

# Hands-On Lab z/OSMF Security Configuration Assistant

## 

## Abstract:

The z/OS Management Facility (z/OSMF) provides a web-based graphical interface for system programmers on z/OS. This hand on lab will give an opportunity to learn about the functions and features in z/OSMF first hand. Attendees can use a centralized UI to manage and validate security requirements by product or function.

This session will be useful to systems programmers and security administrator who needs to work with security configuration for scenarios like security validation, security trouble shooting, etc.

## Introduction to z/OSMF Security Configuration Assistant task:

When configure a z/OSMF server or enable a z/OSMF service, security setup is usually involved. The administrator might need execute a set of commands or script to figure out what security requirements they lack. This is usually time-consuming and needs much communication efforts.

The Security Configuration Assistant (SCA) task provides a centralized visual framework for examining the security requirements of z/OSMF and other z/OS components. Specifically, SCA task lists the required resources and access requirements by z/OSMF services. You can also import security descriptor file (in human-readable JSON format) of other z/OS components, or import the security descriptor files you created by your own, into z/OSMF. Authorized administrator can validate and fix those security requirements automatically. This could mitigate the repeated communication between z/OS system programmer, who configures z/OSMF or other z/OS components, and z/OS security administrator. SCA task consists of tabbed sections and tabular reports that can be expanded or collapsed, as needed. This framework provides a comprehensive perspective on your z/OS security setup.

## Key features of the z/OSMF Security Configuration Assistant (SCA) task

With the SCA task, you can:

* Review security requirements by function.
* Automatically validate security requirements on a flexible scope, regardless of what your security product is.
* Fix security failure by reviewing the commands generated by SCA and submitting commands to your security product

With Import function of SCA task, you can

* Create a JSON file (a.k.a. Security Descriptor file) to organize and describe security requirement based on your need.
* Import Security Descriptor files into SCA and review their security requirements in SCA.
* Specify runtime values to variables in security resource profile.
* Perform security validation against user id or group id for a flexible scope.
* Review & Fix validation failures
* Review validation result in graphic chart
* Filter validation results so that you can quickly narrow down to security requirements with specific type of validation result.

**SCA Lab**

This lab consists of 10 tasks:

1. Log on to z/OSMF
2. Launch the Security Configuration Assistant task.
3. Check the result of Validation all
4. View the details of each tab
5. Check the statistics of validation
6. Filter out the failed validation
7. Validate another user
8. Validate Configurable security requirements
9. Import external Security Descriptor files
10. Review & fix validation failures
11. SCA RESTful API

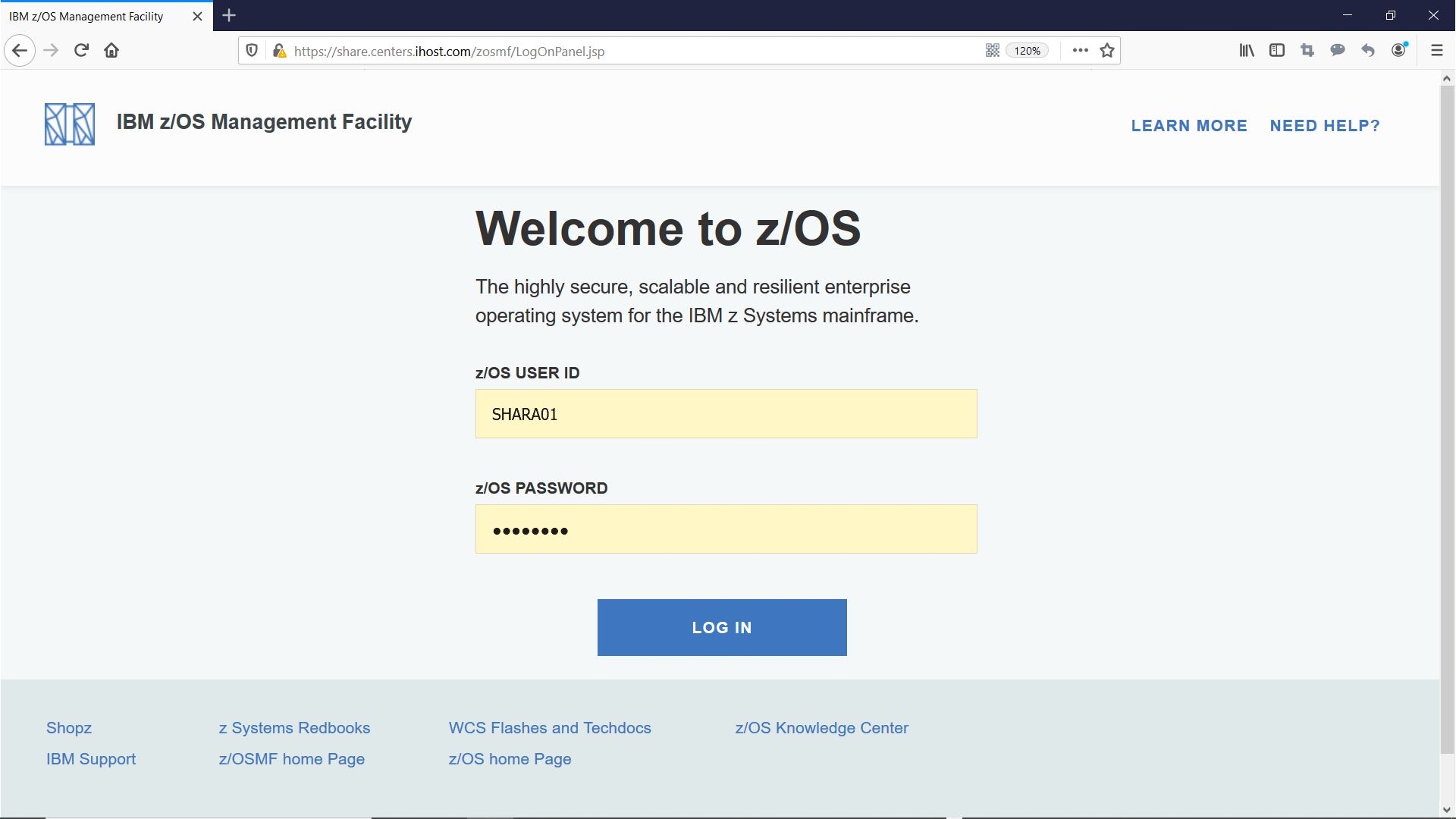
It is recommended that you execute these tasks in the order listed above. As you get familiar with the SCA, you will be able to work directly with the task you need to accomplish.

