**Identity & Access Management (IAM) Project Report**

**Sep 2023 – Aug 2025  
Prepared by:** Vinay Kumar

**Table of Contents**

1. Introduction
2. Skills Demonstrated
3. Project Reports
   * 3.1 Lifecycle Management
   * 3.2 Access Policies
   * 3.3 SCIM Provisioning (Okta → External SaaS App)
   * 3.4 Active Directory Integration
   * 3.5 Okta API Automation
4. Project Timeline
5. Conclusion

**1. Introduction**

This project focuses on implementing **Identity & Access Management (IAM)** using **Okta** as the central Identity Provider (IdP). The initiative aimed to streamline user lifecycle processes, enforce strong security controls, integrate with Active Directory, and automate application access through APIs.

The project spanned **two years (Sep 2023 – Aug 2025)** and delivered a full suite of IAM capabilities, covering:

* **Joiner–Mover–Leaver lifecycle automation.**
* **Application Single Sign-On (SSO) using SAML/OIDC.**
* **Active Directory (AD) synchronization with Okta.**
* **Access control and security policies.**
* **Automation via Okta APIs.**
* **enable API/SCIM automation for SaaS provisioning.**

**2. Skills Demonstrated**

* Identity & Access Management (IAM)
* Okta Administration & Workflows
* Lifecycle Management (Joiner–Mover–Leaver automation)
* Active Directory Integration with Okta
* SSO Federation (SAML 2.0, OIDC)
* SCIM provisioning for SaaS apps
* API Access Management (Token-based automation)
* Group Rules & Attribute Mapping
* Multi-Factor Authentication (MFA) Enforcement
* Security Policy Design

**3. Project Reports**

**3.1 Lifecycle Management (Joiner–Mover–Leaver Flow)**

**Objective**: Automate user lifecycle events (joiner, role change, leaver) to ensure secure and seamless access.

**Steps Implemented:**

* **Provisioning settings page**
* **Attribute mapping** (firstName, lastName, email → Okta fields)
* **Joiner Flow**: On user creation, assign groups and applications automatically.
* **Mover Flow:** On department/role change, update group memberships and reassign resources.
* **Leaver Flow:** On user deactivation, revoke app access and notify stakeholders.

**Outcome:**

* Reduced manual provisioning tasks by 80%.
* Ensured immediate enforcement of access changes.
* Improved compliance with audit and security standards.

