**Target OSTv3 – ServiceNow Intake Automation Script Version 2.x**

Last Modified: 3/27/2018

Table of Contents

[Process Outline & Prerequisites 3](#_Toc509839147)

[**1.** **OSTv3 New Tenant Fulfillment Process Overview** 3](#_Toc509839148)

[**2.** **Intake Automation Script** 3](#_Toc509839149)

[Running the Intake Automation Script 4](#_Toc509839150)

[**1.** **Preparing the Input Folder** 4](#_Toc509839151)

[**2.** **Running the Intake Automation Script** 6](#_Toc509839152)

[**3.** **Processing the Scripts Output** 8](#_Toc509839153)

[**4.** **Validation and Completion** 10](#_Toc509839154)

*The following lists important changes that have been made to this document:*

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Prepared by | Comments |
| 0.0 | 02.07.2018 | David Moreno | Initial Version |
| 1.1 | 03.26.2018 | David Moreno | Updated process for Release Version 2.x of the script |

# Process Outline & Prerequisites

**Key Terms**

|  |  |
| --- | --- |
| Terms, acronyms and Definitions | Description |
| OIM | Oracle Identity Manager |
| MyAccess | Target Corporation’s Identity and Access Management tool built on OIM platform; My Access is the user interface used to request, approve, and fulfill access requests |
| OpenStack (OST) | open-source software platform for cloud computing, mostly deployed as infrastructure-as-a-service. |
| Change Request (CR) | Service Now form that will serve as the intake ticketing system for new openstack tenants. |
| Bastion Host (BHO) Virtual Machine | Virtual machine running windows within the secure segment of target’s firewall |
| Requestable Role | A requestable role can be requested by an enduser in the MyAccess entitlement and role catalog. These roles do not require rules. |

### **OSTv3 New Tenant Fulfillment Process Overview**

Each time a new tenant space is created in OpenStack environment, a new CR will be created in Service Now.

The ticket will require the following items be created in order for the ticket to be considered ***closed and complete.***

The automation script will take in the CR input and automate the creation of 5 files that can be used to satisfy the

following requirements for any number of requests in a single execution.

### **Intake Automation Script**

The script will make it very efficient and time effective to map the CR inputs to flat feeds files used by yourself and other OIM Admins to enact changes on catalog items, create roles, and modify roles. The mapping is automatic which reduces time spent mapping CR description information to catalog items and reduces steps between intake and fulfillment.

# Running the Intake Automation Script

### **Preparing the Input Folder**

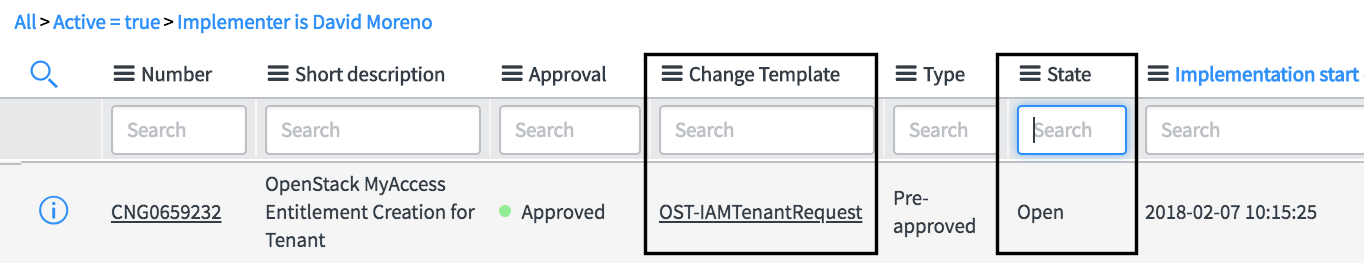
The automation script will require an Input Folder with some input files in order to run.