* 1. Logon to z/OSMF
* Launch browser from your workstation
* Point browser to z/OSMF – enter the following URL

[https://share.centers.ihost.com/zosmf/](https://share.centers.ihost.com/zosmf/LogOnPanel.jsp)

* Login with SHARE userid/pw as provided by the lab instructor
* Each workstation has been assigned a unique z/OS user id

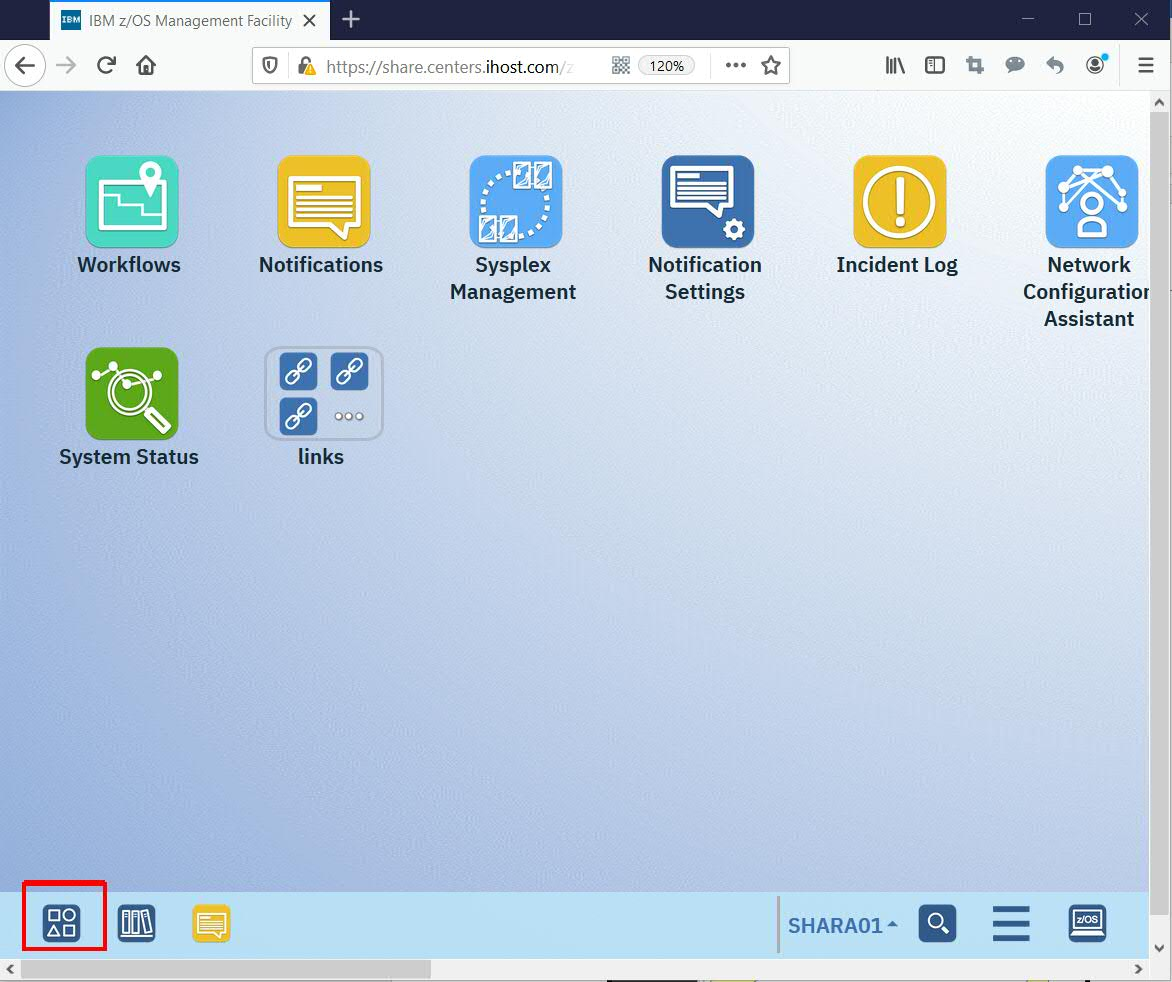
Note: All screen captures in the handout show the ID SHARA01, your browser will be slightly different to reflect the User ID that you were given.



* 1. Launch the Security Configuration Assistant task.

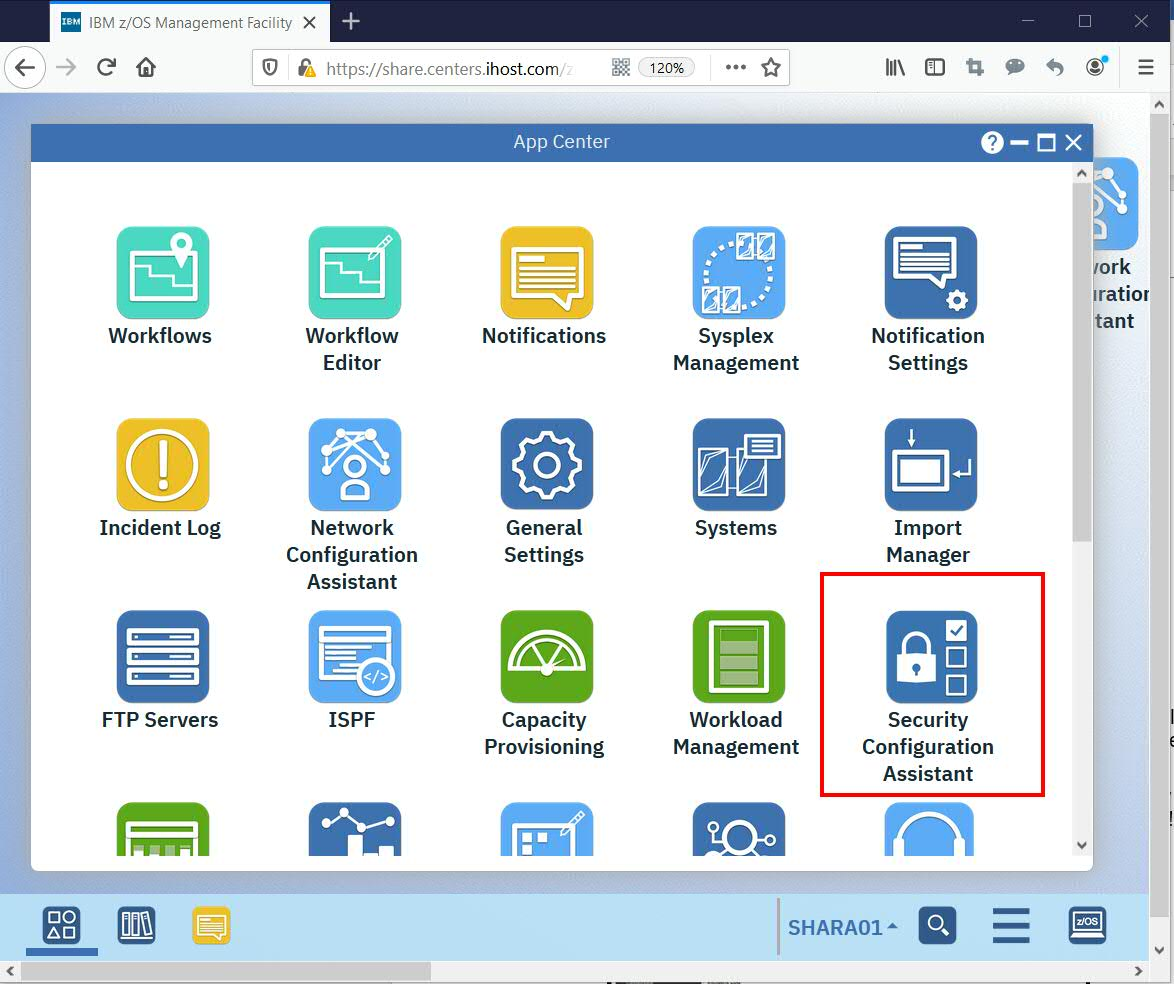
Step 2a: Open z/OSMF App Center

Click on the icon of App Center on the bottom left of z/OSMF desktop



Step 2b: Open Security Configuration Assistant task

You can enter character S to quickly locate the icon of “Security Configuration Assistant”. Then double click on the icon of “Security Configuration Assistant” to open Security Configuration Assistant (SCA) plugin. You can double click on the SCA window title to maximize the plugin window.