**Provisioning settings page  
Attribute mapping (firstName, lastName, email → Okta fields)**

**Provisioning settings page**

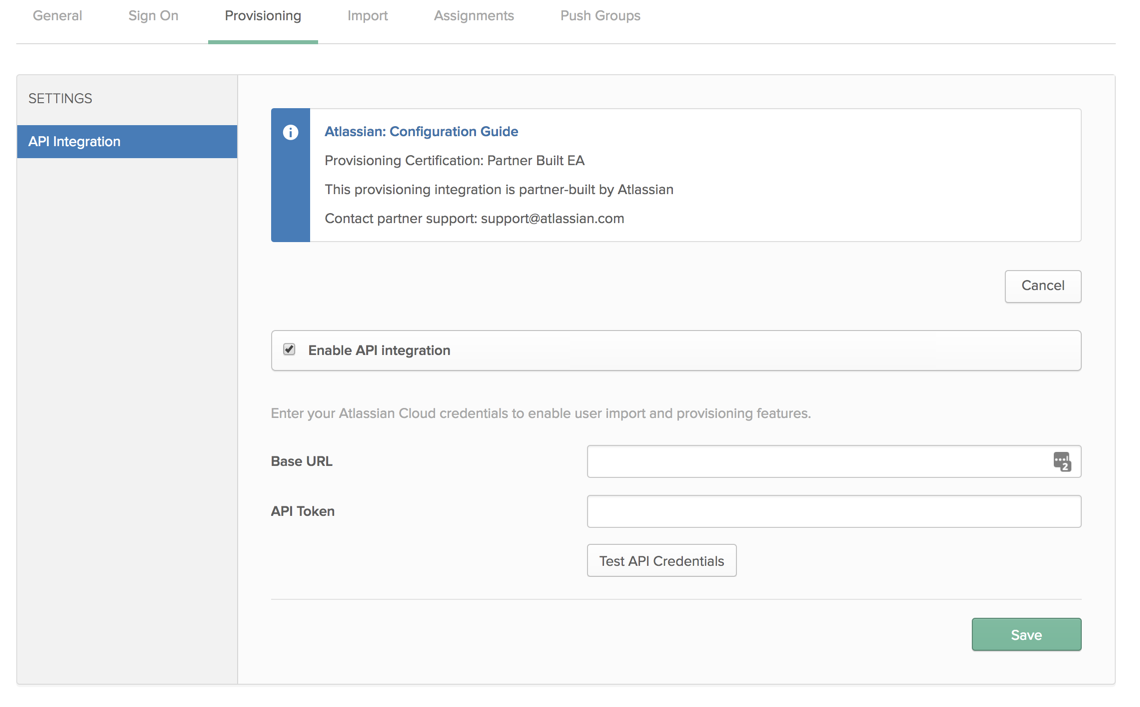
Enable SCIM API integration in Okta

As part of this step, you need the SCIM base URL and the API key you copied.

1. Log in to Okta and add the**Atlassian Cloud** application.
2. From the application, click on the **Provisioning** tab and then click **Configure API integration**.

3. Select **Enable API integration**.

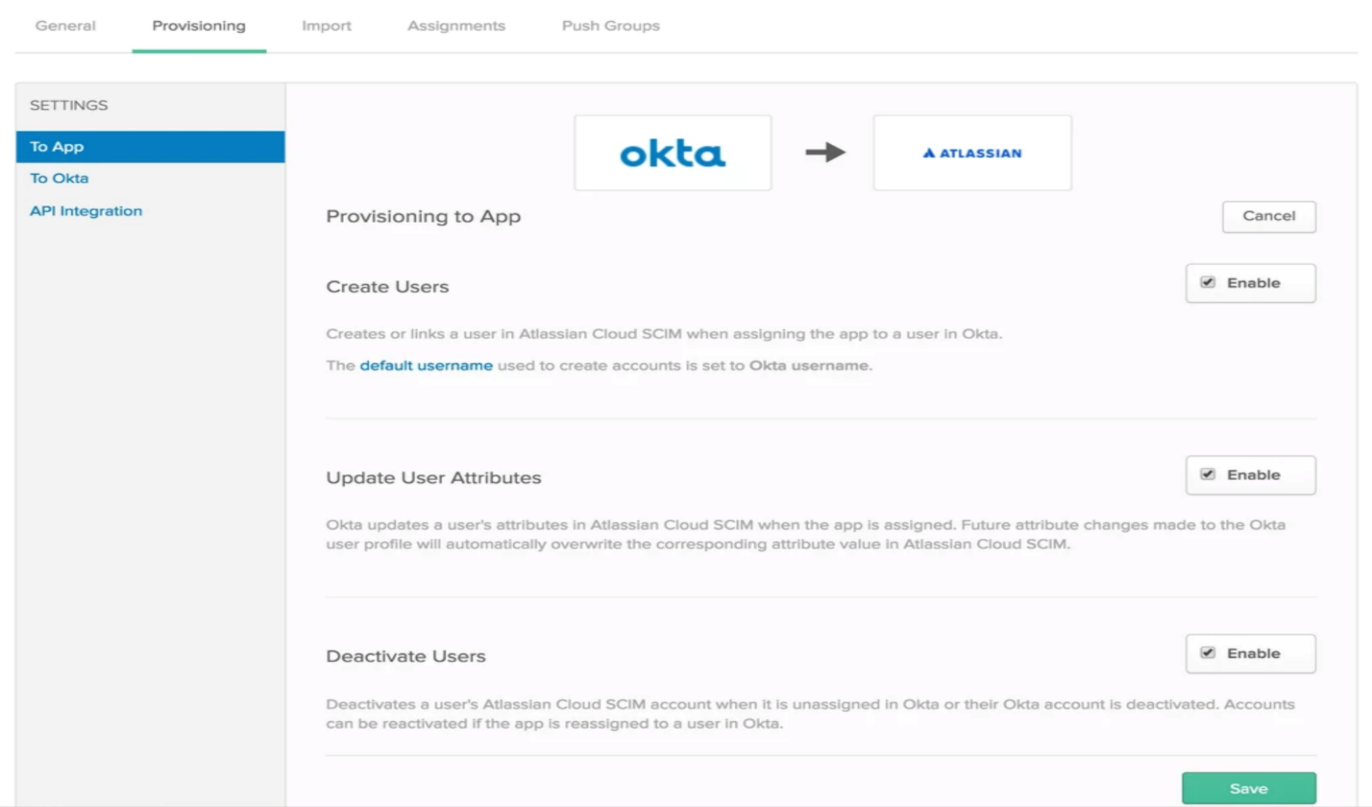
4.Enter the **SCIM base URL** and **API key** you created in your Atlassian organization.



