

OIG Beta User-Entitlement (Group) Load

This mechanism has been built to import an external apps users and entitlements into Okta for Access Certification.

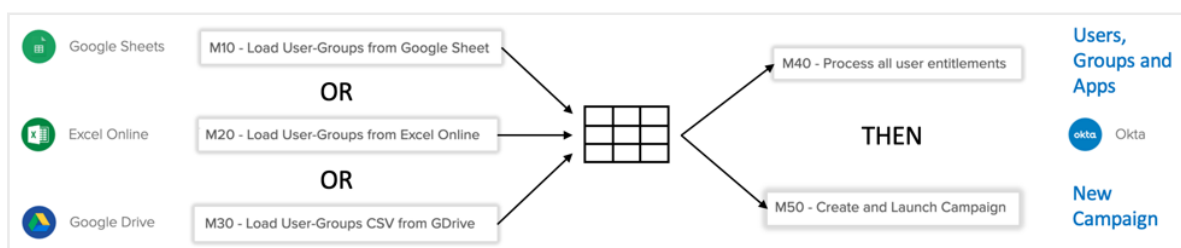
It is designed for test data and may not cope with tens of thousands of user-entitlement entries. It has only been tested on a small volume. It stores user-entitlement entries as rows in a Workflows table and a single list of objects in a main flow.

To use this mechanism there is some pre-work, then the data needs to be imported, and finally a flow is run to use the data to create the appropriate objects and relationships in Okta. For the import there are three options to load the data: 1. Google Sheets, 2. Excel Online, 3. Drive?

NOTE - there is another approach using a CSV directory documented here: <https://iamse.blog/2022/08/18/certifying-access-for-disconnected-application-in-okta/>. That approach achieves the same outcome of a cert campaign, but involves deployment of the On-Prem Provisioning (OPP) agent. It does not present the app entitlements as groups so the entitlement visibility is limited.

Overview

Okta Identity Governance (OIG) does not currently support importing external apps and their access entitlements into Okta. This integration, implemented in Okta workflows, provides three examples of importing a CSV file into an Okta workflows table, taking that data and creating groups to represent the entitlements and associating them with the app, and then creating and launching a campaign.



The end result is an Access Certification campaign, with users assigned to the external app, and the resources to review being the "groups" which are shown as an app-entitlement label.

Access certification

Test_Load Entitlements review

Due date: 11/18/2022 (in 29 days) Created by: Okta Admin (okta.admin@davidaedwards.com)

Description: Review all Entitlements associated with the Test_Load application

Pending reviews: 4, Approved: 0, Revoked: 0, Reassigned: 0, Progress: 0%

Pending reviews

Approve or revoke access to resources for the users. You can also reassign a review to another user. [View best practices](#)

Resource: All

User	Email	Resource	Actions
<input type="checkbox"/>	Yosemite Sam	yosemite@adjoined.local	Test_Load-User_ReadWrite
<input type="checkbox"/>	Daffy Duck	daffy@adjoined.local	Test_Load-User_ReadOnly
<input type="checkbox"/>	Yosemite Sam	yosemite@adjoined.local	Test_Load-User_ReadOnly
<input type="checkbox"/>	Elmer Fudd	elmer@adjoined.local	Test_Load-All_Admins

Review details

User details

User: Elmer Fudd

Email: elmer@adjoined.local

User status: Active

Title: Not defined

Cost center: Not defined

Organization: Not defined

Department: Not defined

Manager: Kent Brockman

Resource details

Group: Test_Load-All_Admins

History

10/19/2022

Okta Admin (okta.admin@davidaedwards.com) assigned this review to you

The remainder of this document will walk through the configuration of this.

Pre-Work

Create Okta App for External App

Before loading the data you will need to create an app that matches the app name you will use in the CSV file. The assumption is that one execution of processing the CSV file will be for a single app (if you have multiple apps you need to run this once for each app and its data). The mechanism is not built to handle multiple external apps at once.

Create the User-Groups CSV File

You will also need to create a CSV file. The Workflows and Workflow table are based on the following columns: firstName, lastName, email, managerEmail, appName, groupName (the last field, groupName, is the app entitlement to be used and defined as a group). The managerEmail field is not currently used (it is expected that the users in Okta have a managerId defined).

Create Workflows Connections

There are different workflows connectors needed for each of the sets of flows.