* 1. Check the result of Validation all.

**Step 3a: Validate all security requirements managed by SCA**

Click **Validate all,** you may wait for a few seconds for the validation process to be completed.



**Step 3b: Check the message of SCA Task**

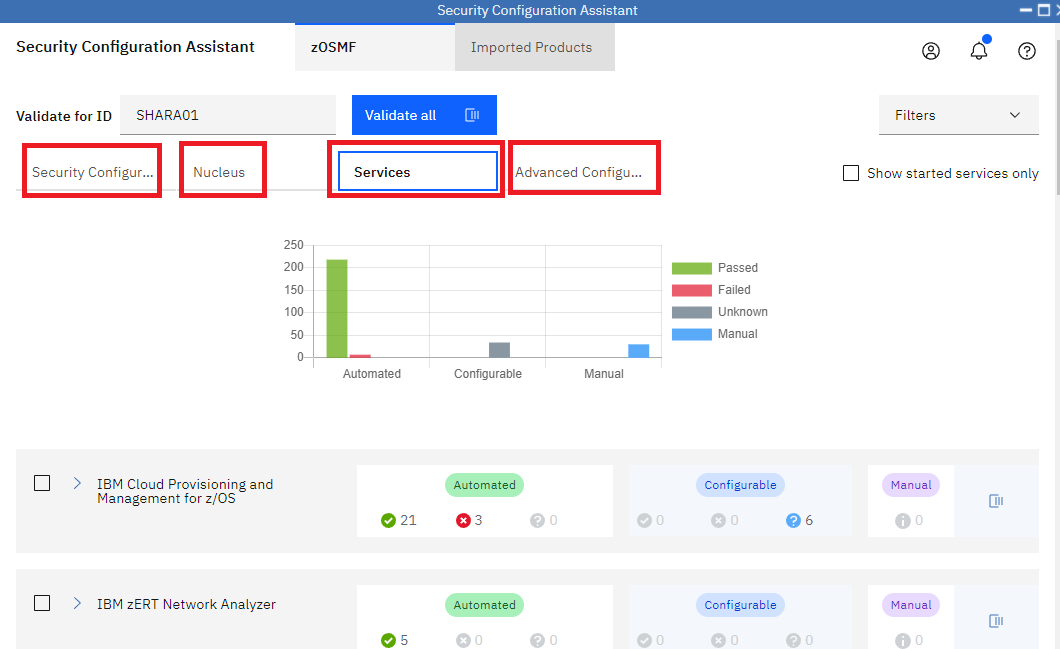
When the **Validation all** is completed, messages will be popped up to indicate the validation status. You can also check the messages by clicking on the bell icon on the top right.

A screenshot of a cell phone

Description automatically generated

**Step 3c: Check different tabs of SCA task**

You can see there are 4 tabs for security requirements of z/OSMF itself in SCA task. Each tab contains security requirements and result for different group of z/OSMF functions.



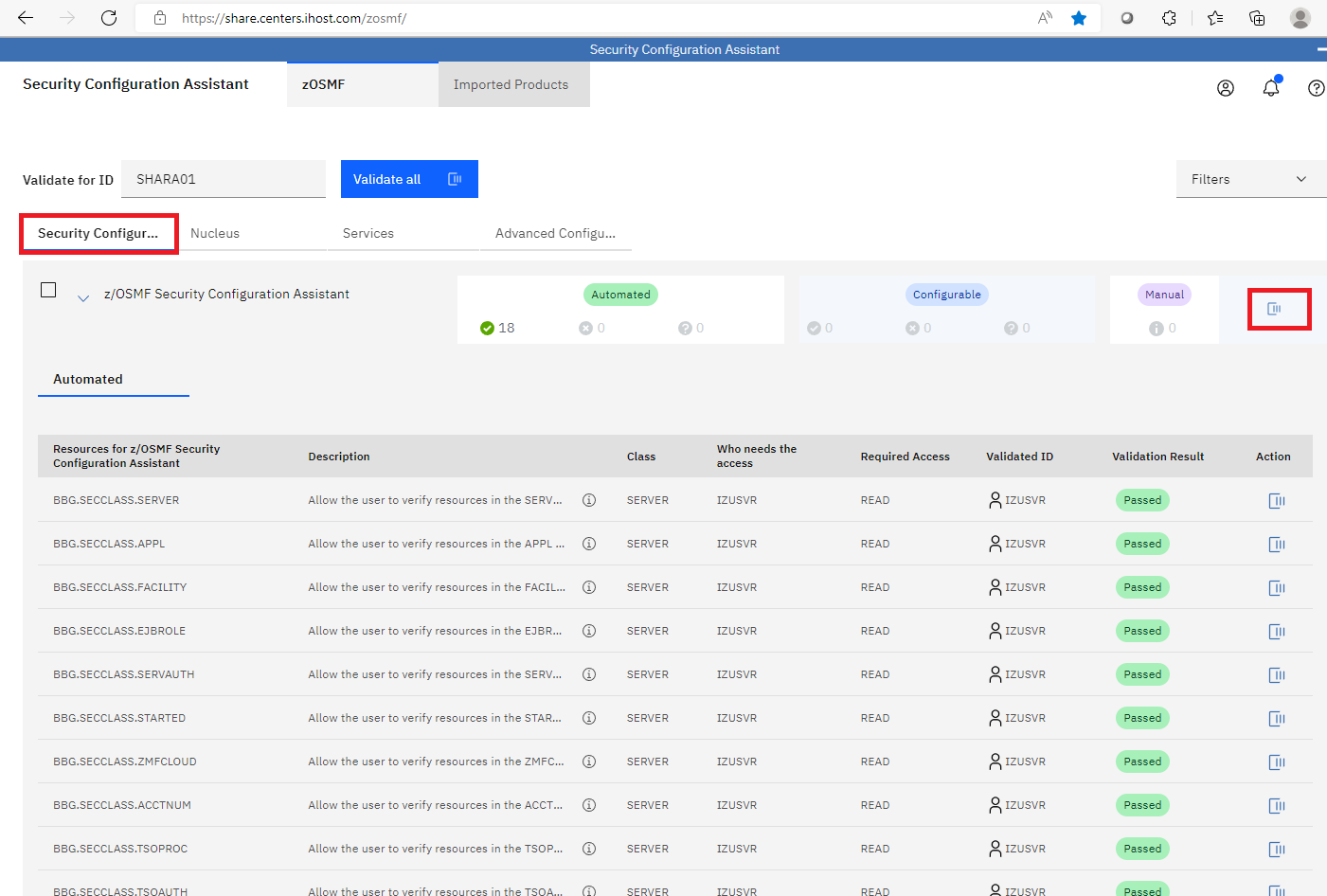
* 1. View the details of each tab

Now let’s check out details in each tab.