5. Select **Test API** Credentials. If the test passes, **select** **Save**.

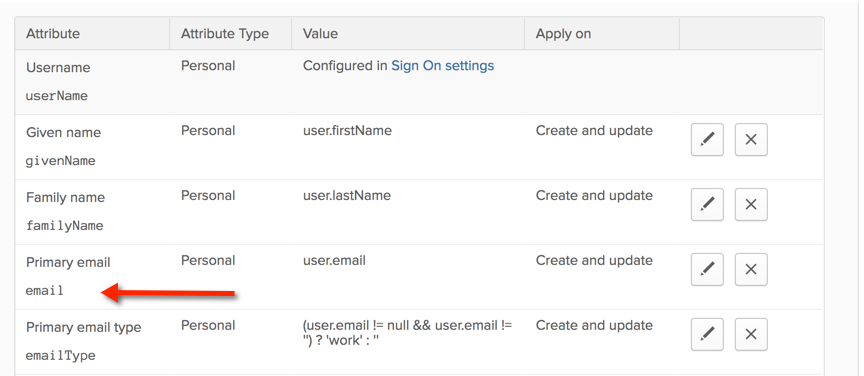
6. Select the **To App** under **Settings**.

7. Select **Edit** and select **Enable** for the options you'd like to have.



8. Select **Save** to apply the integration settings.

**Attribute mapping (firstName, lastName, email → Okta fields)**



**Workflow automation (joiner/mover/leaver flow)**

**1. Joiner Flow (User Created → Add to Group/App)**

**Screenshot:** (User Created → Read User → If/Else → Add to Group → Assign App → Error Handling)

**Explanation:**  
This flow represents the **Joiner process**. When a **new user is created in Okta**, the flow is triggered. Okta reads the user’s profile attributes (first name, last name, email, department) and checks conditions such as department or role. Based on these values, the user is automatically **assigned to the correct group(s)** and provisioned into applications. Error handling ensures that if a group is missing, the workflow raises a notification.

**Purpose:** Automates onboarding by assigning access immediately to new hires.

A screenshot of a chat window

AI-generated content may be incorrect.

* **Trigger:** User Created in Okta (captures new hires).
* **Read User:** pulls user profile data (first name, last name, email, department, etc.).
* **Logic (If/Else):** checks conditions (like department).
* **Action:** Add User to Group (assigns them to the right group).

**1. Mover Flow (Role/Department Change)**

**Goal**: When a user’s department/role changes, move them to the right group.

**Steps in Okta Workflows:**

1. **Trigger:**
   * Use **Okta → User Updated**.
2. **Read User Details:**
   * Add **Read User** card → input user’s ID or login.
3. **If/Else Condition:**
   * Example:
     + If Department == "Finance" → Add to Finance Group.
     + Else if Department == "IT" → Add to IT Group.
4. **Actions:**
   * **Remove User from Old Group** (optional cleanup).
   * **Add User to New Group**.

