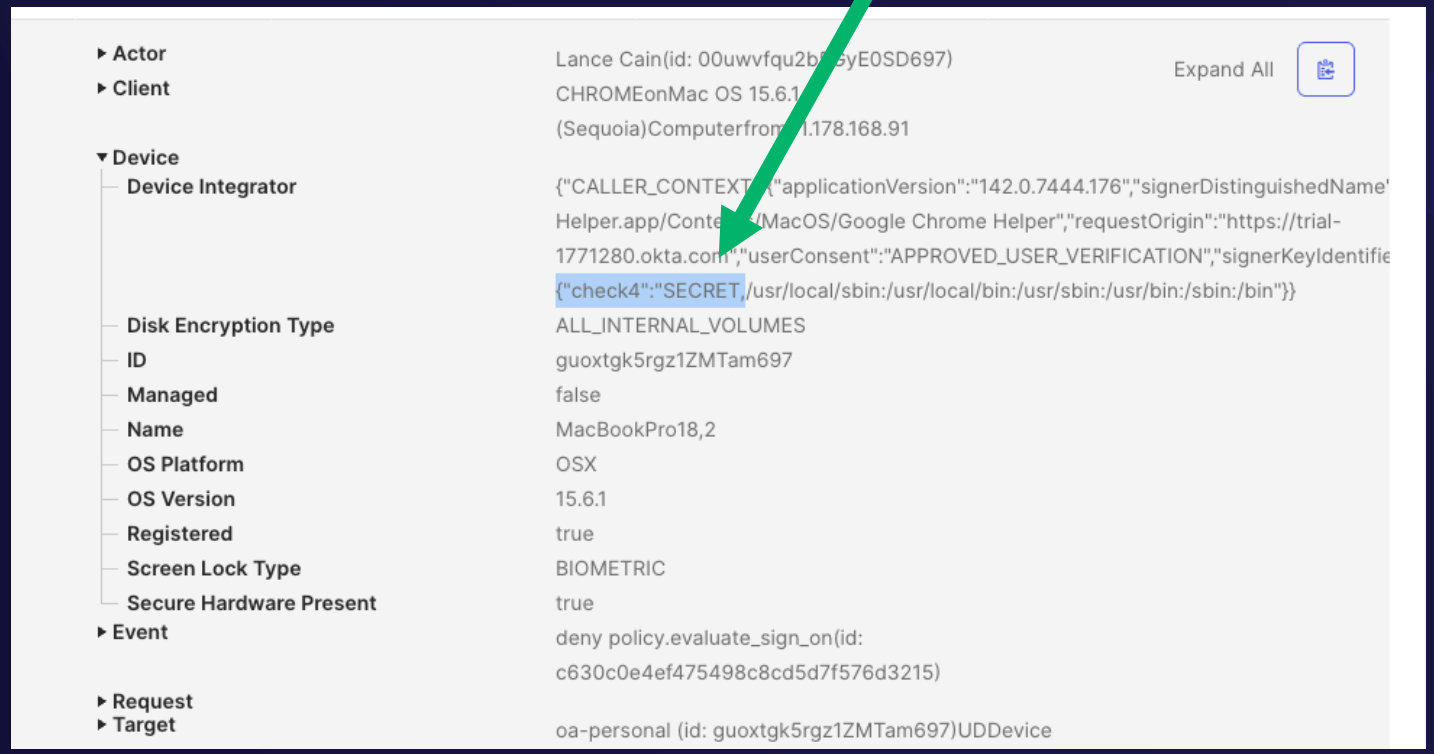
- Advanced Posture Checks evaluate system attributes beyond what Okta natively supports
- Uses OSQuery when authentication events are triggered
  - Enabled/Disabled
  - TextBox
- Supported queries documented - https://www.osquery.io/schema/5.18.1/
  - docker_container_envs
  - file
  - plist
  - processes
  - process_envs
  - shell_history
  - many more



# Advanced Posture Checks - Leaking Information

- Advanced posture checks log output to Okta's System Log
- We have successfully leaked:
  - Docker Container environment variable secrets
  - Docker Container process arguments
  - Plist contents of configuration files
  - Process command line arguments
  - Root shell history

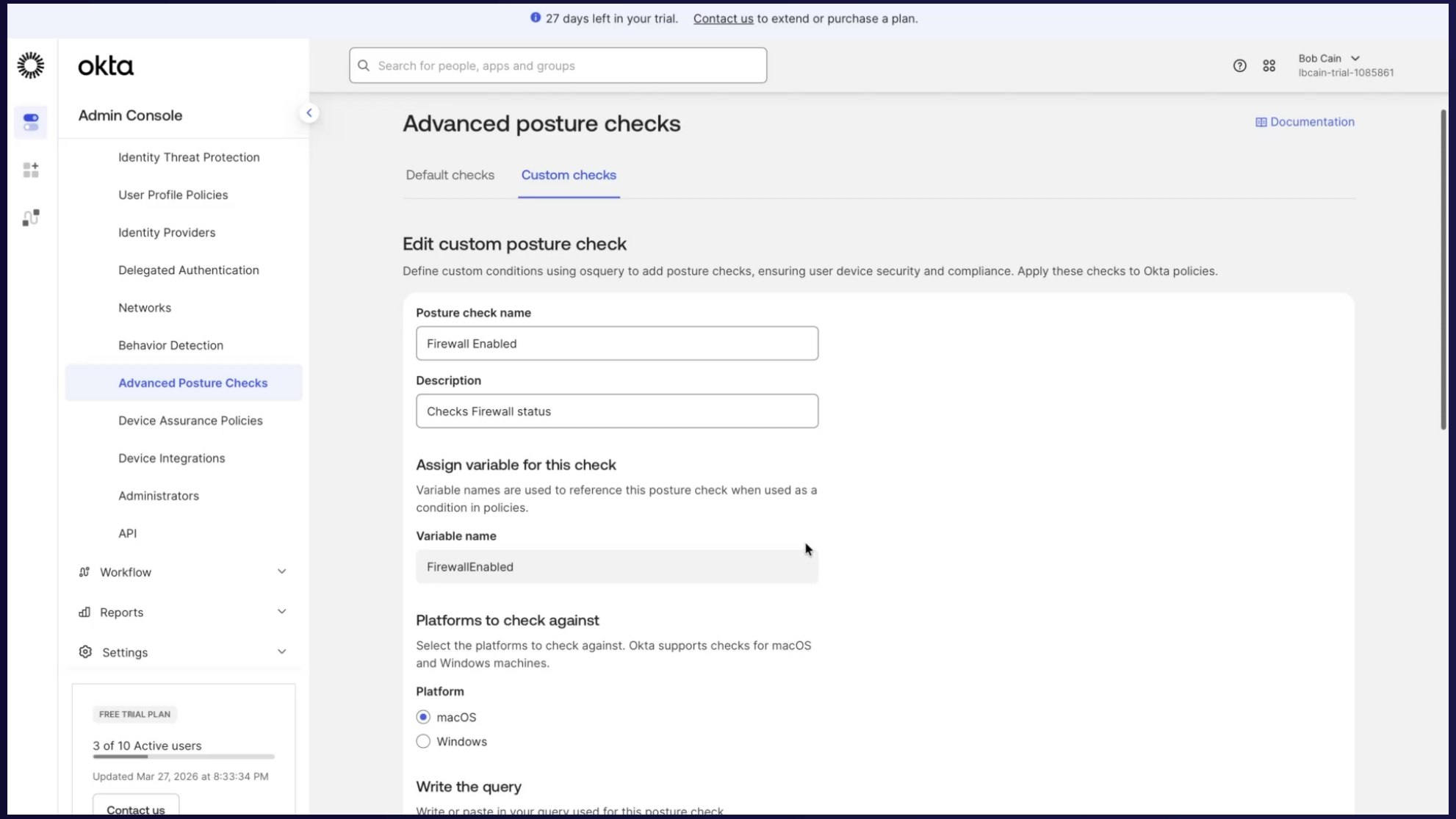
- docker run -it -e MYSECRET=SECRET busybox sh