In order for the script to run correctly, it is very important to make sure that**:**

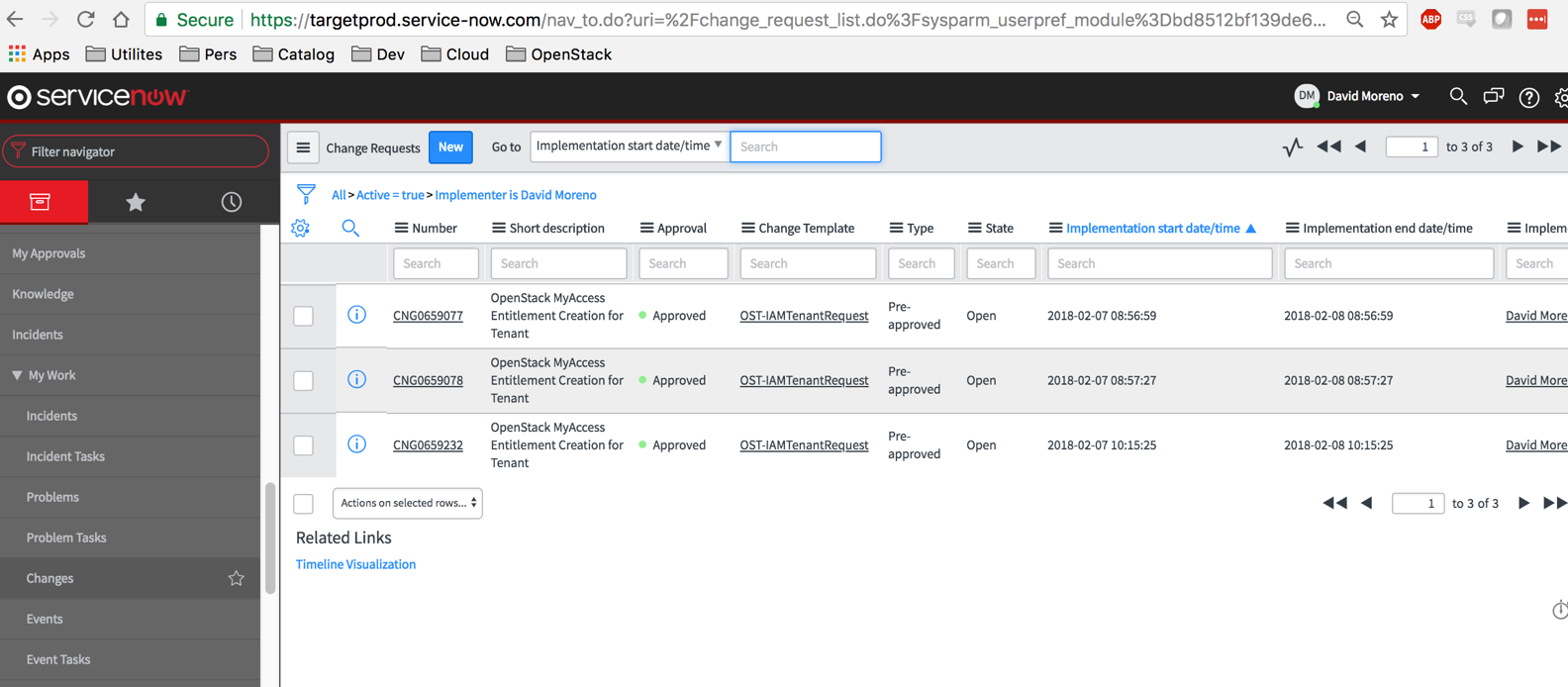
* The input folder be called **Input**
* Each of the CR descriptions be saved as **.txt** files, and placed *within* the input folder

*Note: in release version 2.x of this script, the Bi Publisher report is no longer needed*