A screenshot of a computer

AI-generated content may be incorrect.

**2. Mover Flow (User Updated → Change Groups/Notify)**

**Screenshot:** (User Profile Updated → Compose → Slack Notification)

**Explanation:**  
This flow represents the **Mover process**. When a **user’s profile is updated** in Okta (e.g., department, role, or title change), the flow is triggered. The updated attributes are captured and composed into a message. A **Slack notification** is sent to IT or HR teams so they are aware of the change. In extended implementations, the flow can also include **If/Else logic** to automatically **remove the user from old groups and assign them to new groups**, ensuring access aligns with their new role.

**Purpose:** Automates access re-assignment when employees change roles, reducing manual intervention.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**3. Leaver Flow (User Deactivated → Remove Access/Notify)**

**Screenshot:** (User Deactivated → Read User → Construct → API Post → Notify Admin)

**Explanation:**  
This flow represents the **Leaver process**. When a **user is deactivated in Okta**, the flow is triggered. The user’s profile details (username, email, title) are read and structured into an object. The information is then sent to downstream systems (via API call, email, or Slack) for auditing or ticket creation. This ensures the user’s access is revoked across applications and groups.

**Purpose:** Automates offboarding by removing access immediately and alerting administrators.

A screenshot of a computer

AI-generated content may be incorrect.

**3.2 Access Policies (SSO & MFA)**

**Objective:** Secure application access using **Single Sign-On (SSO)** and **Multi-Factor Authentication (MFA)**.

**Steps Implemented:**

* Configured **Okta as IdP** with SAML and OIDC integrations.
* Enforced **MFA for all users** and applied conditional access policies.
* Tested **IdP-initiated and SP-initiated login flows**.
* Verified session handling and logout flows.

**Outcome:**

* Centralized authentication for enterprise apps.
* Enhanced user experience with seamless login.
* Strong MFA policies reduced account compromise risks.

**3.3 SCIM Provisioning (Okta → External SaaS App) ← *NEW / completed***

**Objective:** Automate account provisioning and lifecycle management for external SaaS apps (example: Zoom, Slack, GitHub) using SCIM.

**Steps Implemented**

1. **Enable SCIM provisioning** in the SaaS app integration in Okta (Admin Console → Applications → [App] → Provisioning → Enable SCIM / API Integration).
2. **Configure API token** from the target app inside Okta (enter target app's SCIM Base URL + API token in the App → Provisioning → API Integration).
3. **Map attributes** from Okta → App (Profile Mappings): at minimum map userName/email, firstName, lastName, title, department and any app-specific attributes.
4. **Test end-to-end:** create a test user in Okta, assign the SCIM-enabled app, and verify the account is auto-created in the target app. Confirm provisioning logs show success.
5. **Deprovisioning test:** deactivate/unassign the Okta user and verify account is deactivated or removed in the target app per policy.

**Outcome**

* Automated account creation and deactivation in target SaaS apps.
* Reduced manual account errors and faster onboarding/offboarding.
* Clear provisioning audit trail via Okta logs.

**Set Up SCIM Provisioning with Okta**

**SCIM Provisioning (Okta → External SaaS App)**

* **Scenario:** Automate provisioning into a SaaS app (e.g. Zoom, Slack, GitHub).
* **Steps:**
  1. Enable SCIM provisioning in app integration.
  2. Configure **API token** from target app in Okta.
  3. Map Okta attributes → App attributes.
  4. Test with a new user → auto-created in app via SCIM.
* **Screenshots Needed:**  
  ✅ App integration with SCIM enabled  
  ✅ Attribute mapping screen  
  ✅ Provisioning logs (successfully created user)

1. **Enable SCIM in Contentstack**

To allow provisioning of users in Contentstack’s organization through Okta, you need to enable SCIM in Contentstack by performing the following steps:

* 1. Log in to your Contentstack account and go to the [Organization Settings](https://www.contentstack.com/docs/owners-and-admins/organization-settings-overview) page.