- The first set of flows to import the CSV file have three options and you only need to use one of them
 - The M1* flows use a Google Sheets connection (I called mine "Test User-Groups")

- The M2* flows use an Excel Online connection (I called mine "*Test User-Groups*")
- The M3* flows use a Google Drive connection (I called mine "*Okta.Admin at deadwoods.com*")
- The M4* and M5* flows use an Okta connection (I called mine "*myOkta*")

If you set up different connectors you will need to go through the supplied flows and select your connectors.

Import the Workflows and Tables

Create a new folder and import the .folder file that was shared with this doc. It will create all of the flows and three tables.

If you created connections (as per above) and named them differently, you will need to go through and select the appropriate connections in each connection card.

Setup the Environment Variables Table

There is a table, called Environment Variables, that stores variables used across the flows.



rowid (auto)	updated (auto)	varName	varValue
bbfefec0-4f42-11ed-a783-c	10/19/22 12:12am UTC	oktaAdminId	00i...j5d5
71aa1420-4ecc-11ed-8635-	10/18/22 10:05am UTC	oktaDomain	deadwoods.okta.com
613324b0-4ecc-11ed-a89a-	10/19/22 12:38am UTC	apiToken	00_8...l-k4Rin

There are two columns, varName (variable name) and varValue (variable value). There variables are required for this set of flows:

- **oktaAdminId** - this is the id for a user who will be the fallback reviewer in the campaign. It could be any user.
- **oktaDomain** - this is the domain name (minus the preceding https://) that is used to build API URLs
- **apiToken** - this is an API token for API access into Okta (you will need to create one, preferably as a Super Admin)

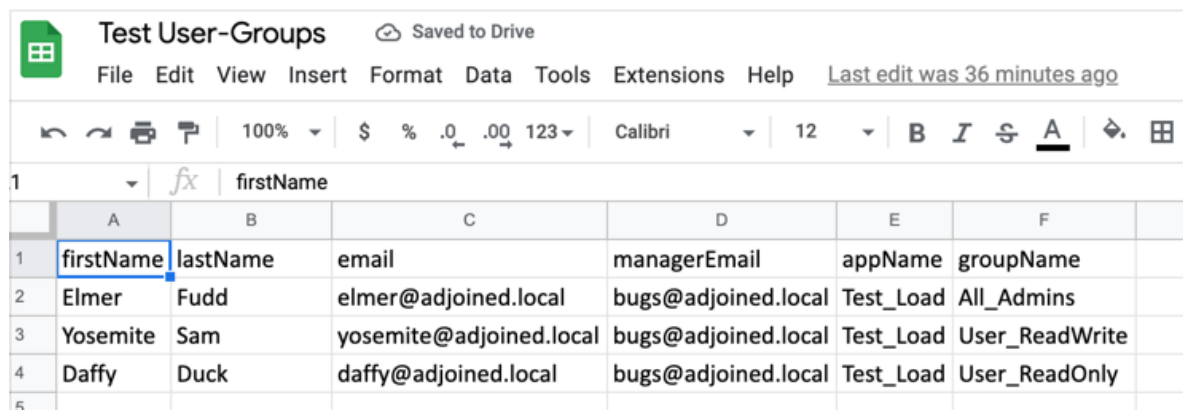
With this configured, you are ready to run one of the import flows.

Step 1 - Run One of the Import flows

Use one of the following depending on where/how you want to store the CSV info.

M1* - Import from Google Sheets

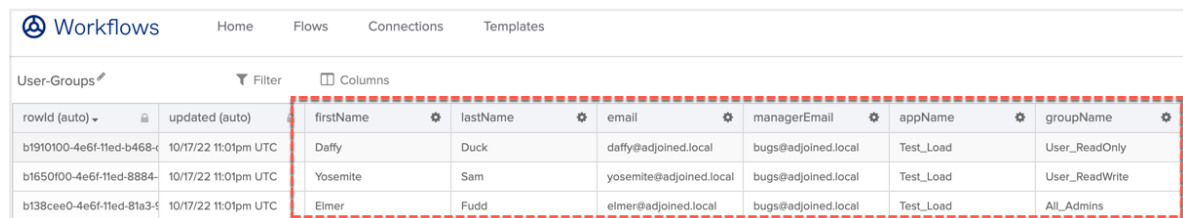
1. Need a Google Sheets connection to a users Google account
2. Create a Google Sheets file with the contents (example below) in that users account