# Advanced Posture Checks - Leaking Information

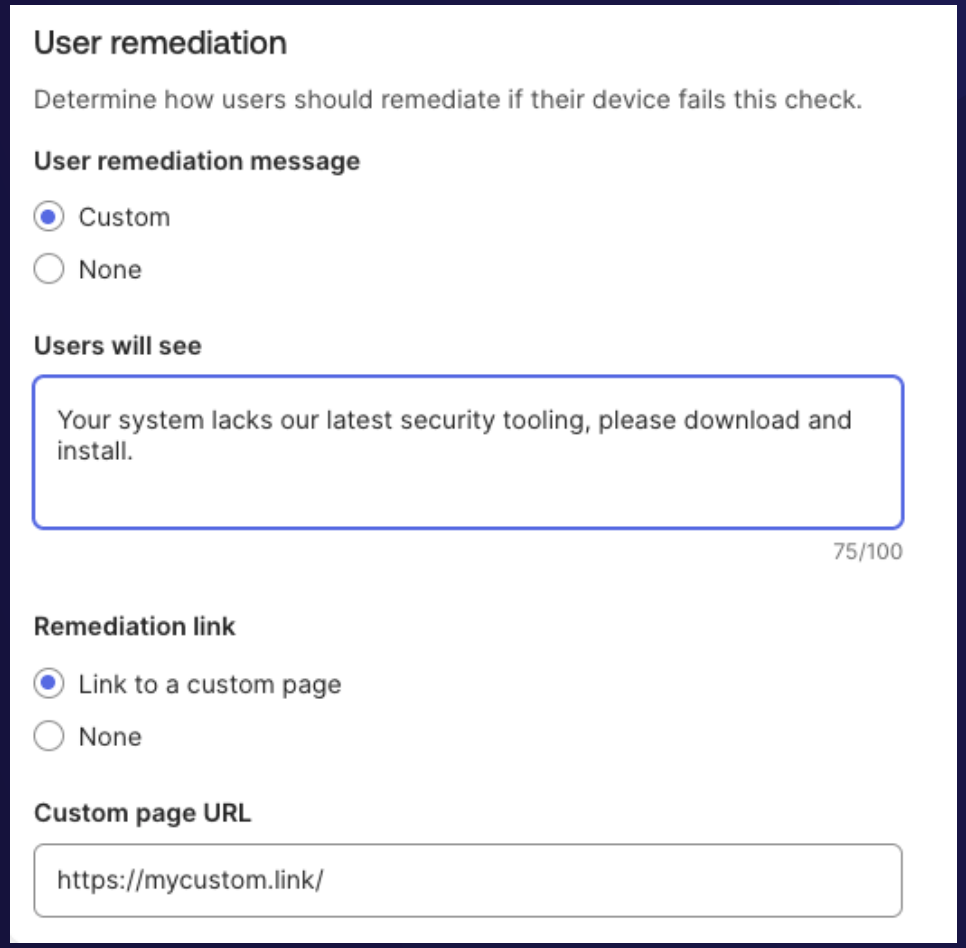

| | |
|---|---|
| **Last Seen by BloodHound:** | 2026-04-09 16:06 EDT (GMT-0400) |
| **Last Updated:** | 2026-04-09T20:02:52+00:00 |
| **Oauth Scopes:** | okta.logs.read |
| | okta.devicePostureChecks.manage |
| **Okta Domain:** | trial-1085861.okta.com |
| **Primary Kind:** | Okta_Application |

# Advanced Posture Checks - Leaking Information



# Advanced Posture Checks - Delivering Payloads

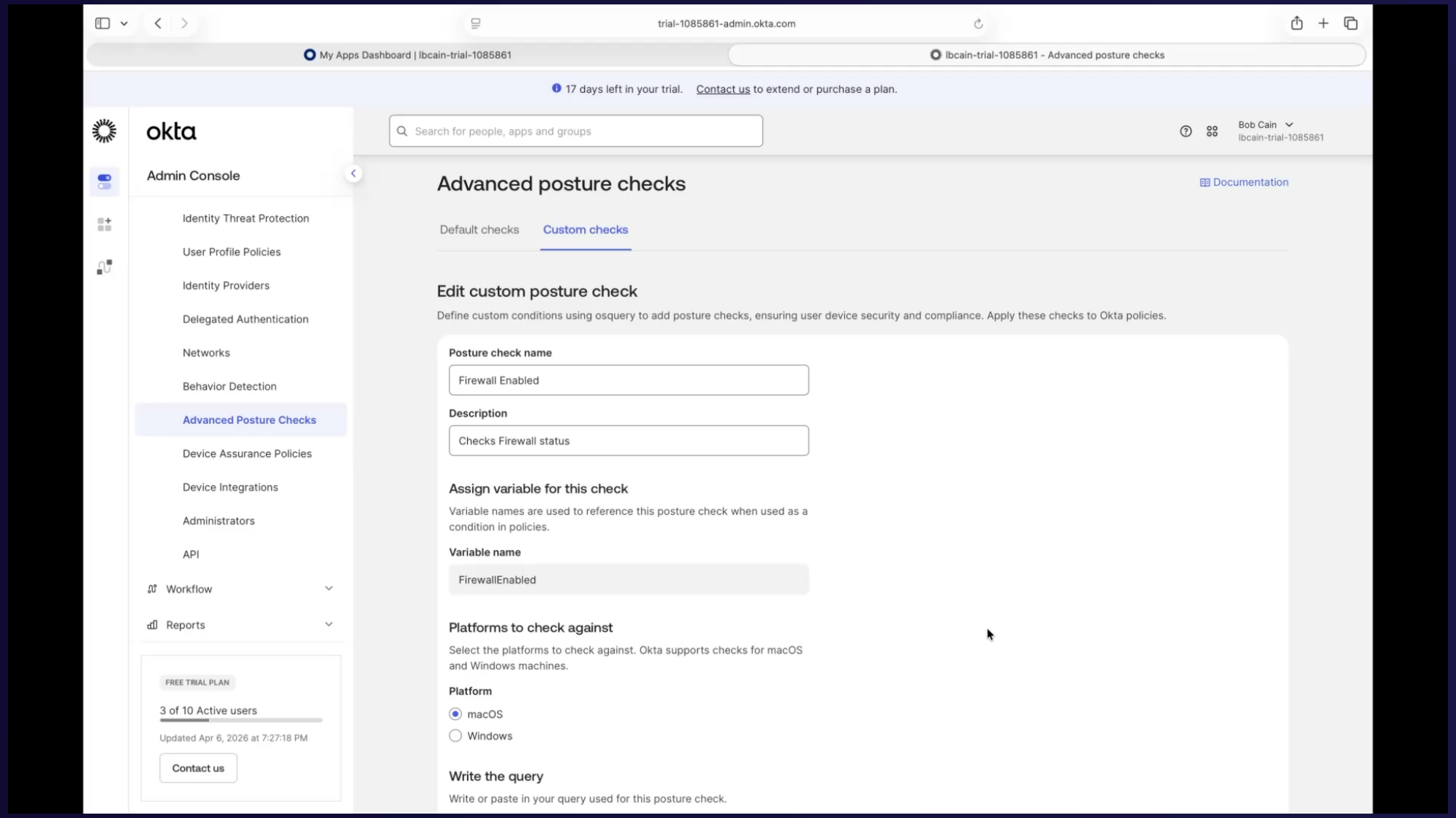
- Another capability of Advanced Posture Checks is presenting users with remediation steps
  - A custom error message to explain what failed
  - Can include **hyperlinks** to resources
- Compromise of posture check access means we can use Okta to direct users to download and run payloads
  - Prevent application access until payload is run



# Advanced Posture Checks - Delivering Payloads


| | |
|---|---|
| **Last Seen by BloodHound:** | 2026-04-09 16:06 EDT (GMT-0400) |
| **Last Updated:** | 2026-04-09T20:02:52+00:00 |
| **Oauth Scopes:** | ~~okta.logs.read~~ |
| | okta.devicePostureChecks.manage |
| **Okta Domain:** | trial-1085861.okta.com |
| **Primary Kind:** | Okta_Application |

# Advanced Posture Checks - Delivering Payloads



# Hybrid Attack Paths

JAMF Pro + Okta

# JAMF Pro + Okta - What's a JAMF Pro?

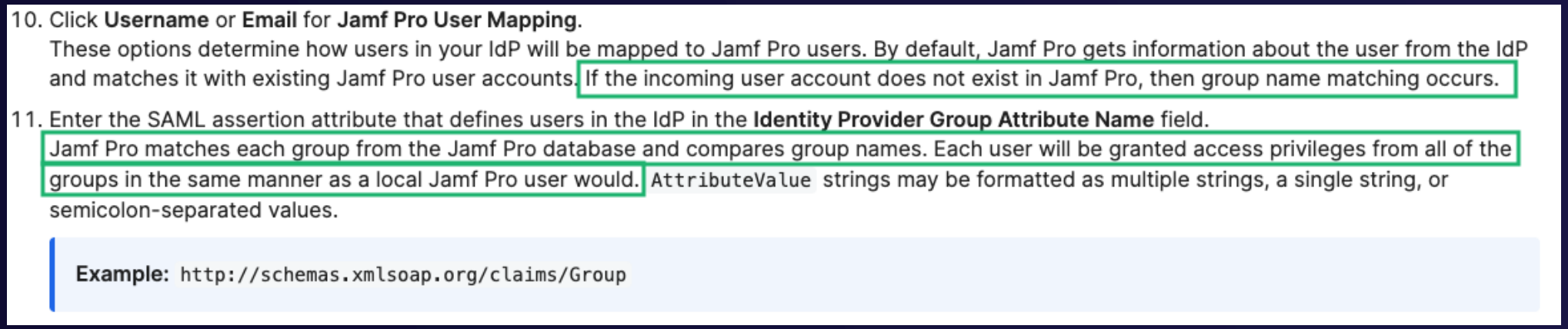
- Mobile Device Manager (MDM) that specializes in Apple device management
  - macOS
  - iOS
  - iPadOS
  - tvOS
- Administrators configure
  - policies
  - scripts
  - configuration profiles
  - applications
  - more