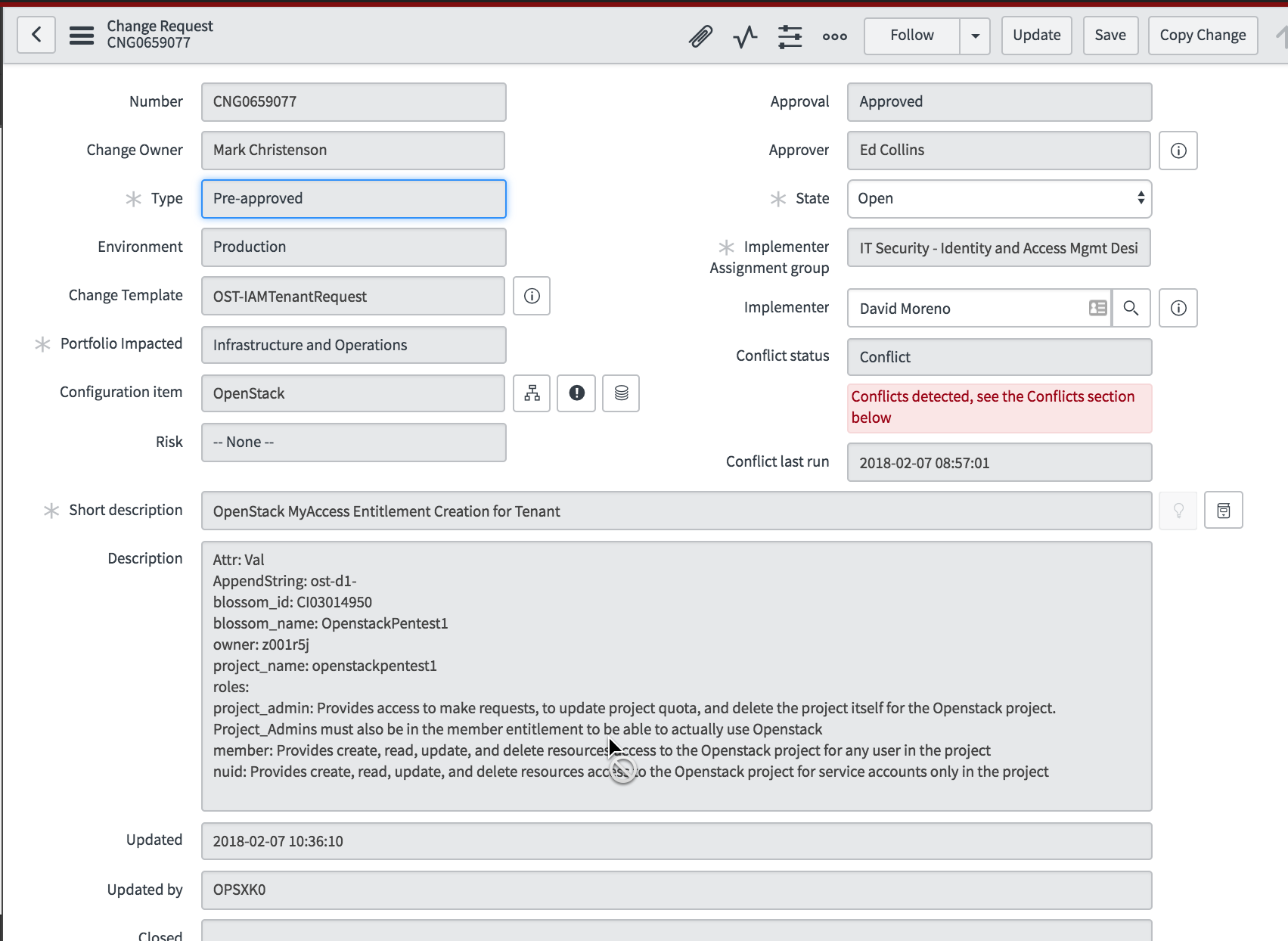
1. **Capture the Service Now CR Descriptions**
   1. Log into the ***targetprod.service-now.com*** page.
   2. Identify which CRs need to be worked on  
      **Useful Filters:**
      1. Set *“State”* to not be ***“Closed Complete”* or *“Closed Skipped”***
      2. Set the *“Change Template”* to be ***“OST-IAMTenantRequest”***



* 1. Assign that CR to yourself, to track which CRs are being currently and actively addressed by others on your team



* 1. Copy and Paste all of the text in the CR Description into a notepad or text editor.
     1. Make sure the formatting is preserved as the formatting is critical to the mapping of the script



* 1. Save the text file as a **.txt** file
     1. The file extension **MUST BE .txt**
     2. Cannot be extension **.doc**, **.docx**, etc
  2. Repeat Steps 1-5 for each CR that you aim to complete in this execution.
  3. Make sure these text files are saved within the input folder of the containerized intake automation script location (i.e. in the input folder that is located within the same folder as the script file)