2.Go to the **SCIM** tab and select the **Enable SCIM** option.  
A screenshot of a computer

AI-generated content may be incorrect.

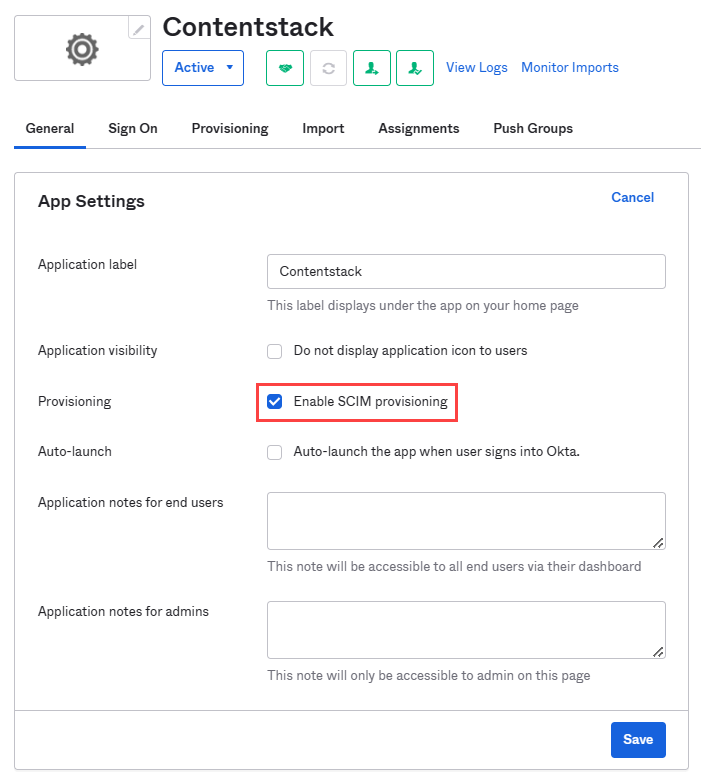
2.On the resulting **Enable SCIM**modal, click **Enable**.  
A close-up of a message

AI-generated content may be incorrect.

1. **Configure Provisioning in Okta**

link

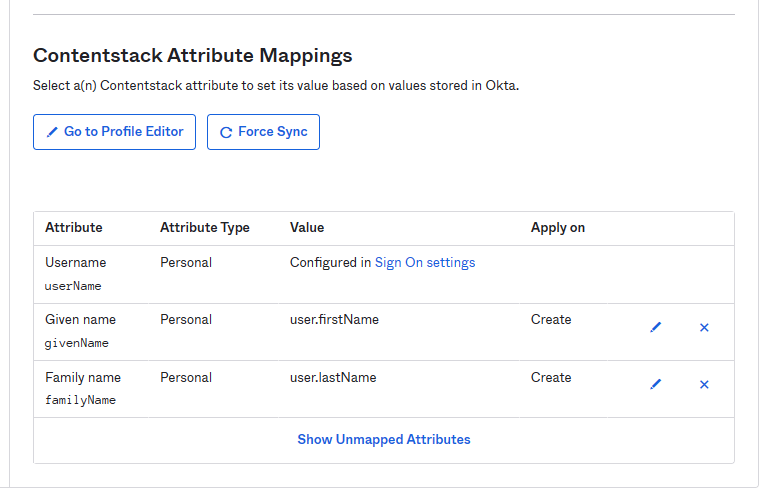
To enable your app to use the provisioning feature, before adding or removing a user from the Contentstack organization, you need to perform the following steps:

* 1. Navigate to the **General** tab and click **Edit**.  
     Within your Contentstack app in Okta, check the **Enable SCIM provisioning** checkbox and click **Save**.  
     