# JAMF Pro + Okta

- JAMF Pro supports SSO using OIDC and SAML
  - OIDC requires *jamfcloud.com* and a JAMF Account
- Authentication occurs in Okta with a JAMF Pro app integration
  - Allows MFA, device compliance, and additional checks
- Okta sends JAMF Pro user attributes for a logon session
  - Can map attributes to local JAMF Pro accounts
    - emails
    - usernames
- User group assignments can optionally be transferred as user attributes
  - Okta group names can match to groups in JAMF Pro

# Attack Okta -> JAMF

- Okta defaults **do** restrict low privileged admins from impacting privileged Okta Super Admins or groups
- Okta defaults **do not** restrict admins resetting user passwords or MFA of external privileged accounts
  - A compromised Okta Help Desk admin can reset a sync'd JAMF Pro administrator expanding a breach
- Application admins can assign new users to applications they control
  - A compromised App Admin can modify matched attributes sent to JAMF Pro
- Group admins and group membership admins can assign users to groups they control



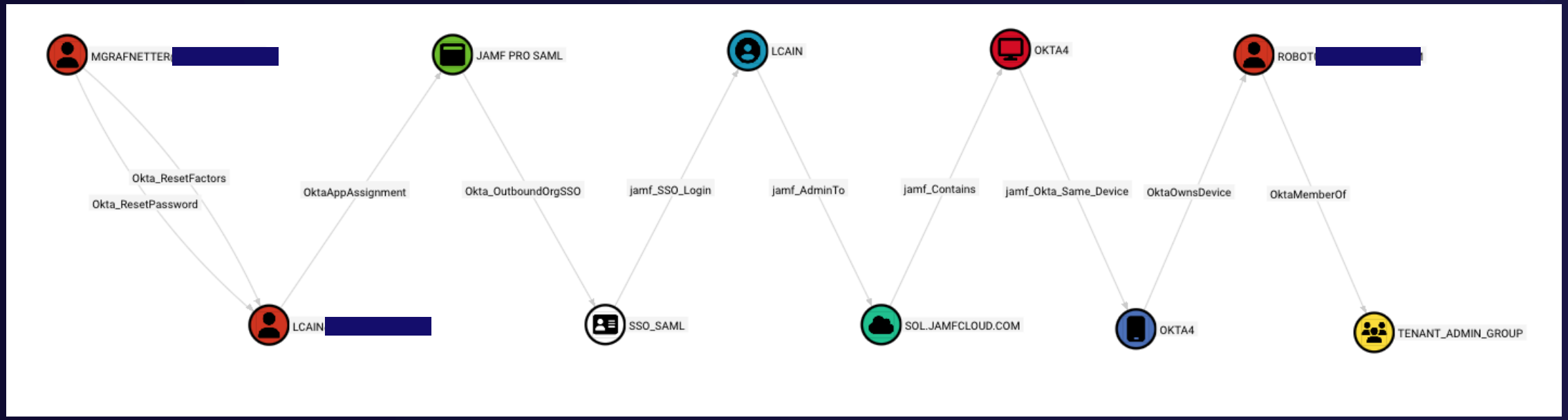
10. Click **Username** or **Email** for **Jamf Pro User Mapping**.
    These options determine how users in your IdP will be mapped to Jamf Pro users. By default, Jamf Pro gets information about the user from the IdP and matches it with existing Jamf Pro user accounts. If the incoming user account does not exist in Jamf Pro, then group name matching occurs.
11. Enter the SAML assertion attribute that defines users in the IdP in the **Identity Provider Group Attribute Name** field.
    Jamf Pro matches each group from the Jamf Pro database and compares group names. Each user will be granted access privileges from all of the groups in the same manner as a local Jamf Pro user would. `AttributeValue` strings may be formatted as multiple strings, a single string, or semicolon-separated values.

    **Example:** `http://schemas.xmlsoap.org/claims/Group`

# Attack JAMF -> Okta

- Compromise of any type of MDM administrator, API client, or security pipeline is a big win for attackers
- Access to JAMF Pro allows target enumeration and payload execution
  - Eve toolkit
- We have leveraged JAMF Pro on red team assessments to pivot to privileged admins
  - Perform internal reconnaissance to identify Super Admins
  - Execute code on managed systems to
    - Access session tokens
    - Enable remote services
    - Exfiltrate API keys