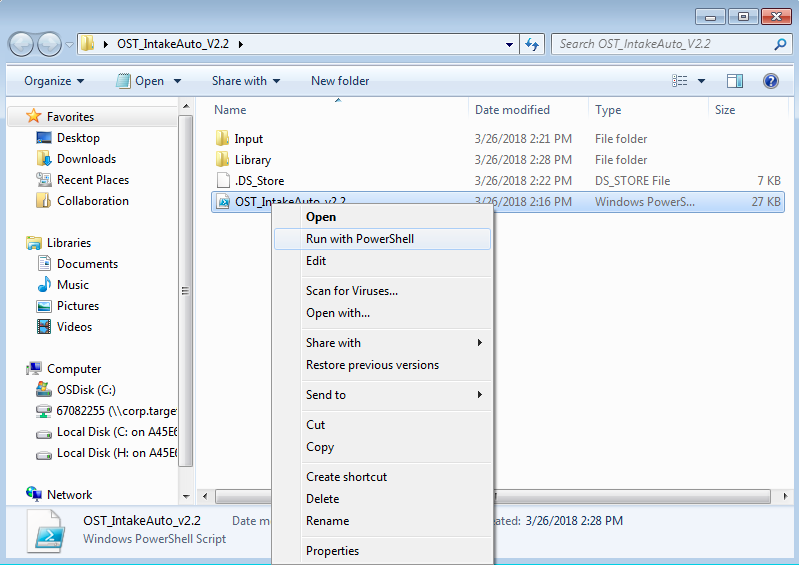
1. **Log into BHO VM pool** 
   1. Log into a Bastion Host (BHO) vm pool within the secure segment of Target’s corporate network.
      1. https://bvo.target.com/

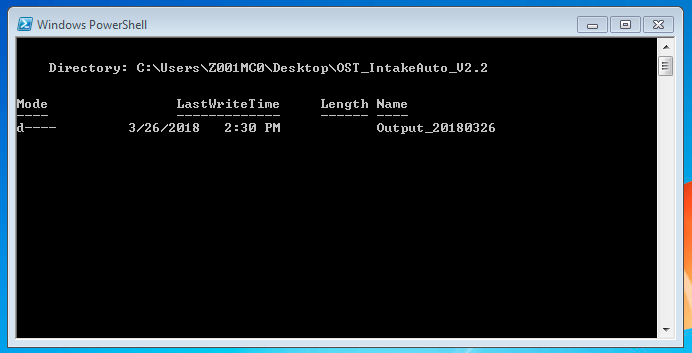
### **Running the Intake Automation Script**

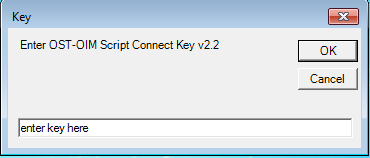
The automation script will require windows operating system, and PowerShell installed to run correctly.

**Technical Requirements:** There must be at least 6mb of free disk space on the Windows VM from which the script will be launched. The VM must be within the secure segment of the corporate network.

1. **Ensure the directory configurations are correct**
   1. If the Automation script is in a zip file then unzip it and extract it to your H:/ Drive
      1. NOTE: The script CAN Be run from your H:// Drive
   2. Make sure that the ***Input*** folder from the previous section is in the current directory along with the PowerShell script. This input folder should contain the CR text files from the previous section.
   3. Make sure the ***Library*** folder is located in this container
      1. If the Library folder is missing another copy of the .zip should be downloaded



1. **Run the script**
   1. *Right Click* on the ***.ps1*** file contained in the automation script folder
   2. Click on *Run with PowerShell*
      1. On the first time running an external PowerShell script on your local machine you might be prompted to ***Verify that you trust the source*** of the script
      2. If Prompted, type in **Y** to note that you trust the source of this script
   3. A PowerShell command window may pop up, signifying that the script is processing the files in the input folder.
   4. A prompt will surface that requires the Scripts Encryption Key to be entered.
      1. Enter the key (alphabetic string) into the text box
      2. Click OK
      3. NOTE: This is *not your user password* it is a key that was communicated to you when this script was delivered to you.
      4. If you require a copy of the key contact [***david.l.moreno@target.com***](mailto:david.l.moreno@target.com)