**Step 4a: Check the tab of Security Configuration Assistant**

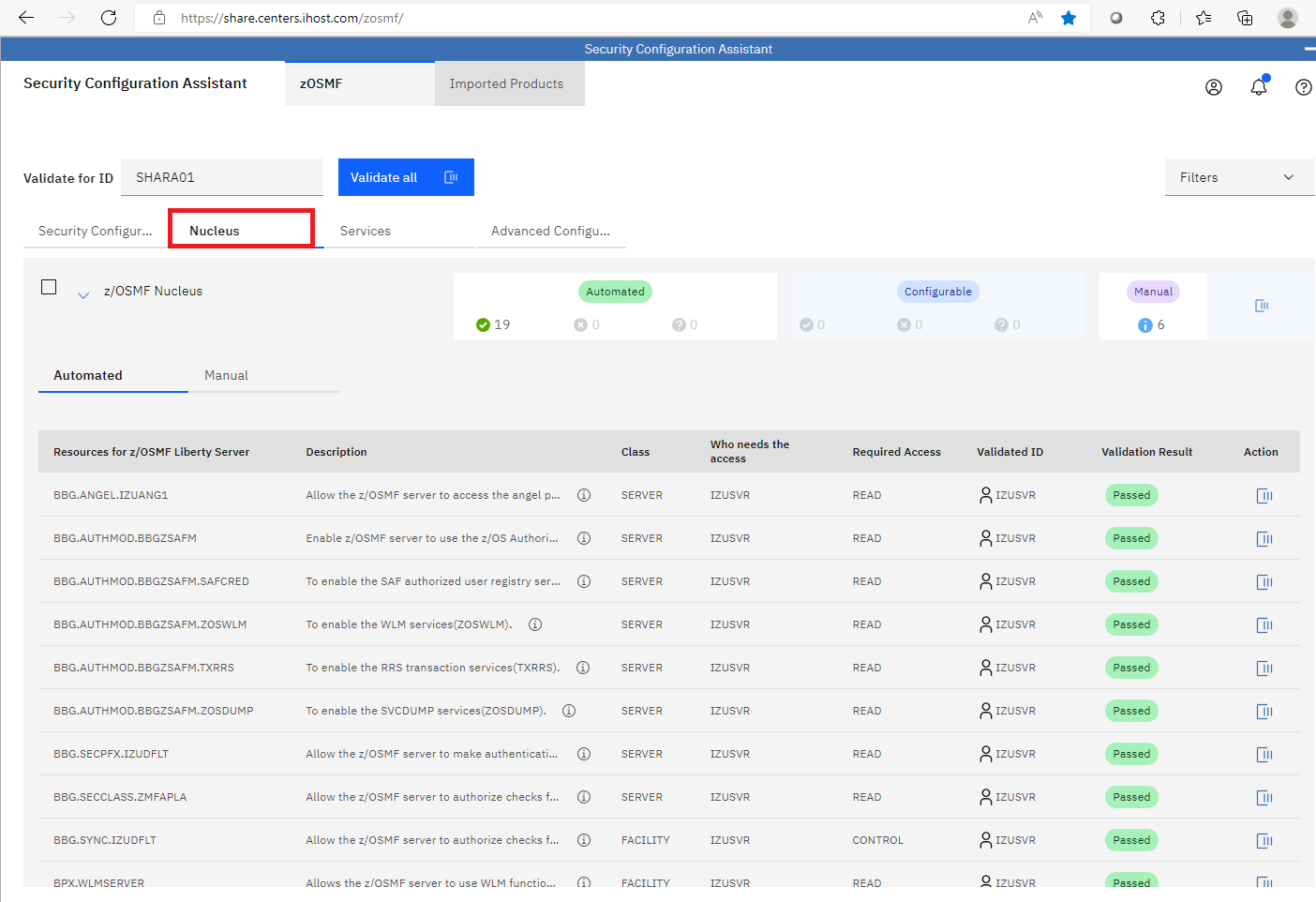
Extend the category of “z/OSMF Security Configuration Assistant”. The list in this tab includes all security requirements that are required to run the SCA task itself. If those security requirements are not satisfied, the later validations done by SCA task automatically may show the ‘Unknown’ status. Each line in the table indicates one specific security requirement which include:

* SAF resource name and explains why the authorization is needed.
* SAF resource class, Who needs access, access level
* User ID of the currently validated user.
* Validation result, which indicate if the authorities has been granted
* Action. If the corresponding security setup is changed later, you can rerun the validation for specific item to verify if the change was successful. To do so, click the refresh icon in this **Action** column. The Security Configuration Assistant task runs validation again to determine whether the user has the required level of access to the selected resource name



