SPOOFING WITH THE TASKS PLUGIN

The spoofing capability allows a tester to modify data associated with a user, in the form of a CSV file, and then upload that data to SailPoint AS IF the data came from the authoritative source, no matter which.

User Guide

The spoofing function is accomplished with two TaskDefinition templates: Plugin Account Extract and Plugin Aggregate Extracted Data

A TaskDefinition template is accessed using the New Task dropdown, on the Tasks page, by scrolling down until the desired task type is found. The Plugin Account Extract allows extracting account data, and Plugin Aggregate Extracted Data allows aggregating that data into a connector.

Account Extract

Selecting Plugin Account Extract creates a new Account Extract task definition.

Per any other new task creation process, the user will do the following:

- Supply a descriptive name. Best practice is to set the Previous Result Action to Rename.
- Next, select the Application you wish to extract data from. Once that is done select either one or more identities that you know have an account on that application, or just select all accounts and all identities which have accounts on that application will be selected.
- Specify an existing folder path for the output file. After that, specify the filename or leave to be the default. In the folder name the following substitutions can be used: $date$ will be substituted with the date in YYYYMMDD format, and $datetime$ will be substituted with the date and time in format YYYYMMDD-HHMMSS. You can also use $applicationName$ and the application name will be substituted.
- Specify a filename. In the file name section you can use $applicationName$ $identityName$ or leave blank to use the default. You can leave the CSV file extension off, and it will be added, but if a file extension is provided, it will be preserved.

This task is especially helpful when you would like to perform a daily extract of application data from a connector, to be able to see the history of data in the system. It currently cannot access the identity data, only account data.

Aggregate Extracted Data

Selecting Plugin Aggregate Extracted Data creates a new Aggregate Extracted Data task definition. Again per any other new task creation process, the user will do the following:

- Supply a descriptive name. Best practice is to set the Previous Result Action to Rename.
- Next, select the Application you wish to aggregate the extracted data into.
- Next, specify an existing folder path for the output file.
- After that, specify the filename to be read from. The file must be in CSV format, and again the CSV file extension is not required.
- If desired, the Disable optimization of unchanged accounts checkbox can be selected.

The task spoofs the file data into the selected connector, as if it were actually aggregated from the connector. This avoids the need to create a second set of identity attribute sources which would be needed when attempting to read data from a delimited file.

Notes:

The normal process I use for testing, is to aggregate one or more users from the authoritative source, and then perform an account extract of all of the accounts into a master file for analysis. This allows me to discover any unusual formatting of the data, especially date data. I often will write an Account Customization rule to reformat all dates to a common YYYY-MM-DD format.

I recommend limiting the use of Microsoft Excel only for viewing the files, not to edit them. This is because Excel reformats dates. This can be overcome, by editing the Account Customization rule to detect and modify any dates that are presented in Excel format. The same would be needed for boolean fields.