******

* 1. When the script is completed the Windows Powershell command window will disappear automatically, signifying the process is complete
  2. There will be an output folder in the same location as where the script ran
     1. Naming format is Output\_<**DateofRun**>

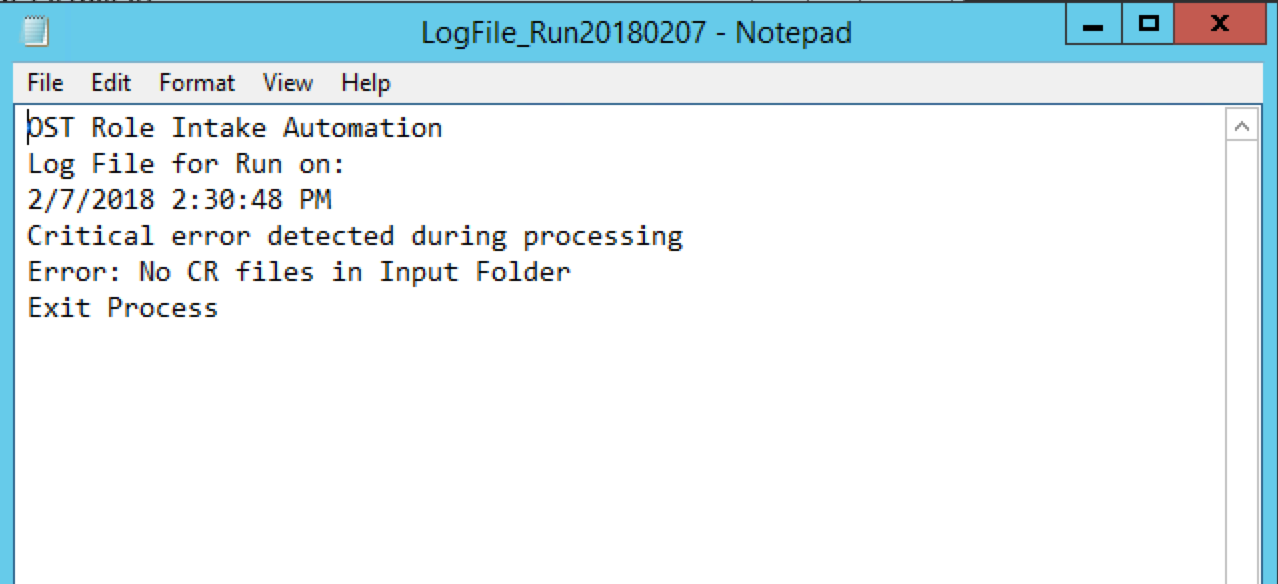
### ../../../../../../Desktop/Screen%20Shot%202018-02-07%20at%202.3**Processing the Scripts Output**

The automation script will produce an output folder with 5 csv files and a single text log file.

The log file will contain important information about when the job was run, which files were processed, and what the outcomes of each file process was.

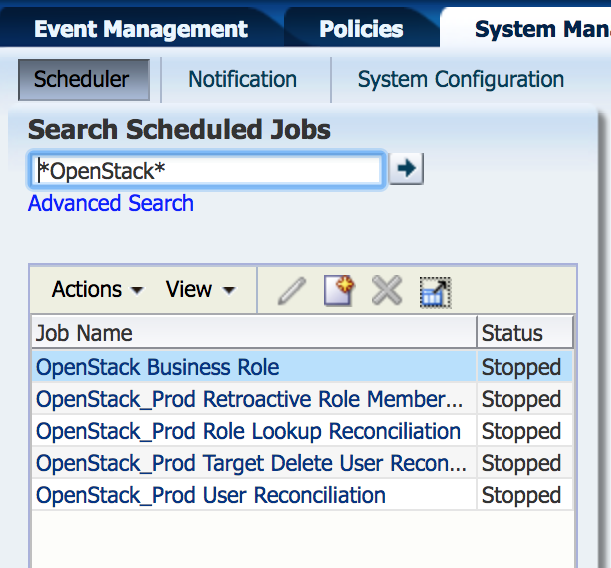
If errors are captured they will be reported in the log file. See Troubleshooting section for help on errors.