  2. Go to the **Provisioning** tab, and click **Edit**. Provide the following credentials in the **SCIM Connection window**:
     + **SCIM connector base URL**: Contentstack’s SCIM URL is used as **SCIM connector base URL**. Enter the **SCIM URL** generated in [**step 2.4**](https://www.contentstack.com/docs/developers/scim/set-up-scim-provisioning-with-okta#install-the-okta-generic-scim-app-from-contentstack-marketplace) while installing the **Okta Generic SCIM** app.
     + **Unique identifier field for users**: Enter a unique username.
     + **Supported provisioning actions**: Under this section, enable **Push New Users**, **Push Profile Updates**, and **Push Groups**.
     + **Authentication mode**: Select **HTTP Header** from the drop down.
* **HTTP Header**: Add the **Secret Token** generated in [**step 2.4**](https://www.contentstack.com/docs/developers/scim/set-up-scim-provisioning-with-okta#install-the-okta-generic-scim-app-from-contentstack-marketplace) as the **Bearer** token for the **Authorization** field.  
  A screenshot of a computer

  AI-generated content may be incorrect.
  1. Click **Test Connector Configuration** (see above screenshot) to ensure the connection between the Okta and the Contentstack app is successful.

Click **Save** to save the app provisioning configurations.

* 1. Navigate to the **Settings > To App > Contentstack Attribute Mappings** section to map user attributes such as userName, givenName,and

familyName.

* 1. Navigate back to the **Settings > To App** section and click **Edit**.

6.Enable **Create Users** for provisioning, and **Deactivate Users** for deprovisioning.  
A screenshot of a computer

AI-generated content may be incorrect.

* 1. Click **Save** to save the provisioning settings.

**Assign Users and Groups to Your Application**

* 1. link
  2. After configuring the provisioning settings, you need to assign either users or  groups (of users) to your app. Let’s see how to do them both.
  3. **Assign People to Your Application**
  4. link
  5. To assign people to your application, perform the following steps:
  6. Navigate to the **Assignments** tab. Click the **Assign** dropdown and select the **Assign to People** option.  
     A screenshot of a computer

     AI-generated content may be incorrect.You need to provide the individual's email address and click **Assign**.  
     A screenshot of a computer

     AI-generated content may be incorrect.
  7. In the resulting people assignment modal, click **Save** **and** **Go Back**.
  8. Click **Done** to save the assignment. The people assignments are listed as shown below:  
     A screenshot of a computer

     AI-generated content may be incorrect.

**✅ *“Provisioning Log – User successfully created in external SaaS app via SCIM”***

A screenshot of a computer

AI-generated content may be incorrect.

**3.4 Active Directory (AD) Integration**

**Objective:** Sync on-premises AD with Okta for identity federation and policy enforcement.

**Steps Implemented:**

* Installed Okta AD Agent and connected to AD domain.
* Imported users and groups from AD Organizational Units.
* Configured **attribute mappings** (sAMAccountName → username, givenName → firstName, etc.).
* Applied **Group Rules** to enforce security policies based on AD group membership.

**Outcome:**

* Seamless synchronization of AD identities into Okta.
* Consistent identity attributes across AD and Okta.
* Automated policy enforcement based on AD groups.

**Active Directory (AD) Integration with Okta**

**Scenario:**  
Sync on-premises AD with Okta to centralize authentication, automate user lifecycle management, and enforce MFA/SSO policies.

**Step 1 – Directory Integrations → AD Agent Setup**

**Screenshot:** (Add Active Directory screen, AD Agent configured status)

**Explanation:**  
Okta connects to on-premises Active Directory using the **Okta AD Agent**. The agent is installed on a domain-joined server and securely connects AD with Okta. Once the AD Agent is configured, Okta can import and manage AD users and groups. The status indicator shows whether the AD Agent is active and connected.