**Step 4b: Check the tab of Nucleus**

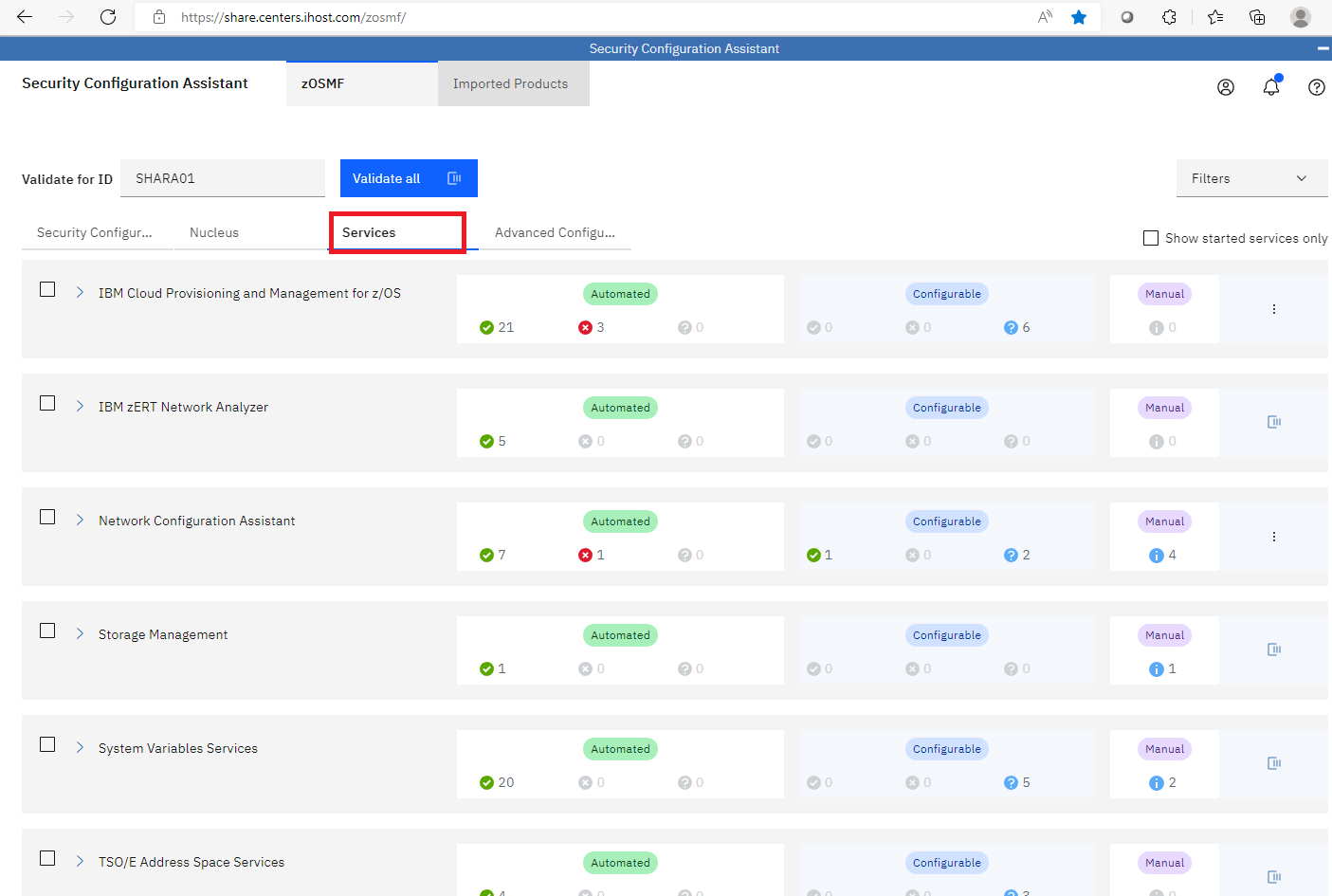
Extend the category of “z/OSMF Nucleus” and scroll down a little bit. The items in this tab includes all security requirement required for z/OSMF nucleus.



**Step 4c: Check the tab of Services**

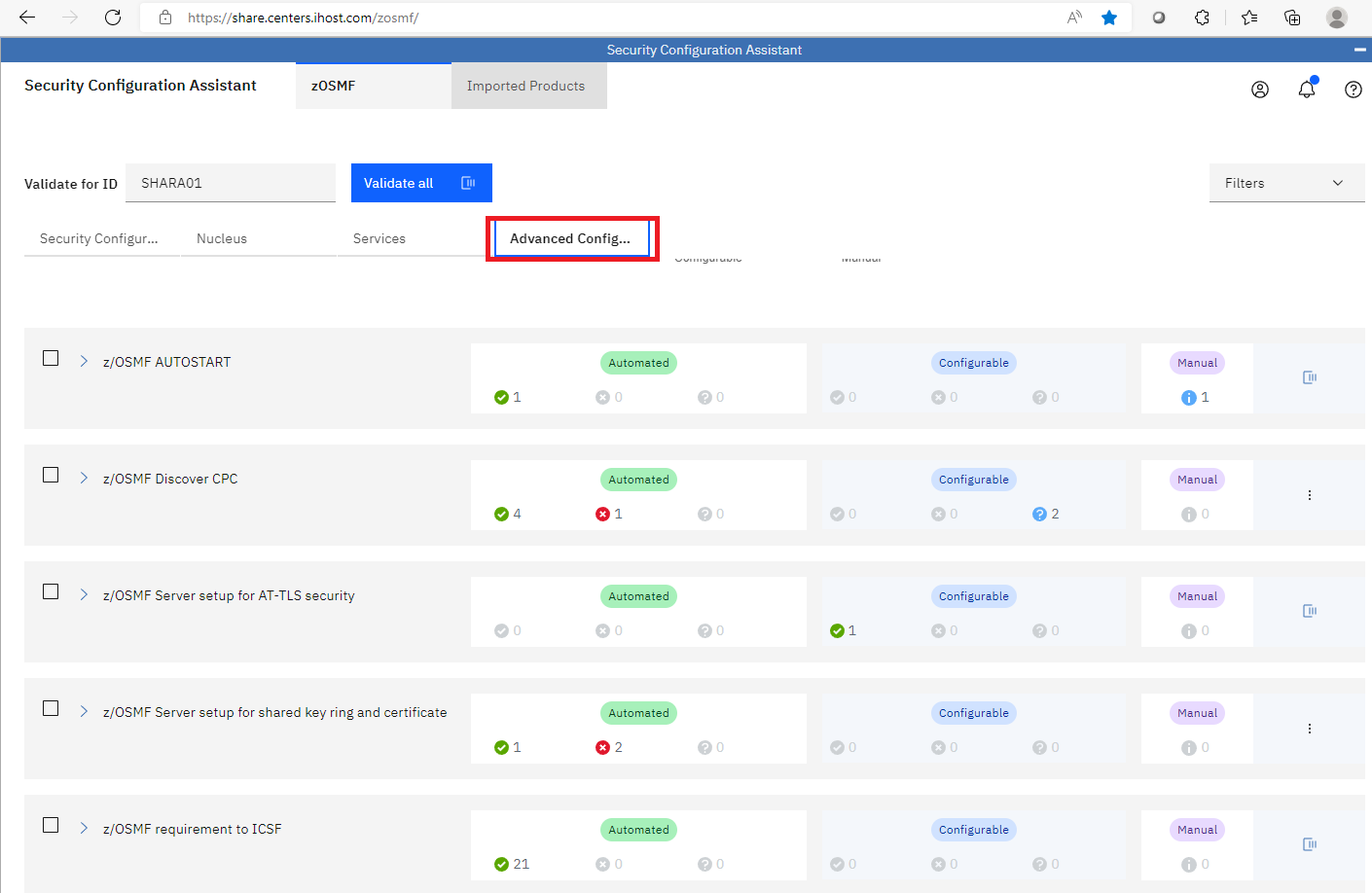
This tab includes security requirements for all z/OSMF services. They are grouped by service. You can extend each category to see the details of each z/OSMF service.

Administrator can leverage this tab to check if the security requirements of a specific z/OSMF service are satisfied or not.



**Step 4d: Check the tab of Advanced Configuration**

The items in this tab include security requirements for z/OSMF advanced configurations.



* 1. Check the statistics of validation

Now let’s continuously focus on “Advanced Configuration” tab and check out the statistics of validation result.

**Step 5a: Count the number of “Passed” and “Failed”**

Extend “z/OSMF Discover CPC” category and check out the number of “Passed” and “Failed”.

A screenshot of a computer

Description automatically generated

**Step 5b: Review the summarized numbers for a category**

There is a summary area for each category in the same row of the category name. It shows the summarized numbers of validation result for the specific category. It should be consistent with the number you counted in step 5a.