1. **Review the Log file for errors**
   1. The log file will capture:
      1. Date and time stamp of run
      2. Which files were processed
      3. Any errors during processing per file (per CR text file)
      4. Any critical errors due to inputs missing, etc
   2. Critical errors will be reported if there is an error captured that will not allow any files to be processed
      1. Example: Input CSV from Bi Publisher report is missing, there is no way for the script to map CR text file parameters to entitlement entity keys or entity Names
      2. Example: Input folder missing. The folder is not named “Input” or is missing entirely. There is nothing for the script to process
      3. Critical errors will prevent the script from running and it will report that a critical error was captured.

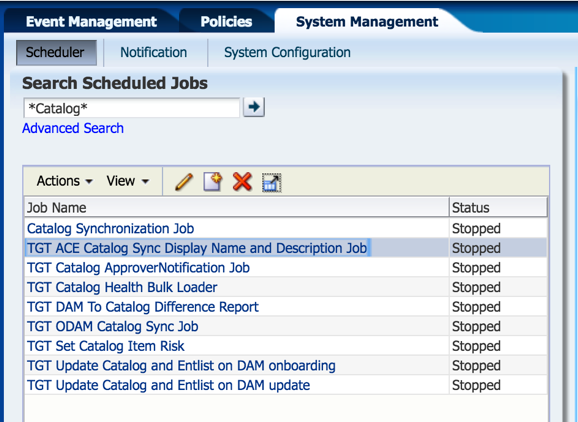


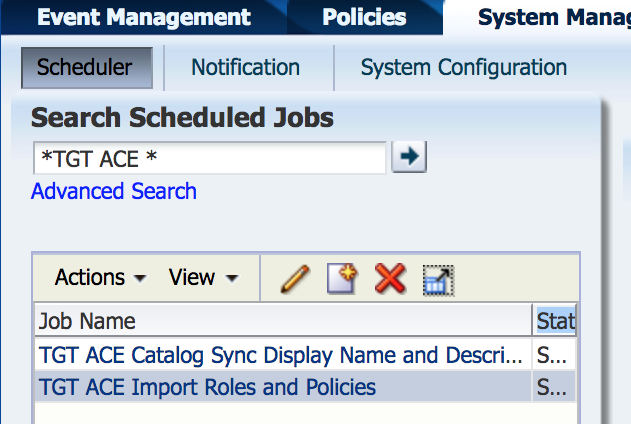
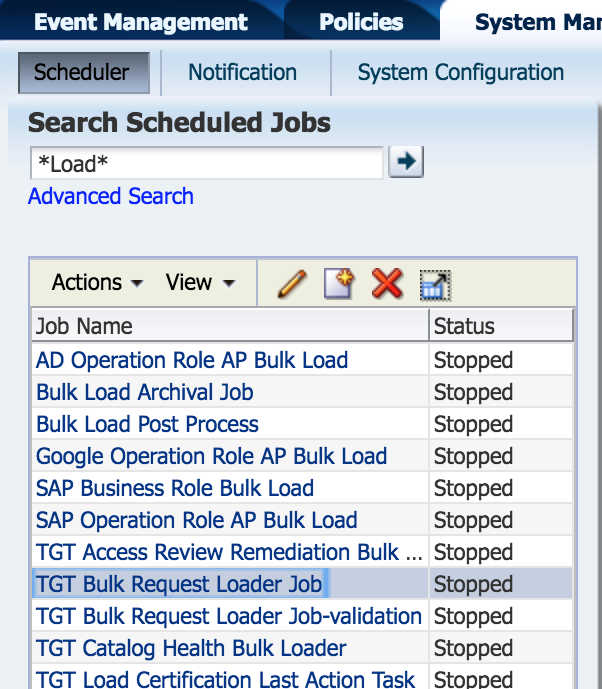
* 1. Noncritical errors will be reported if there is an error captured but the rest of the file can be processed
     1. Example: the current CR Text file expects there to be a match for entitlement in the Bi Pub report but there is no match.
     2. Trouble shoot by either running a fresh bi publisher report and using that fresh report for the script
     3. Ensure there was not a mistake in copy and pasting the CR text into the text file
     4. Ensure there are no typos in the CR text