# Okta and JAMF Pro Hybrid Attack Path

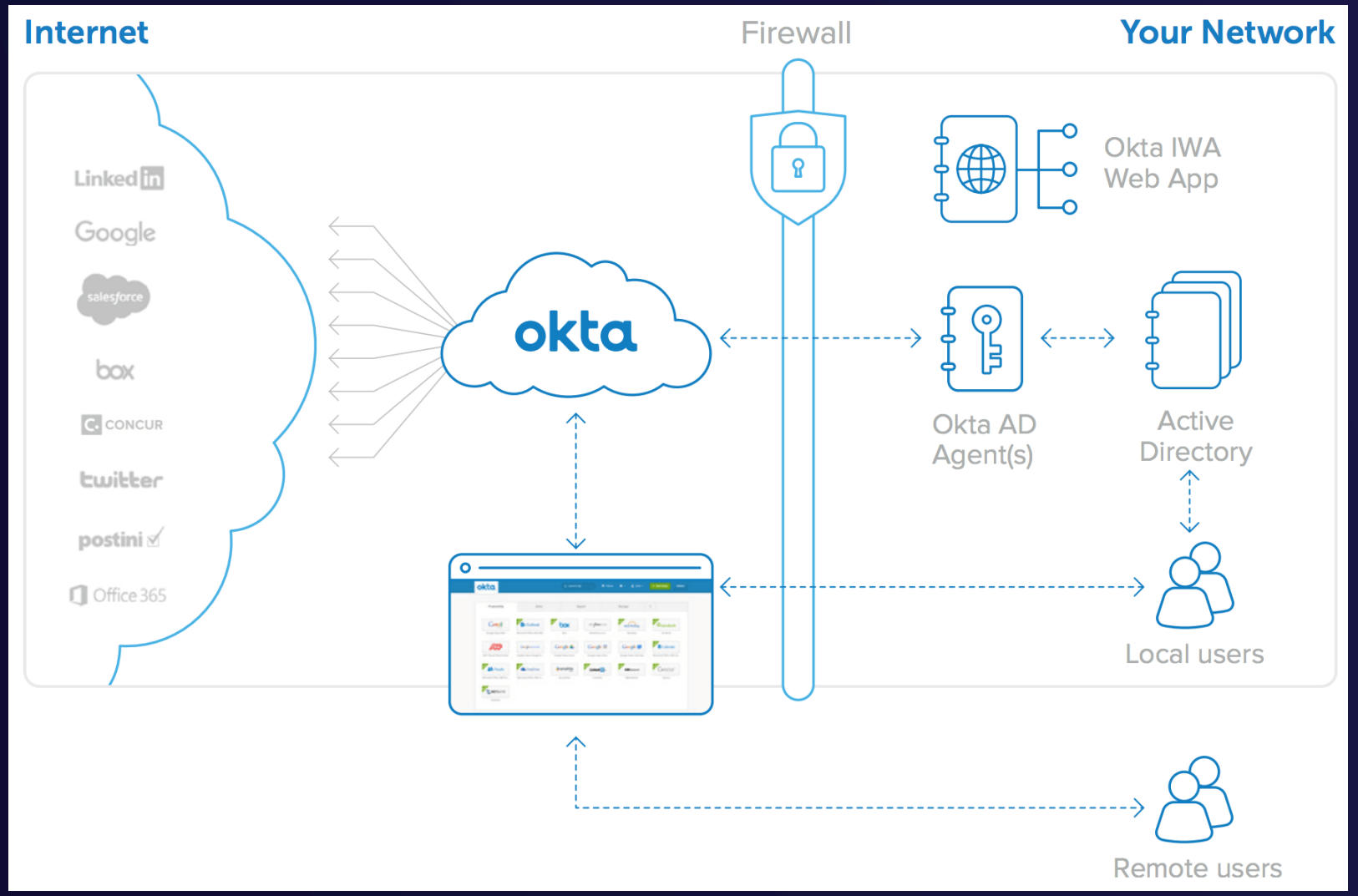


# Hybrid Attack Paths

Active Directory

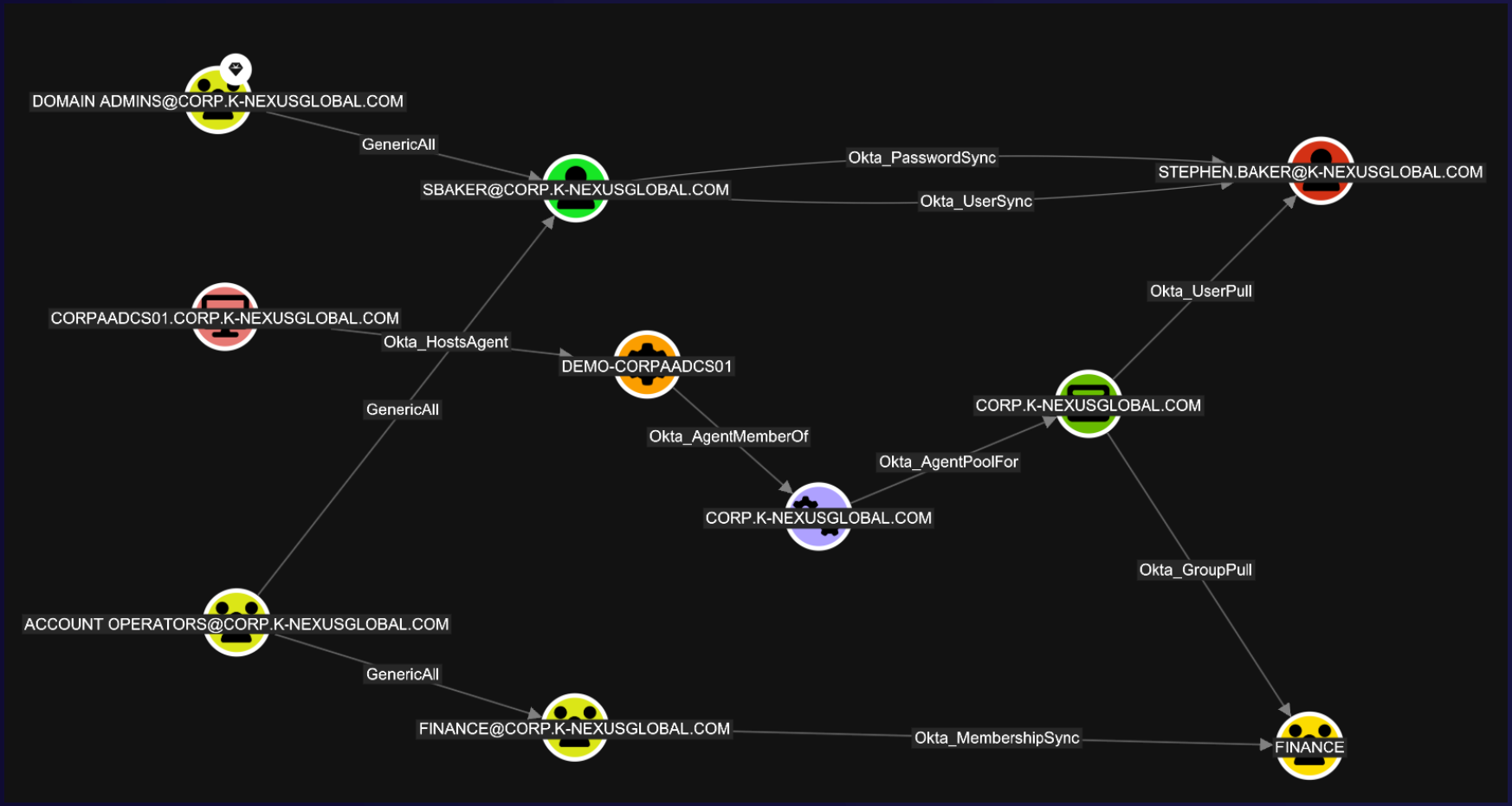
# Okta AD Agents

## Hybrid AD Deployment



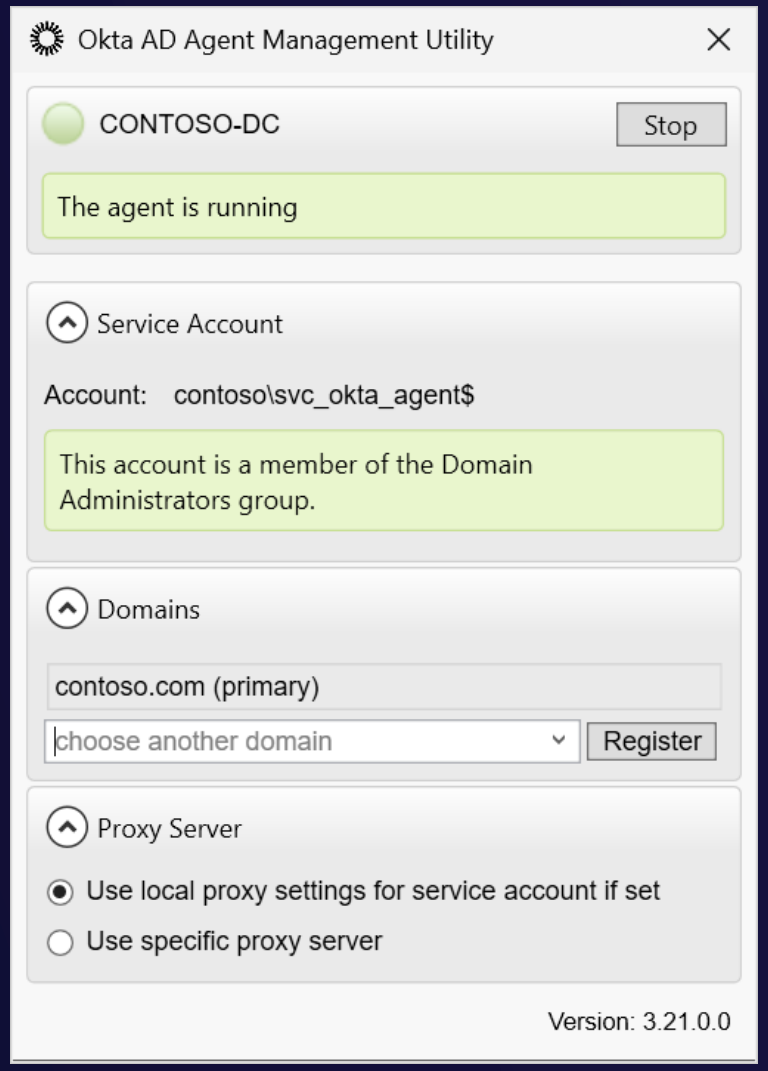
# Okta AD Agents

## Synchronization Model



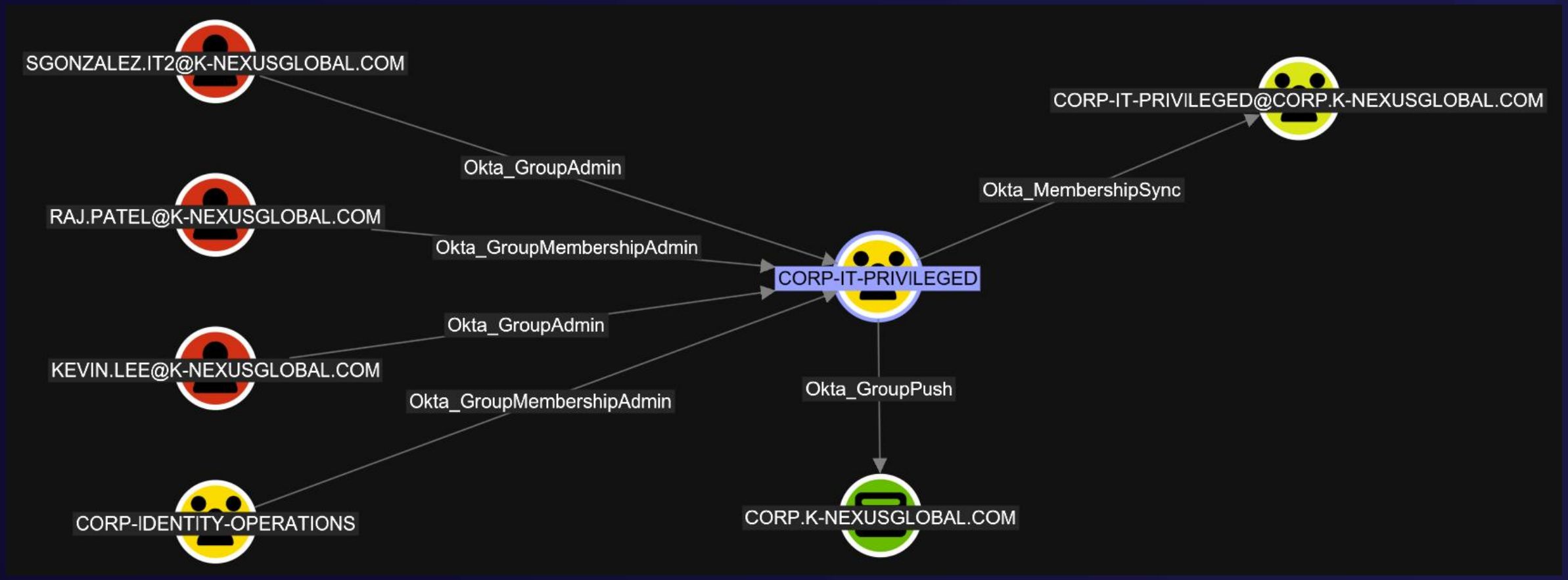
# Okta AD Agents

## Object Writeback Permissions



# Okta AD Agents

## Group Writeback



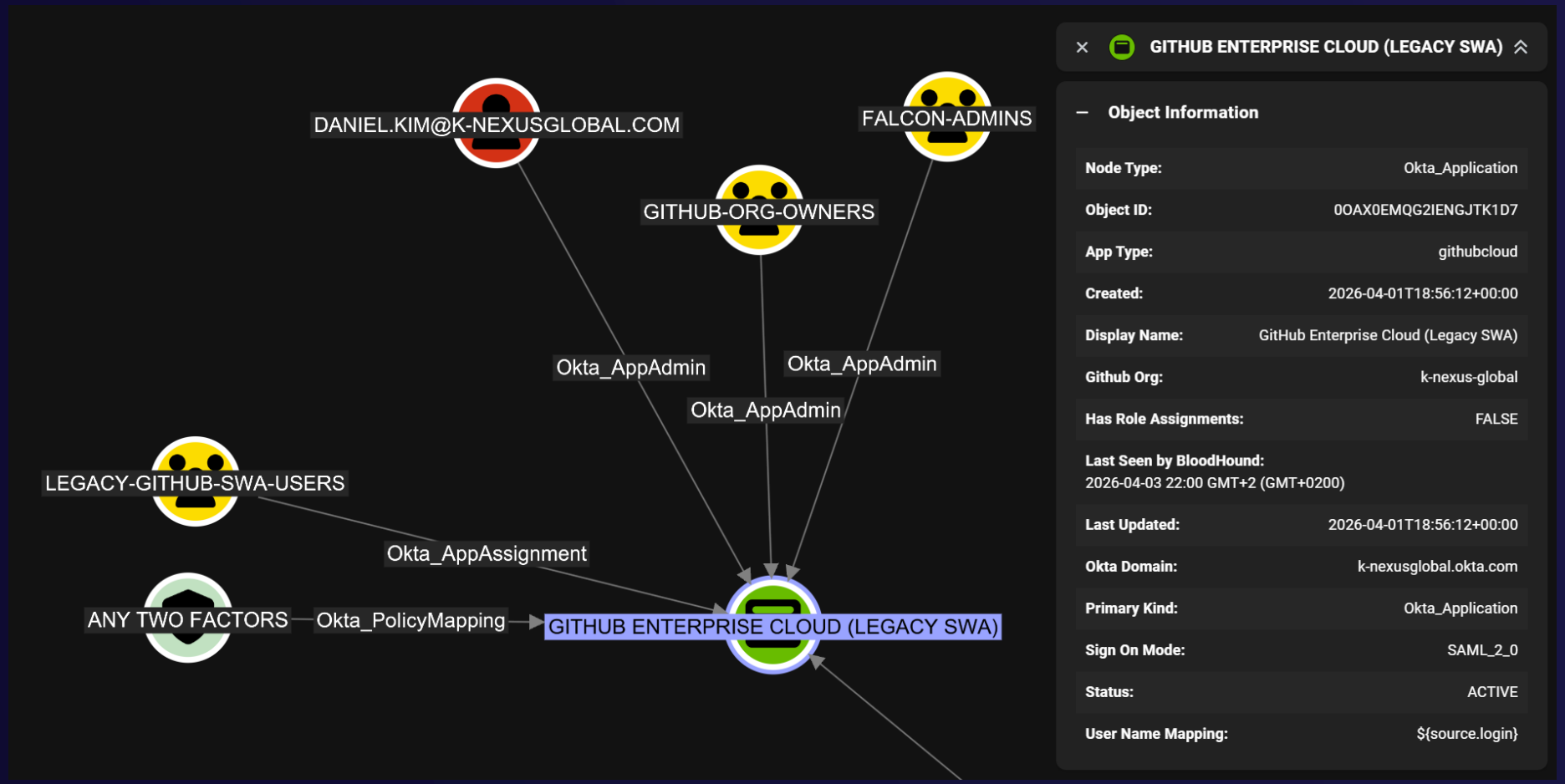
# Agentless Desktop SSO

# Hybrid Attack Paths

GitHub

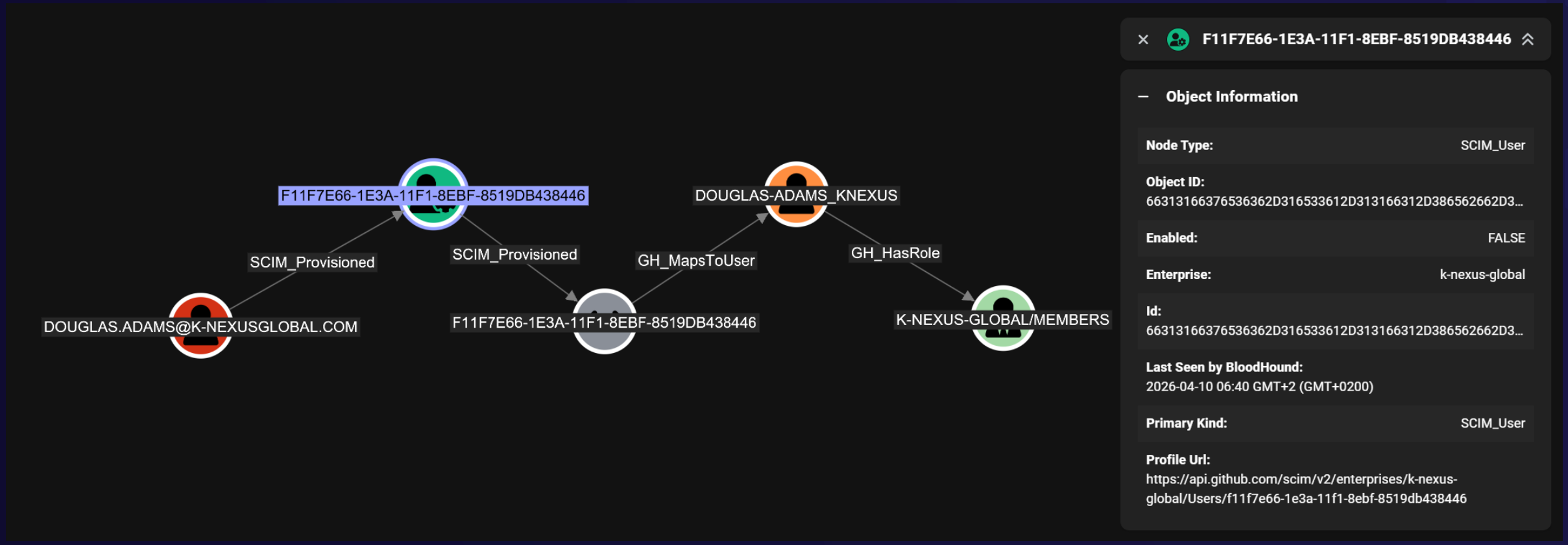
# Hybrid Attack Paths

## GitHub Application Configuration



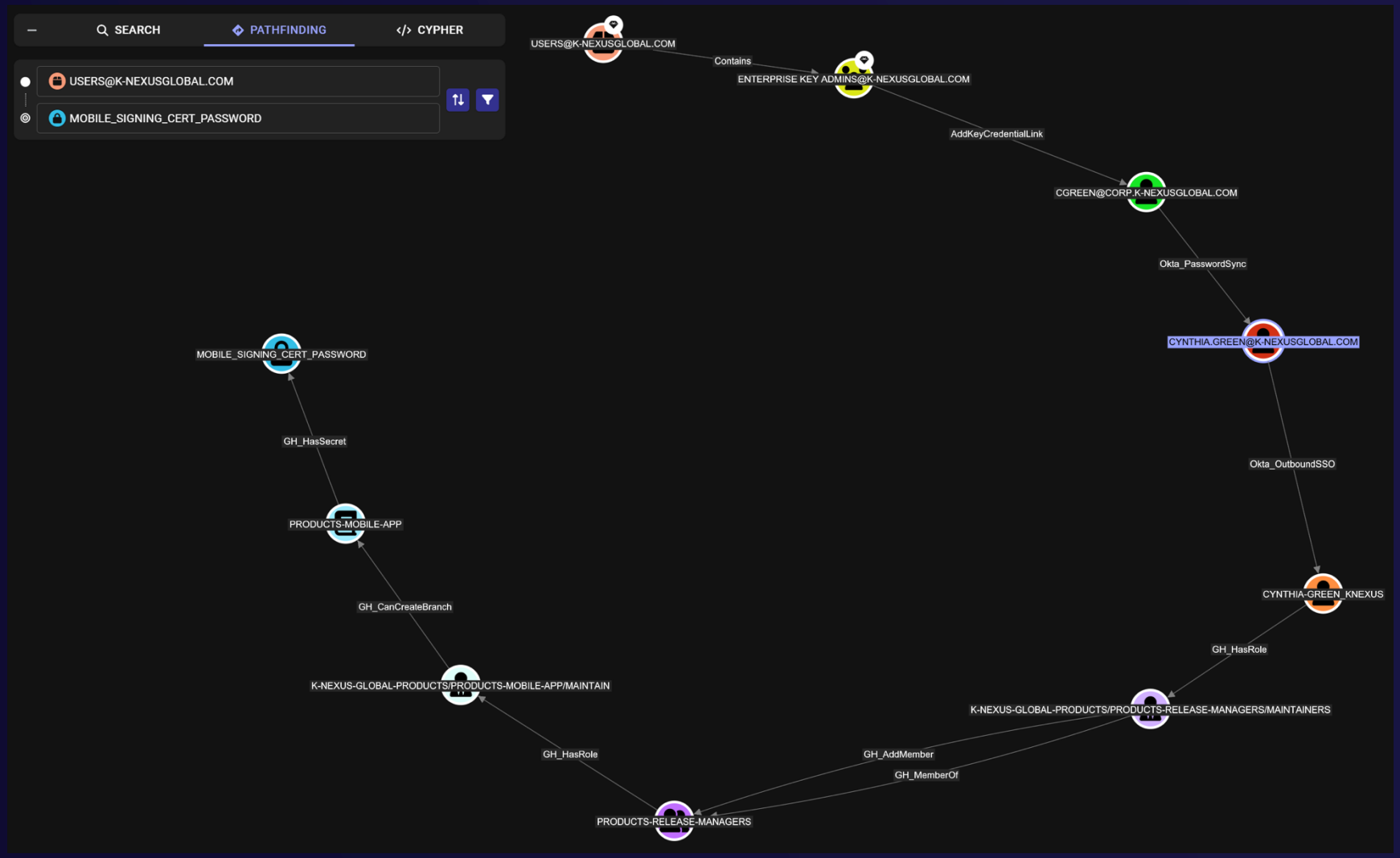
# Hybrid Attack Paths

## GitHub User Synchronization / SCIM



# Hybrid Attack Paths

## Active Directory User -> Okta -> GitHub Secret

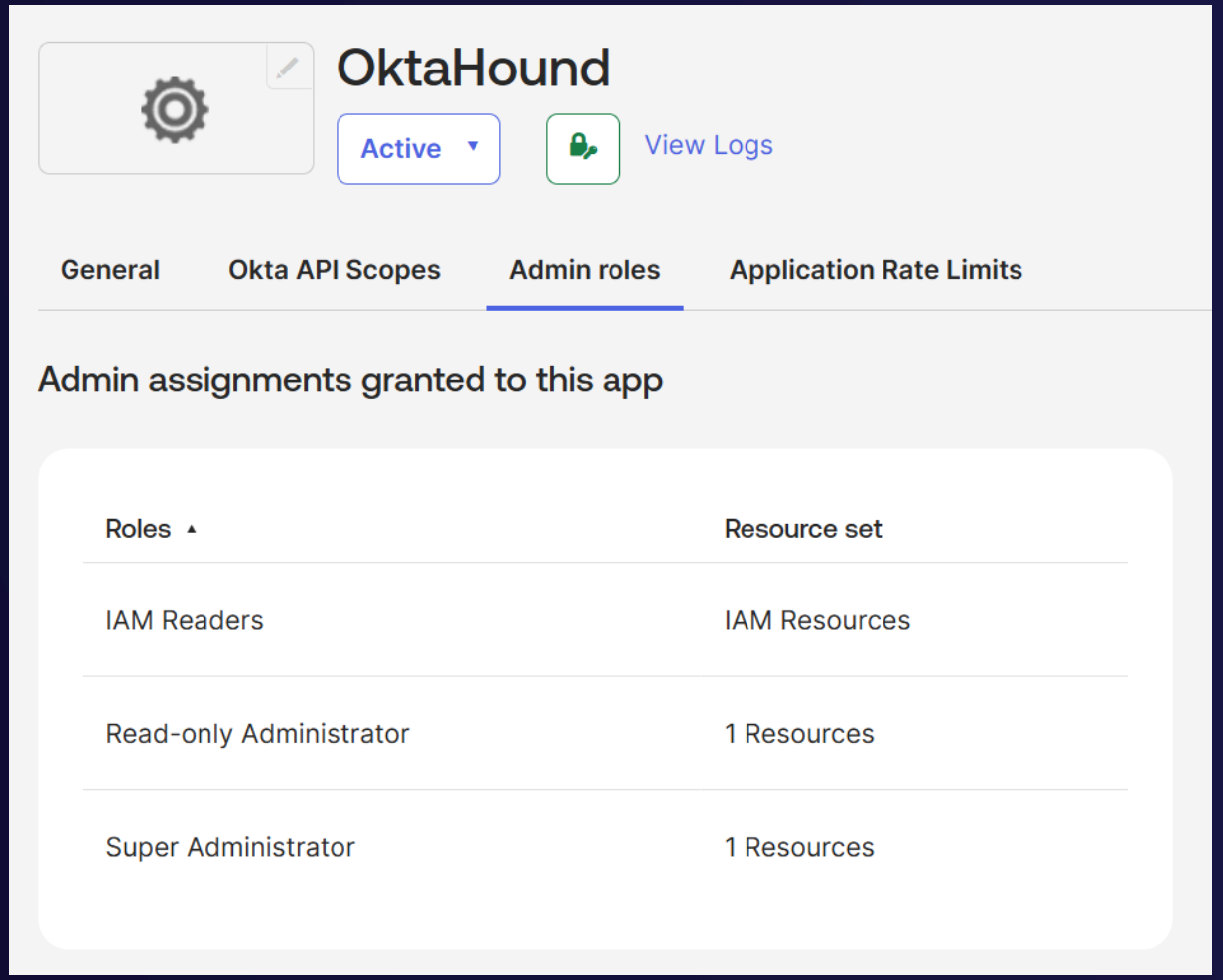