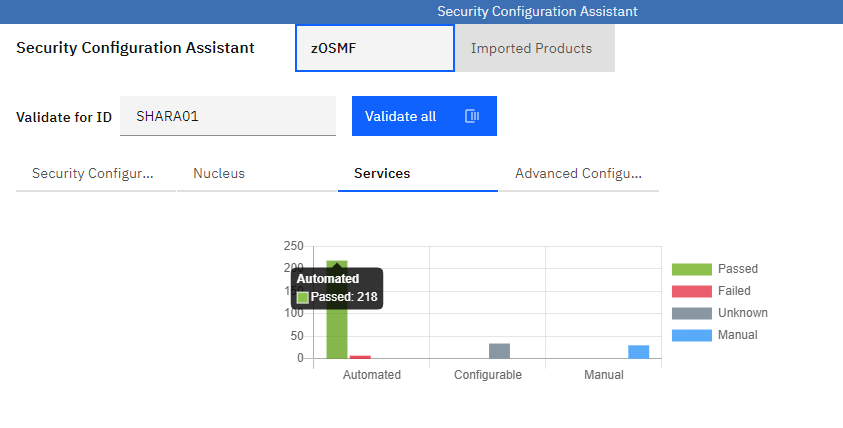
The Manual Checks indicates how many security requirements can not be automatically validated and require user’s manual check. You can click on the sub tab of “Manual” to see the details

A screenshot of a computer

Description automatically generated

**Step 5c: Check the overall summary via chart**

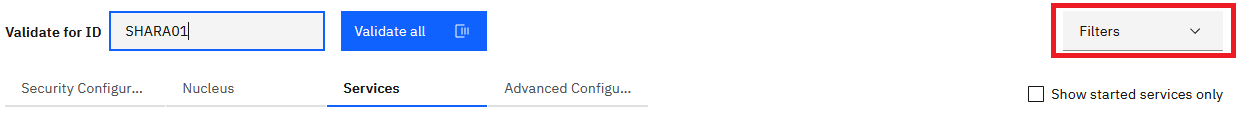
On the top of each tab, there is a chart summarizes the validation result for the specific tab.



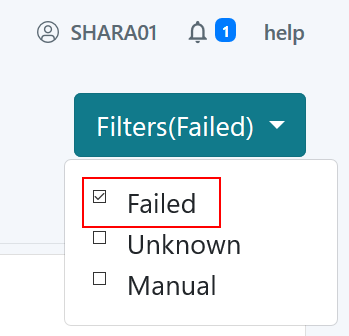
* 1. Filter out the failed validation

Sometimes, you may only care about the validation failures, the Filter function can help you with that.

Click on the button ‘Filters’ on the top right corner.



Then select ‘Failed’ option



Then extend some categories and only validation failures are displayed so that you can quickly find out what security requirements have not been satisfied.

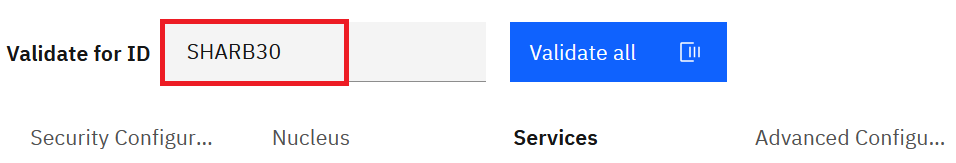
Unselect the ‘Failed’ check box in the Filter drop down menu so that we can continue with next step.

* 1. Validate another user

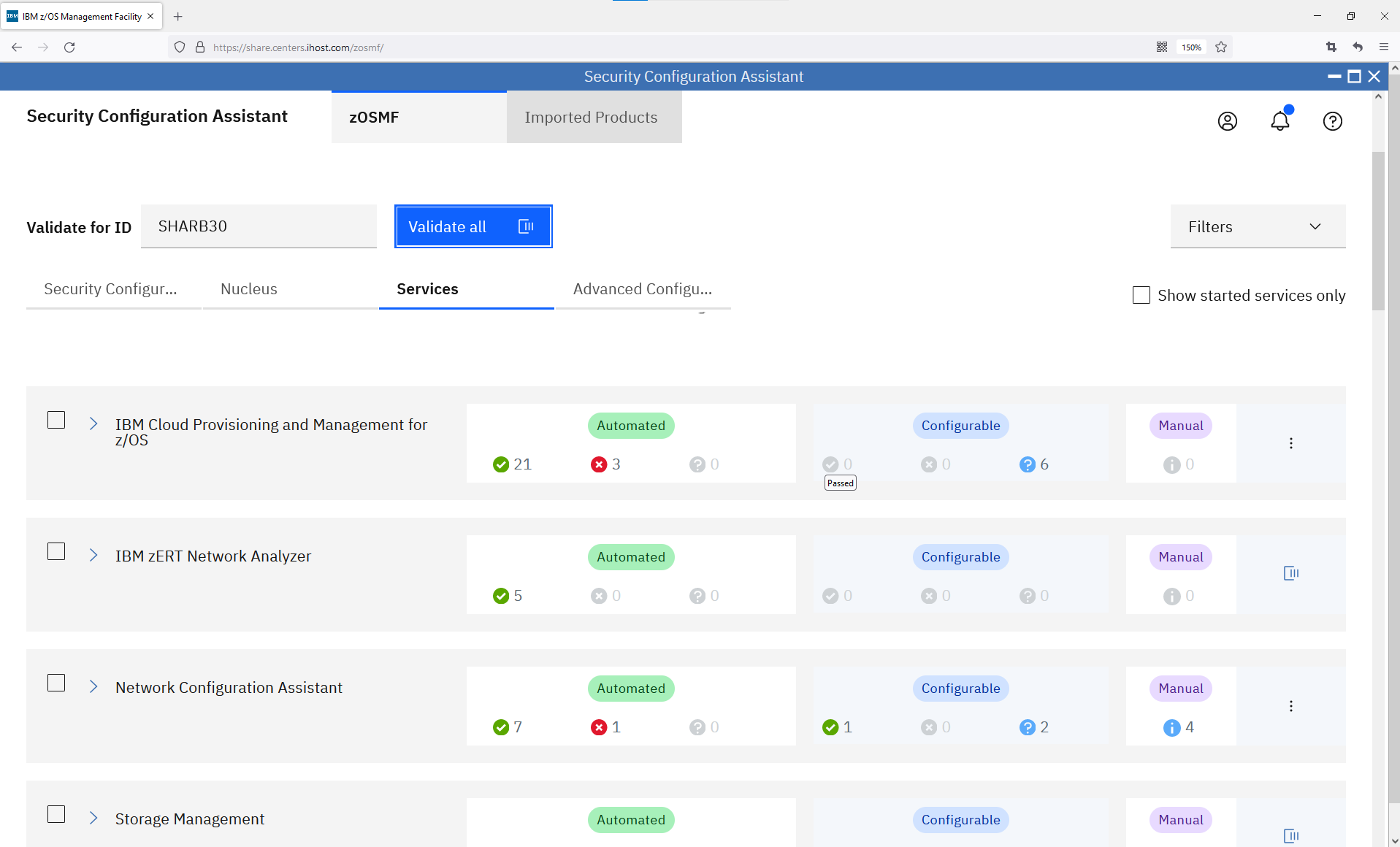
As an authorized administrator, you can validate z/OSMF security requirements for specified user id or group id.