1. **Process the files that are ready for the Scheduled Jobs (Assumes there are not errors reported in the log file)**

The automation script will output 5 CSV files. Of those, 4 are ready for schedule jobs without any additional work. *Must have access to MyAccess-Ops.Target.com/sysadmin to run scheduled tasks*

* 1. Run the job to bulk create the OST Roles and OST Access policies
     1. Job is called ***“OpenStack Business Role”***
     2. Refer to the runbook for SAP business role, AD business Role, or GCP Business Role jobs for more information on how to run this job
     3. Use the file called **xxxxxxxx-OSTroleCreation.csv** as the input file for this job



* 1. Run the job to update the catalog descriptions for the entitlements in the CR text files.
     1. Job is called ***“TGT ACE Catalog Sync Display Name and Description Job”***
     2. Use the file called **xxxxxxxx-Description.csv** as the input file for this job
  2. Run the job to create Approver Groups
     1. Job is called ***“TGT ACE Import Roles and Policies”***
     2. Use the file called **xxxxxxxx-ApprGroupCreation.csv** as the input file for this job
  3. Run the job to assign users to the approver groups
     1. Job is called ***“TGT Bulk Request Loader Job”***
     2. No approvals should be required, as we are just adding a member to the approver group
     3. Use the file called **xxxxxxxx-ApproGroupBulkAddUser.csv** as the input file for this job

1. **Update the catalog Sync file**

The automation script will output 5 CSV files. Of those, 4 are ready for schedule jobs without any additional work. The last one requires some additional manual work to get it ready for the job.

* 1. Make sure that all steps in Part B of processing the files is completed
  2. Open the file called **xxxxxxxx-CatalogSync.csv** file using Microsoft Excel or similar processor
     1. There will be fields in this file that do not have an appropriate value for the job that it is for
     2. These fields will be noted by the text string: **“NeedInput-”**
  3. **Update the Entity\_Key** for OST Roles
     1. Each field that needs to be updated will have, as an example:   
        ***“NeedInput-ConvertToKey-OST-d1-openstackpentest1-project\_ddmin”***
        1. “**NeedInput-”** meaning the cell needs to be updated
        2. “**ConvertToKey”** meaning that the field needs a key
        3. **“OST-d1-openstackpentest1-project\_admin”** meaning that in this case, the entity whose name is “OST-d1-openstackpentest1-project\_admin” needs to be mapped to an entity key, and the entity\_key needs to go into this cell
     2. Follow this example for the approver roles.
     3. Make sure all values in the column labeled “**Entity Key**” are keys, and match the rest of the cells of that row
  4. **Update the Approver Role**  to be a key
     1. Not all fields will need this populated, those fields that are blank can stay blank.
     2. Only update those fields that have the string **“NeedInput-”**
     3. As an example:“NeedInput-ConvertToKey-Approvers\_OpenStack\_openstackpentest1”
        1. “**NeedInput-”** meaning the cell needs to be updated
        2. “**ConvertToKey”** meaning that the field needs a key
        3. **“**Approvers\_OpenStack\_openstackpentest1**”** meaning that in this case, the entity whose name is “Approvers\_OpenStack\_openstackpentest1” needs to be mapped to a key so that they will be assigned as the approver group for that row
  5. **Update the Certifier User** to be a key
     1. All fields in this column will need to be updated
     2. As an example:  
        “NeedInput-ConvertToKey-z001r5j”
        1. **NeedInput-”** meaning the cell needs to be updated
        2. “**ConvertToKey”** meaning that the field needs a key
        3. **“**z001r5j**”** meaning that in this case, the user whose login is z001r5j needs to be converted to a user Key
  6. Save the update file
     1. Make sure there are no more fields in the entire document that start with the string **“NeedInput-”**
  7. Run the catalog sync job
     1. Use this updated, saved csv file as the input

### **Validation and Completion**

1. Validate the jobs ran in steps B and C from previous section ran correctly
   1. For more information regarding how to validate these jobs refer to the runbook for these jobs
2. Close out those CRs that you had assigned yourself and completed in this run