# BloodHound OpenGraph Collectors

## OktaHound

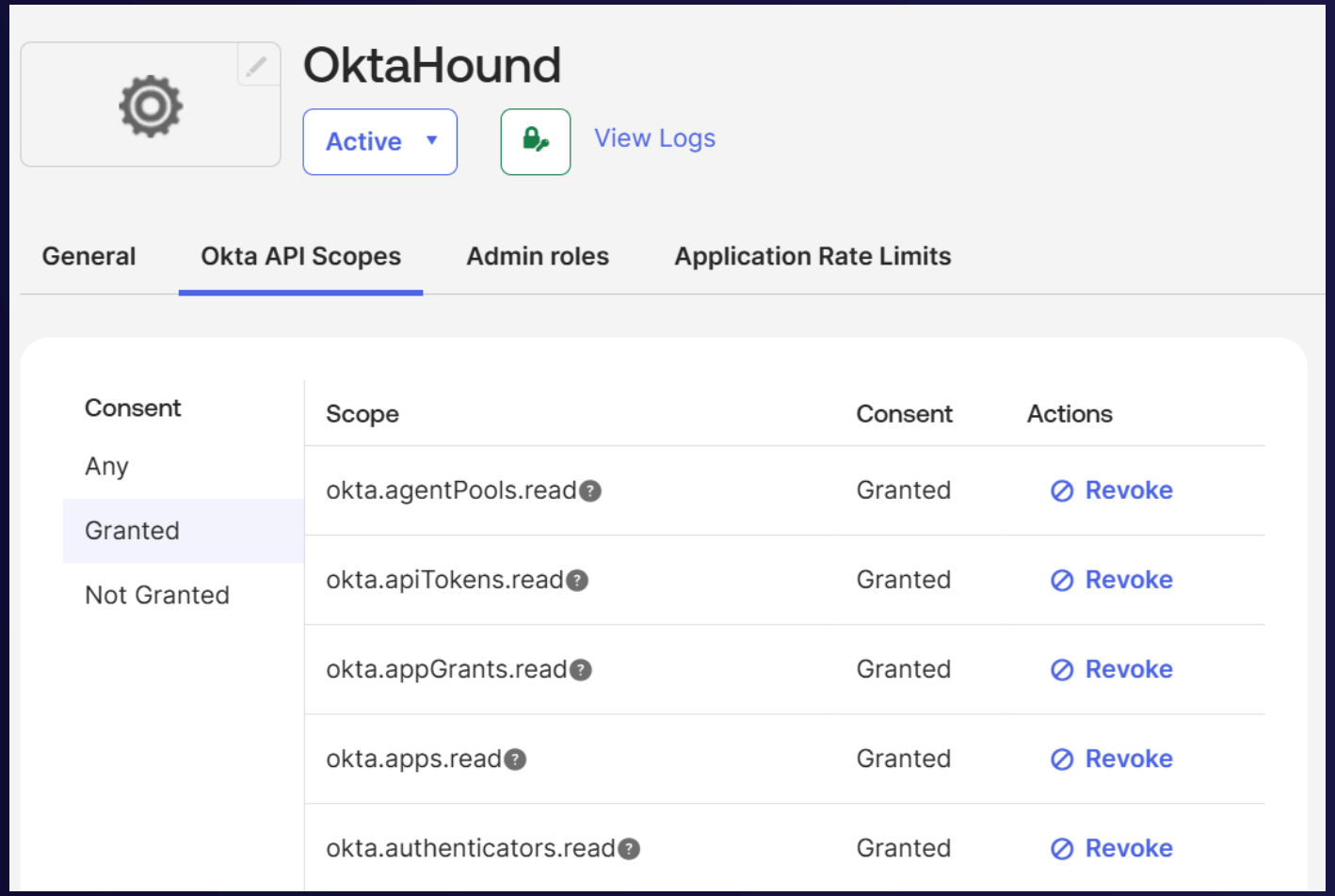
# App Registration

## Role Assignment



# App Registration

## Okta Management API Scope Grants

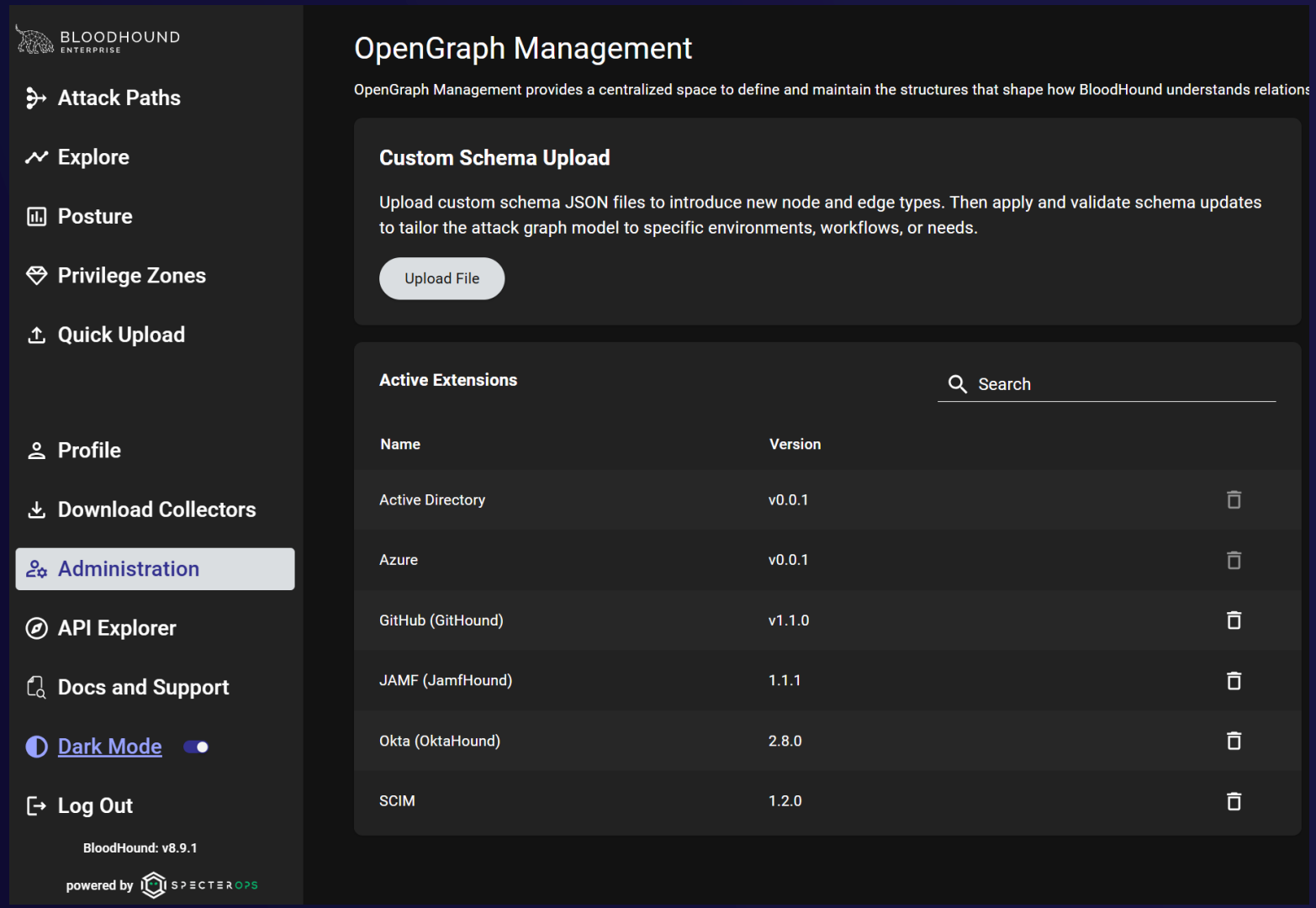


# Data Collection

Ubuntu

```
michael@GRAY:~/oktahound$ ./oktahound collect --zip --skip-mfa --verbosity Debug
[16:33:53] dbug: OktaHound[0] Using existing output directory at /home/michael/oktahound/output.
[16:33:53] info: OktaHound[0] Loading Okta configuration...
[16:33:53] warn: OktaHound[0] Using API Token (SSWS) authentication. It is recommended to use OAuth 2.0 with an API Service App for better security and auditing.
[16:33:53] dbug: OktaHound[0] Authenticating to https://integrator-5415459.okta.com using SSWS...
[16:33:53] info: OktaHound[0] Fetching Okta organization information...
[16:33:54] dbug: OktaHound[0] Initializing the graph...
[16:33:54] info: OktaHound[0] Fetching users...