The screenshot shows a Google Sheet titled "Test User-Groups" with a menu bar (File, Edit, View, Insert, Format, Data, Tools, Extensions, Help) and a toolbar. The sheet contains a table with the following data:

	A	B	C	D	E	F
1	firstName	lastName	email	managerEmail	appName	groupName
2	Elmer	Fudd	elmer@adjoined.local	bugs@adjoined.local	Test_Load	All_Admins
3	Yosemite	Sam	yosemite@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadWrite
4	Daffy	Duck	daffy@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadOnly
5						

3. Edit the M10 flow to select the correct connection and select the specific file
4. Make sure the column names match the ones above - if not you will need to update the User-Groups table in workflows and the various table cards (they are coded to those specific fields)
5. Run M10 to populate the Workflows table



The screenshot shows a "Workflows" table with columns: rowId (auto), updated (auto), firstName, lastName, email, managerEmail, appName, and groupName. The data is filtered to show three rows, which are highlighted with a red dashed border:

rowId (auto)	updated (auto)	firstName	lastName	email	managerEmail	appName	groupName
b1910100-4e6f-11ed-b468-	10/17/22 11:01pm UTC	Daffy	Duck	daffy@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadOnly
b1650f00-4e6f-11ed-8884-	10/17/22 11:01pm UTC	Yosemite	Sam	yosemite@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadWrite
b138cee0-4e6f-11ed-81a3-	10/17/22 11:01pm UTC	Elmer	Fudd	elmer@adjoined.local	bugs@adjoined.local	Test_Load	All_Admins

M2* - Import from Excel Online

Similar to above...

1. Need an Excel Online connection to a users O365 account
2. Create an Excel Online file with the required contents (example below) in that users account

	A	B	C	D	E	F
1	firstName	lastName	email	managerEmail	appName	groupName
2	Elmer	Fudd	elmer@adjoined.local	bugs@adjoined.local	Test_Load	All_Admins
3	Yosemite	Sam	yosemite@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadWrite
4	Daffy	Duck	daffy@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadOnly

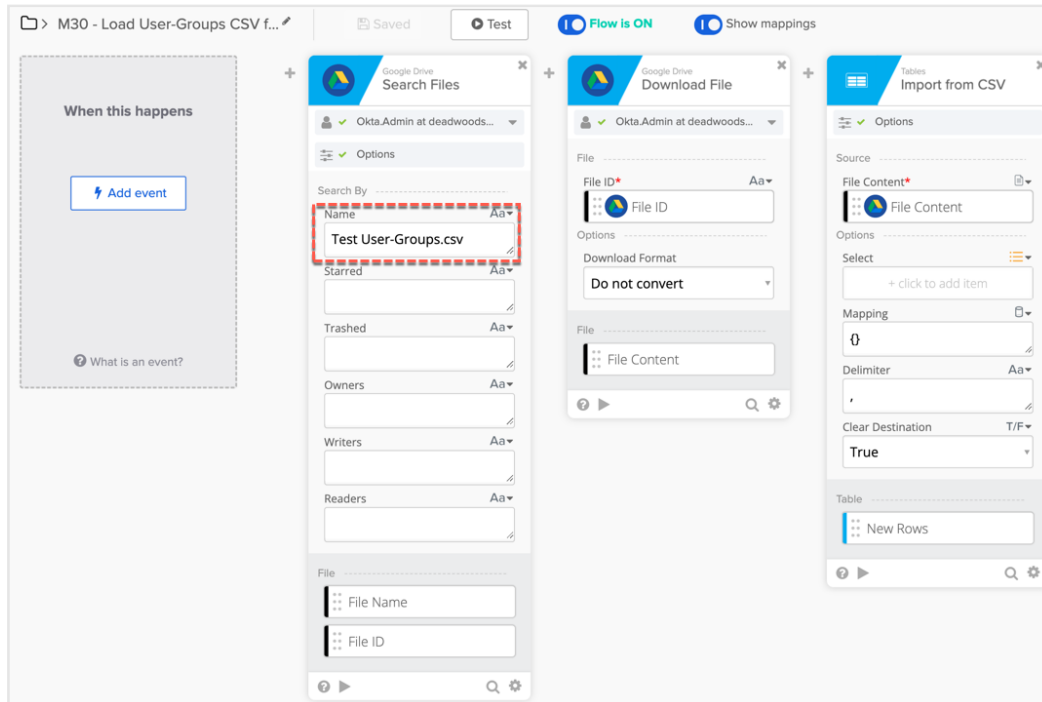
3. Edit the M20 flow to select the correct connection and select the specific file
4. Make sure the column names match the ones above - if not you will need to update the User-Groups table in workflows and the various table cards (they are coded to those specific fields)
5. Run M20 to populate the table