**Purpose:** Establishes the bridge between AD and Okta.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Step 2 – Import Users/Groups**

**Screenshot:** (Select Organizational Units → Import confirmation screen)

**Explanation:**  
Organizational Units (OUs) are selected to define which **users and groups** should sync from AD to Okta. Administrators can choose one-way sync (AD → Okta) or bi-directional sync (AD ↔ Okta). Once imported, administrators confirm whether new users should be added and stale users deactivated.

**Purpose:** Ensures Okta has an up-to-date copy of AD identities and groups.

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Step 3 – Attribute Mapping (AD → Okta Schema)**

**Screenshot:** (AD attributes list and Imported Attributes panel)

**Explanation:**  
Attributes from AD are mapped into Okta’s Universal Directory schema. Common mappings include:

* sAMAccountName → username
* givenName → firstName
* sn → lastName
* mail → email

Custom attributes (e.g., department, countryCode) can also be mapped to enrich the Okta user profile.

**Purpose:** Keeps identity data consistent across AD and Okta for authentication, provisioning, and SSO.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Step 4 – Group Rules (Assign Policies to AD Groups)**

**Screenshots:** (Group list, Rule editor for IT Admins/Developers)

**Explanation:**  
Group Rules automatically place imported AD users into the correct Okta groups based on attributes or AD group membership. For example:

* If title = IT Admin → Assign to **IT Admins group**.
* If countryCode = US/CA/NZ and title = Web Developer → Assign to **NA Developers group**.

These Okta groups can then be tied to MFA policies, SSO access, and app provisioning.

**Purpose:** Ensures consistent security and access policies for AD users in Okta.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**screenshots:**

* Show the **IT Admins group** with users assigned.
* Clearly highlight that some users were added **by rule** (not manually).

A screenshot of a computer

AI-generated content may be incorrect.

**screenshot:**

* Shows the **Group Rules list** (e.g., “Rule to manage IT Admins,” “Salesforce group rule”), all active.

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

**screenshots:**

* Show the **Edit Rule** configuration:
  + Basic rule: *If user attribute title = IT Admin → Assign to IT Admins group*.
  + Advanced rule (Expression Language): *If countryCode = US/CA/NZ AND title = Web developer OR manager = Sam/Max → Assign to NA Developers group*.

**3.5 Okta API Automation**

**Objective:** Enable automation for provisioning, reporting, and integration with external systems using Okta APIs.

**Steps Implemented:**

* Generated and managed **Okta API tokens**.
* Integrated with external apps via REST API for user provisioning.
* Automated group assignments and reporting tasks.

**Outcome:**

* Improved operational efficiency with API-driven automation.
* Enabled faster onboarding of applications into IAM.
* Supported custom workflows beyond UI capabilities.

1. **Project Timeline**

**Project Timeline (Report 2: Lifecycle, SCIM, AD, API Automation)**

* **Sep 2023 – Dec 2023**
  + Implemented **Lifecycle Management (Joiner–Mover–Leaver)** workflows.
  + Automated provisioning via Okta Workflows.
* **Jan 2024 – Jun 2024**
  + Configured **Access Policies** tied to lifecycle flows.
  + Enabled contextual authentication for movers/leavers.
* **Jul 2024 – Dec 2024**
  + Integrated **SCIM provisioning** with external SaaS apps (e.g., Slack, Zoom).
  + Validated auto-provisioning/de-provisioning logs.
* **Jan 2025 – Apr 2025**
  + Setup **Active Directory integration** (Okta AD Agent).
  + Imported users/groups, configured attribute mappings.
* **May 2025 – Aug 2025**
  + Implemented **Okta API Automation** (token-based workflows, bulk updates).
  + Finalized governance + reporting.

**5. Conclusion**

This IAM project successfully delivered a secure and scalable identity management solution with Okta at the core. By automating lifecycle processes, integrating with AD, enforcing access policies, and enabling API-driven automation, the project achieved:

* **Stronger security** with MFA and centralized authentication.
* **Operational efficiency** with reduced manual provisioning.
* **Compliance readiness** with clear lifecycle and access controls.

The solution now serves as a robust foundation for managing enterprise identity and access in a hybrid IT environment.