[16:33:54] info: OktaHound[0] Skipping user authentication factors collection.
[16:33:54] info: OktaHound[0] Fetching applications...
[16:33:54] info: OktaHound[0] Fetching API service integrations...
[16:33:54] info: OktaHound[0] Fetching identity providers...
[16:33:54] info: OktaHound[0] Fetching custom roles...
[16:33:54] info: OktaHound[0] Fetching policies...
[16:33:54] dbug: OktaHound[0] Fetching OKTA_SIGN_ON policies...
[16:33:54] info: OktaHound[0] Fetching devices...
[16:33:54] info: OktaHound[0] Fetching realms...
[16:33:55] info: OktaHound[0] Fetching built-in roles...
[16:33:55] dbug: OktaHound[0] Fetching info about the built-in role API_ACCESS_MANAGEMENT_ADMIN...
[16:33:55] dbug: OktaHound[0] Fetching info about the built-in role GROUP_MEMBERSHIP_ADMIN...
[16:33:55] dbug: OktaHound[0] Fetching info about the built-in role APP_ADMIN...
[16:33:55] info: OktaHound[0] Fetching custom resource sets...
[16:33:55] info: OktaHound[0] Fetching API tokens...
[16:33:55] info: OktaHound[0] Fetching agent pools...
[16:33:55] info: OktaHound[0] Fetching authorization servers...
[16:33:55] dbug: OktaHound[0] Processing policy Default Policy (00pw0o8igxWLqAYQr697)...
[16:33:55] dbug: OktaHound[0] Fetching PASSWORD policies...
[16:33:55] dbug: OktaHound[0] Processing resource set Test resource set (iamwldyxudjNf2nnk697)...
[16:33:55] dbug: OktaHound[0] Processing resource set IAM Resources (iamwwdsp2oNvvqr0s697)...
[16:33:55] dbug: OktaHound[0] Processing resource set Workflows Resource Set (WORKFLOWS_IAM_POLICY)...
[16:33:55] info: OktaHound[0] Successfully processed 3 resource sets.
[16:33:55] dbug: OktaHound[0] Processing API token OktaHound (00T3ugczs7Im9Qeju697)...
[16:33:55] dbug: OktaHound[0] Processing API token Postman (00T36fk75smeJybKx697)...
```