rowId (auto)	updated (auto)	firstName	lastName	email	managerEmail	appName	groupName
b1910100-4e6f-11ed-b468-c	10/17/22 11:01pm UTC	Daffy	Duck	daffy@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadOnly
b1650f00-4e6f-11ed-8884-c	10/17/22 11:01pm UTC	Yosemite	Sam	yosemite@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadWrite
b138cee0-4e6f-11ed-81a3-c	10/17/22 11:01pm UTC	Elmer	Fudd	elmer@adjoined.local	bugs@adjoined.local	Test_Load	All_Admins

M3* - Import CSV from Google Drive

This approach involves storing the CSV file on a Google Drive (the approach should also work for other drives like Box and OneDrive, but I haven't tested it).

1. Need a Google Drive connection to a users Google account
2. Make sure the CSV file has the same column names as the columns in the Workflows table. If not you will need to edit the file or the table in Workflows
3. Copy the CSV file into the drive
4. Edit the M30 flow and select the correct connection and also in the Search Files card enter the exact filename for the CSV file.



5. Run the M30 flow

Workflows								
User-Groups								
rowId (auto)	updated (auto)	firstName	lastName	email	managerEmail	appName	groupName	
b1910100-4e6f-11ed-b468-	10/17/22 11:01pm UTC	Dafy	Duck	dafy@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadOnly	
b1650f00-4e6f-11ed-8884-	10/17/22 11:01pm UTC	Yosemite	Sam	yosemite@adjoined.local	bugs@adjoined.local	Test_Load	User_ReadWrite	
b138cee0-4e6f-11ed-81a3-	10/17/22 11:01pm UTC	Elmer	Fudd	elmer@adjoined.local	bugs@adjoined.local	Test_Load	All_Admins	

That's it to load the users, groups, and apps into the Workflows table. We will next use another flow to load all of this into Okta.

Step 2 - Use Loaded Data to Update Okta

Now that one of the methods above has been used to load all the user-entitlement data into the Workflows table, we run another flow to check the data and apply any updates.

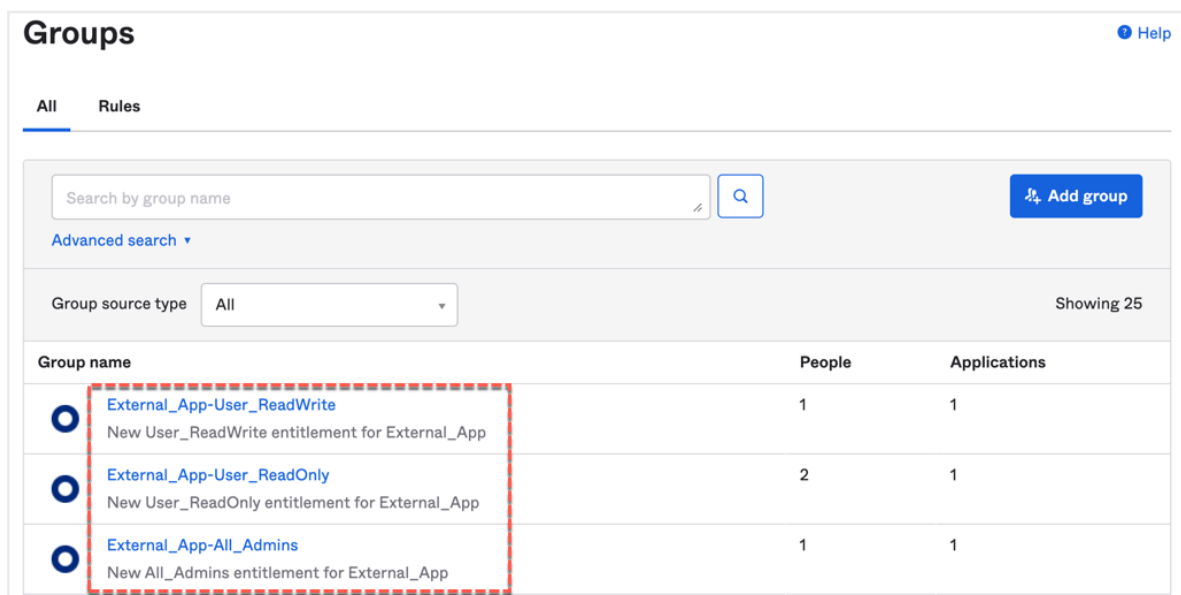
The M40* flow contains the following steps:

1. Read all the table rows into a common object that will be used for all the following steps
2. Check all users are valid Okta users by email address using the M41 flow (if any are not valid, the flow will error out)
3. Check the app is valid by name, using the M42 flow (if there's no app by the name in the CSV, the flow will error out) and

return the app ID

4. Strip out a unique group (entitlement) list from the CSV data using the M43 flow
5. For each unique group (entitlement) create an Okta group (name is <app>-<group>), add the relevant users and assign that group to the app using the M44 flow

Run this flow. You will see in Okta there are new groups created, that they have the users assigned as per the CSV and the group is assigned to the app.



Group name	People	Applications
External_App-User_ReadWrite New User_ReadWrite entitlement for External_App	1	1
External_App-User_ReadOnly New User_ReadOnly entitlement for External_App	2	1
External_App-All_Admins New All_Admins entitlement for External_App	1	1

With these Okta objects and relationships we can now create and launch a campaign.

Step 3 - Create and Launch a Campaign

The last step (set of flows) is to use the new data to create and launch a campaign, where the resources are the entitlements (i.e. groups called <app>-<group>).

The M50* flow contains the following steps

1. Setup some common variables for the two sub flows
2. Create the campaign using the M51 flow
3. Launch the new campaign using the M52 flow

You don't need to change any campaign parameters. It will set the date as the next day (D+1), run for 30 days, use the user.profile.managerId field for reviewer and it uses a single Okta user

(Id stored in the Environment Variables table) for the fallback reviewer. If you want to alter any of these settings, you will need to go into the M51 flow and modify there.

Run the M50 flow and you should see a new campaign with the resources being all the groups for the app as per the CSV.

Access certification campaigns

[+ Create campaign](#)

Active [Scheduled](#) [Closed](#)

Campaign	Start date	End date	Certification Progress
External_App Entitlements	10/19/2022	11/18/2022	0% <div></div>

[<](#) [>](#)

External_App Entitlements

● Active [Actions](#)

Created: 10/19/2022 Start date: 10/19/2022 Start time: 5:10 PM GMT+11 End date: 11/18/2022 Duration: 30 Days

Description: Review all Entitlements associated with the External_App application

Overview

Resources

Groups: External_App-All_Admins, External_App-User_ReadOnly, External_App-User_ReadWrite

[View all](#)

Reviewers

Reviewers defined using EL

Users

All Users

Remediation

Approved: Don't take any action

Revoked: Don't take any action

No response: Don't take any action

Progress

0%

Total reviews
4

Pending
4

Approved
0

Revoked
0

Pending Closed

Pending Reviews

Reviews that reviewers have not yet taken an action on. Pending reviews can be reassigned to another reviewer.

Resources

All

Reassign (0)

<input type="checkbox"/>	User	Resource	Reviewer	Action
<input type="checkbox"/>	Elmer Fudd	External_App-All_Admins	Kent Brockman	<div>Reassign</div>
<input type="checkbox"/>	Daffy Duck	External_App-User_ReadOnly	Kent Brockman	<div>Reassign</div>
<input type="checkbox"/>	Yosemite Sam	External_App-User_ReadOnly	Kent Brockman	<div>Reassign</div>
<input type="checkbox"/>	Yosemite Sam	External_App-User_ReadWrite	Kent Brockman	<div>Reassign</div>

←

→

From here it is standard access certification execution. There is no removal of access on a revoke. You could build additional flows that would respond to the review-revoke event and go update the CSV in whichever location (Excel Online, Google Sheet, Google Drive) but that has not been done here.