**Step 7a:** **Specify a different user to be validated**

Specify the user id in the input box on the top

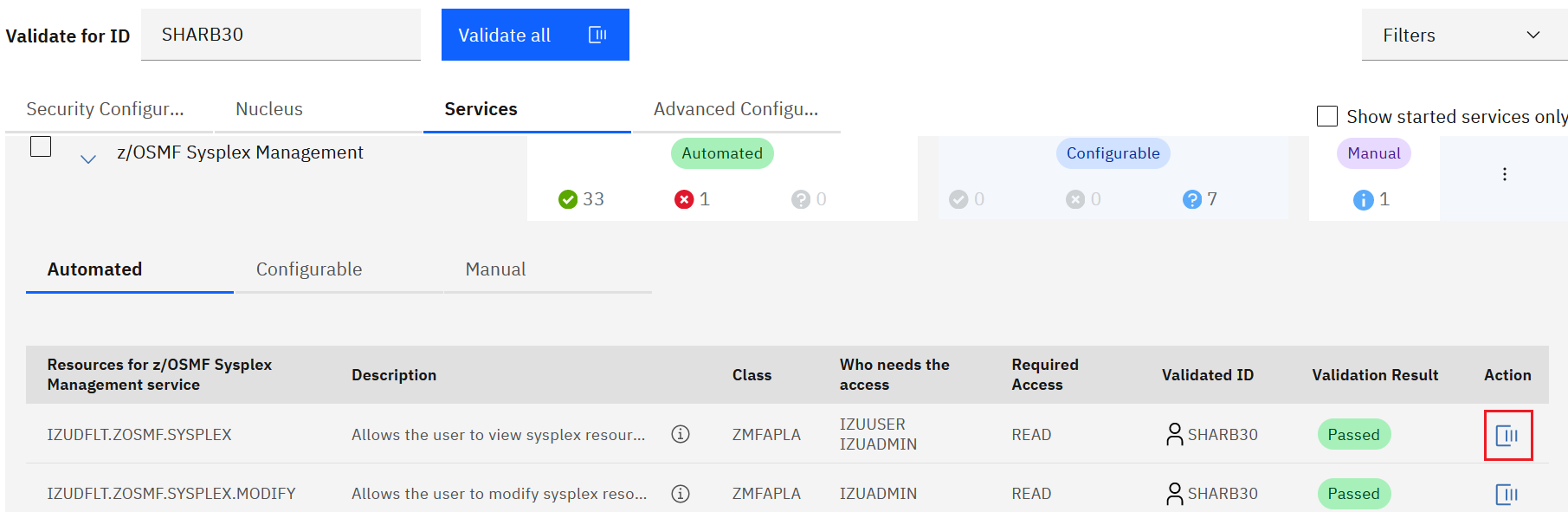


Then click on the “Services” tab to list all the services and plugins.