# Extension Schema



# Extension Schema

## Nodes

| Icon | Node Kind |
|---|---|
| | Okta_Agent |
| | Okta_AgentPool |
| | Okta_ApiServiceIntegration |
| | Okta_ApiToken |
| | Okta_Application |
| | Okta_AuthorizationServer |
| | Okta_ClientSecret |
| | Okta_CustomRole |
| | Okta_Device |
| | Okta_Group |
| | Okta_IdentityProvider |
| | Okta_JWK |
| | Okta_Organization |
| | Okta_Policy |
| | Okta_Realm |
| | Okta_ResourceSet |
| | Okta_Role |
| | Okta_RoleAssignment |
| | Okta_User |

## Edges

| Relationship Kind | Traversable |
|---|---|
| Okta_AddMember | ✓ |
| Okta_AgentMemberOf | ✓ |
| Okta_AgentPoolFor | ✓ |
| Okta_ApiTokenFor | ✓ |
| Okta_AppAdmin | ✓ |
| Okta_AppAssignment | ✗ |
| Okta_Contains | ✓ |
| Okta_CreatorOf | ✗ |
| Okta_DeviceOf | ✗ |
| Okta_GroupAdmin | ✓ |
| Okta_GroupMembershipAdmin | ✓ |
| Okta_GroupPull | ✓ |
| Okta_GroupPush | ✗ |
| Okta_HasRole | ✗ |
| Okta_HasRoleAssignment | ✗ |
| Okta_HelpDeskAdmin | ✓ |
| Okta_HostsAgent | ✓ |
| Okta_IdentityProviderFor | ✓ |
| Okta_IdpGroupAssignment | ✗ |
| Okta_InboundOrgSSO | ✓ |
| Okta_InboundSSO | ✓ |
| Okta_KerberosSSO | ✓ |
| Okta_KeyOf | ✓ |
| Okta_ManageApp | ✓ |
| Okta_ManagerOf | ✗ |
| Okta_MemberOf | ✓ |
| Okta_MembershipSync | ✓ |
| Okta_MobileAdmin | ✓ |
| Okta_OrgAdmin | ✓ |
| Okta_OrgSWA | ✗ |
| Okta_OutboundOrgSSO | ✓ |
| Okta_OutboundSSO | ✓ |
| Okta_PasswordSync | ✓ |
| Okta_PolicyMapping | ✗ |
| Okta_ReadClientSecret | ✓ |
| Okta_ReadPasswordUpdates | ✓ |
| Okta_RealmContains | ✓ |
| Okta_ResetFactors | ✓ |
| Okta_ResetPassword | ✓ |
| Okta_ResourceSetContains | ✓ |
| Okta_ScopedTo | ✗ |
| Okta_SecretOf | ✓ |
| Okta_SuperAdmin | ✓ |
| Okta_SWA | ✗ |
| Okta_UserPull | ✗ |
| Okta_UserPush | ✗ |
| Okta_UserSync | ✗ |

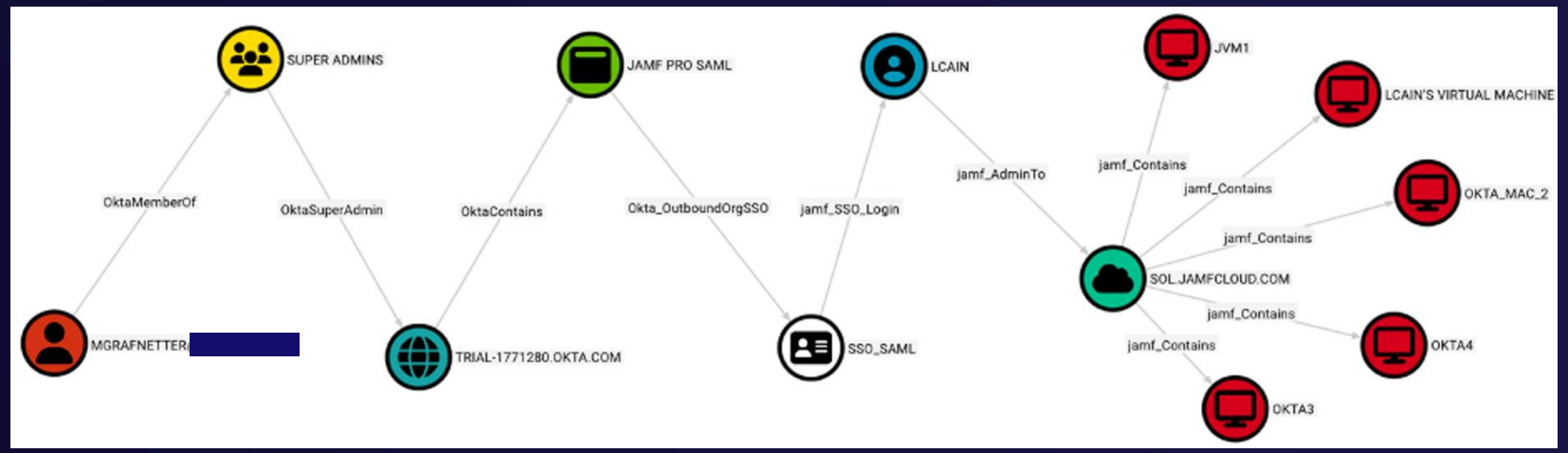
# BloodHound OpenGraph Collectors

JamfHound

# JamfHound

- OpenGraph collector for JAMF Pro cloud and on-site tenants
- Released at BlackHat USA 2025
- Continued researching attack paths in JAMF Pro
- As of March 2026, the latest version of the collector now supports new Okta hybrid paths
  - SSOIntegration Node
  - jamf_Okta_Same_Device Edge
  - jamf_SSO_Login Edge

# JamfHound


SUPER ADMINS
JAMF PRO SAML
LCAIN
JVM1
LCAIN'S VIRTUAL MACHINE
OktaMemberOf
OktaSuperAdmin
OktaContains
Okta_OutboundOrgSSO
jamf_SSO_Login
jamf_AdminTo
jamf_Contains
jamf_Contains
OKTA_MAC_2
jamf_Contains
MGRAFNETTER
TRIAL-1771280.OKTA.COM
SSO_SAML
SOL.JAMFCLOUD.COM
jamf_Contains
jamf_Contains
OKTA4
OKTA3


OKTA4
jamf_Okta_Same_Device
OKTA4

# Key Takeaways

- While defaults are secure, Okta doesn't inherently mitigate additions such as app integrations and custom roles
- MDM SSO support = expanded and conjoined attack surface with IDPs
  - Always a high-value target for attackers
- Okta protects what it knows about within the org (Super Admins), but does not know the privileges of external identities
  - i.e. JAMF Pro admins, GitHub admins, or AD domain admins
- ***It's critical to know what the full attack surface looks like***
  - OpenGraph updates to BloodHound reveal complex attack paths across different technologies

# Resources

- OktaHound OpenGraph Collector - https://github.com/SpecterOps/OktaHound
- JamfHound OpenGraph Collector - https://github.com/SpecterOps/JamfHound
- GitHound OpenGraph Collector - https://github.com/SpecterOps/GitHound
- SCIM OpenGraph Schema Extension - https://github.com/SpecterOps/bloodhound-scim-extension
- Eve JAMF Pro Exploitation Toolkit - https://github.com/RobotOperator/Eve
- JamfHound Okta Updates - https://specterops.io/blog/2026/03/31/jamfhound-v1-1-update-sso-attack-paths-and-okta-additions/

# Thank you