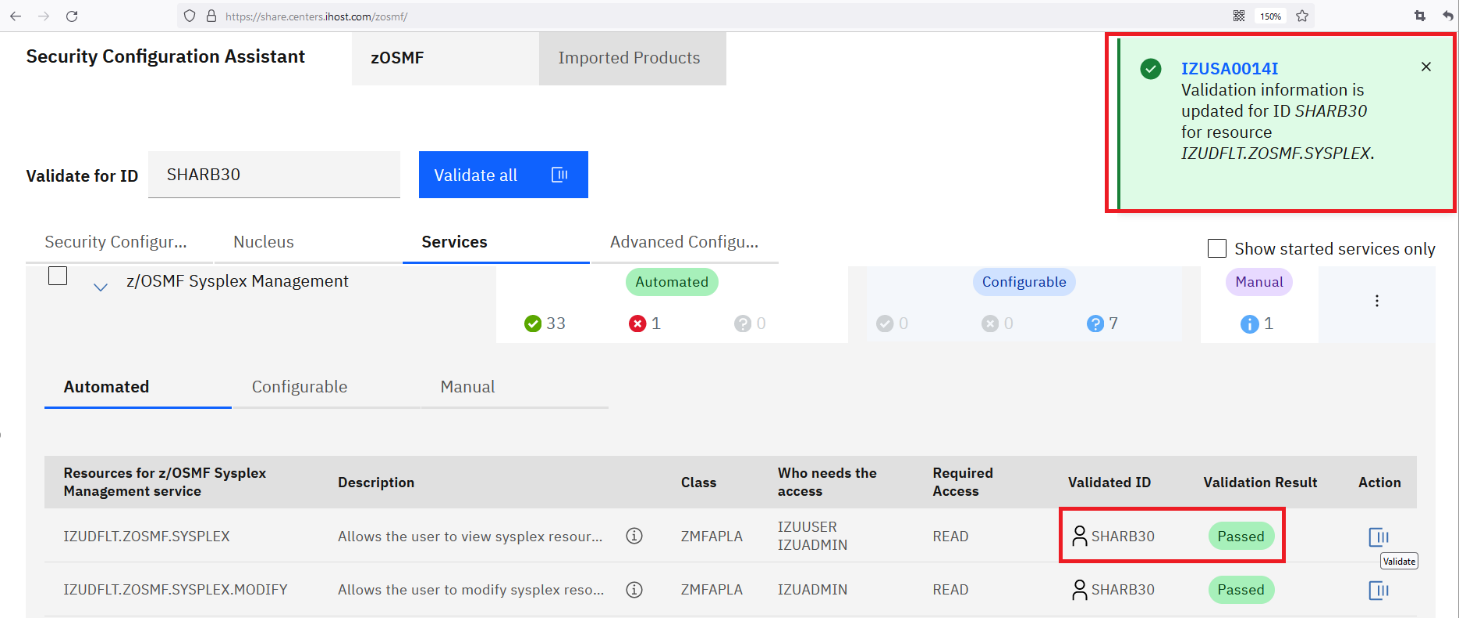


**Step 7b: Validate a specific security requirement for the specified user**

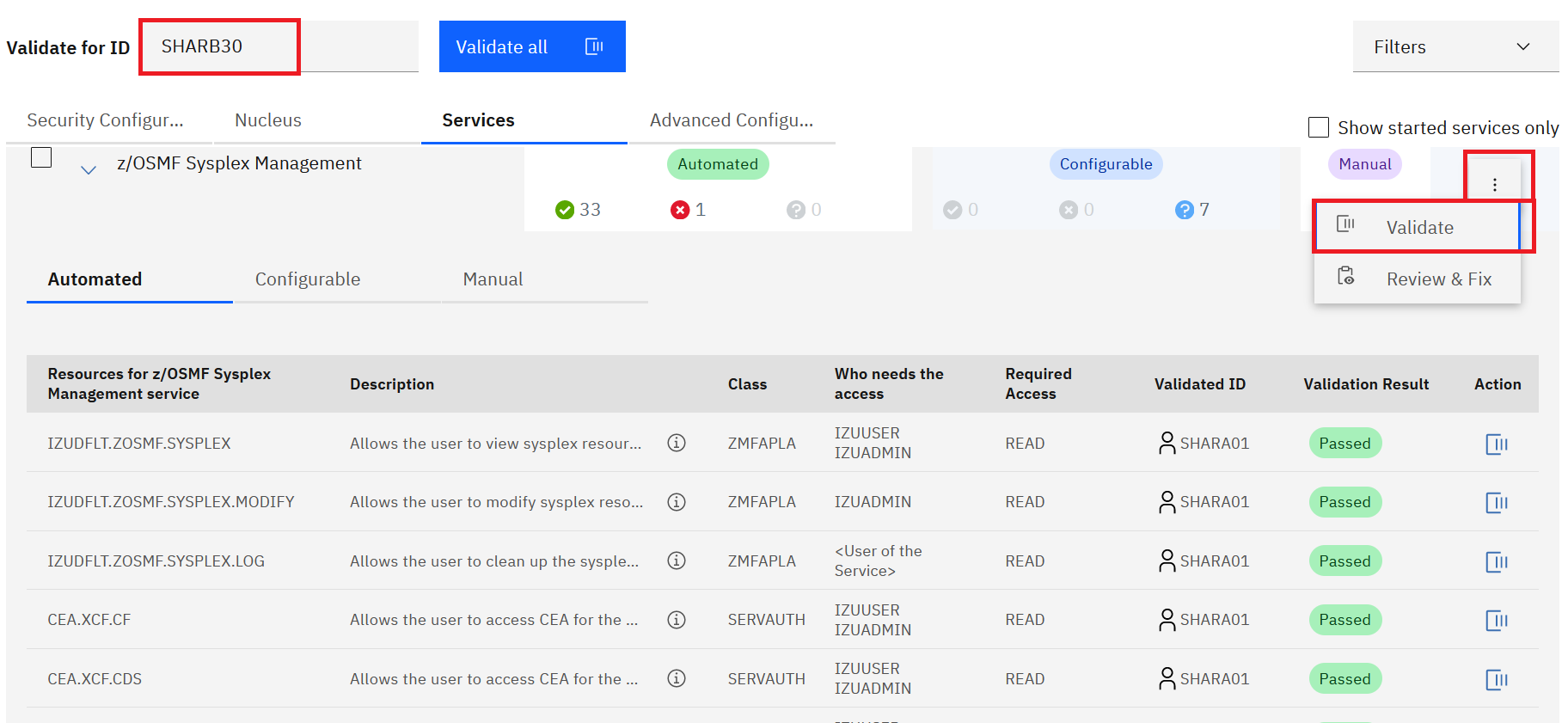
Extend z/OSMF Sysplex Management category, click on the ***Validate*** icon in the first row. The validation will be started to check if “SHARB30” has the “READ” access to z/OSMF Sysplex Management service which is protected by SAF profile IZUDFLT.ZOSMF.SYSPLEX.

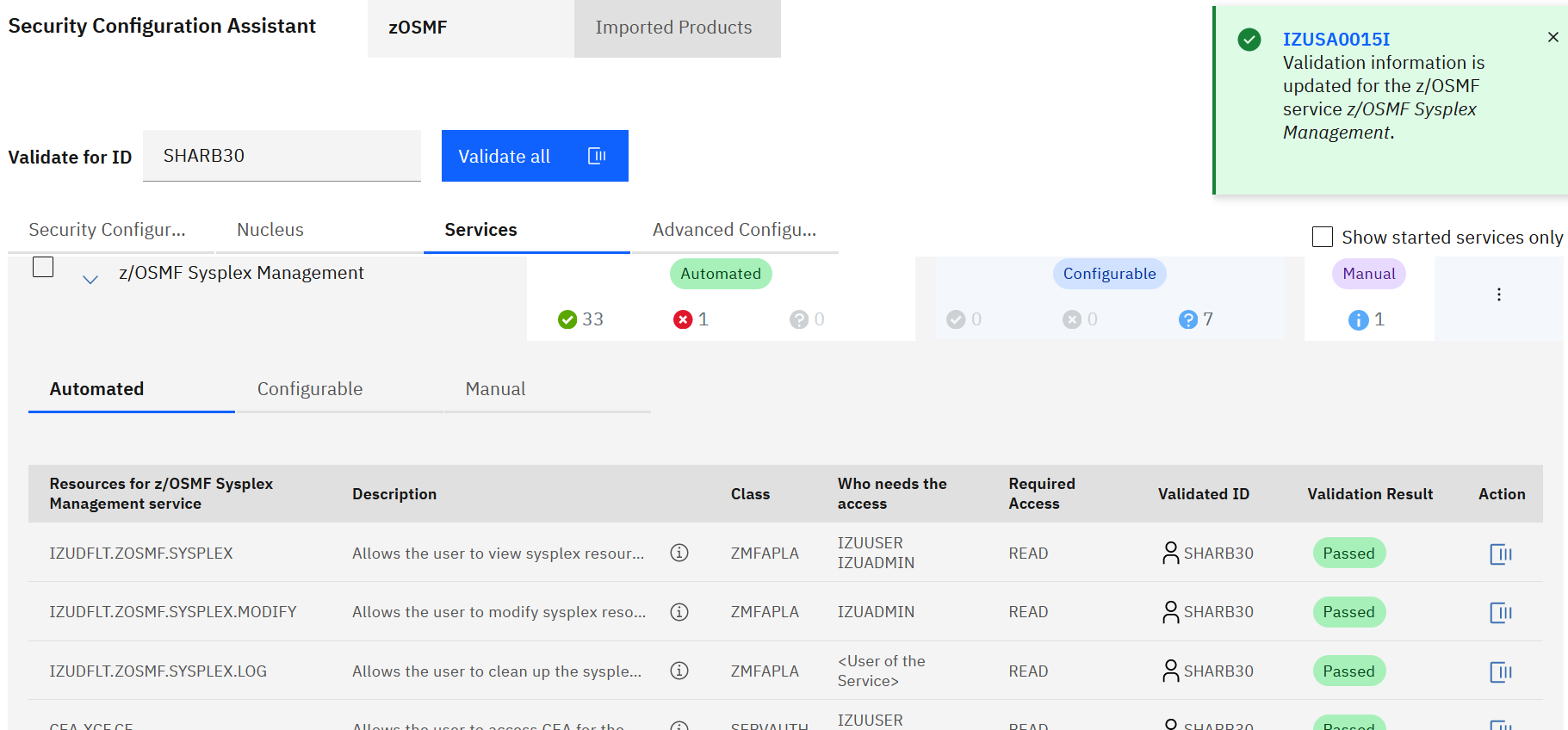
****

When the validation is completed, a message will be popped up to display the result of validation. The user id “SHARB30” is also displayed in the column of “Validated User ID”. Another column right after it shows the status of “Passed” which means the validation is successful.



**Step 7c: Run validation for a specific z/OSMF service**

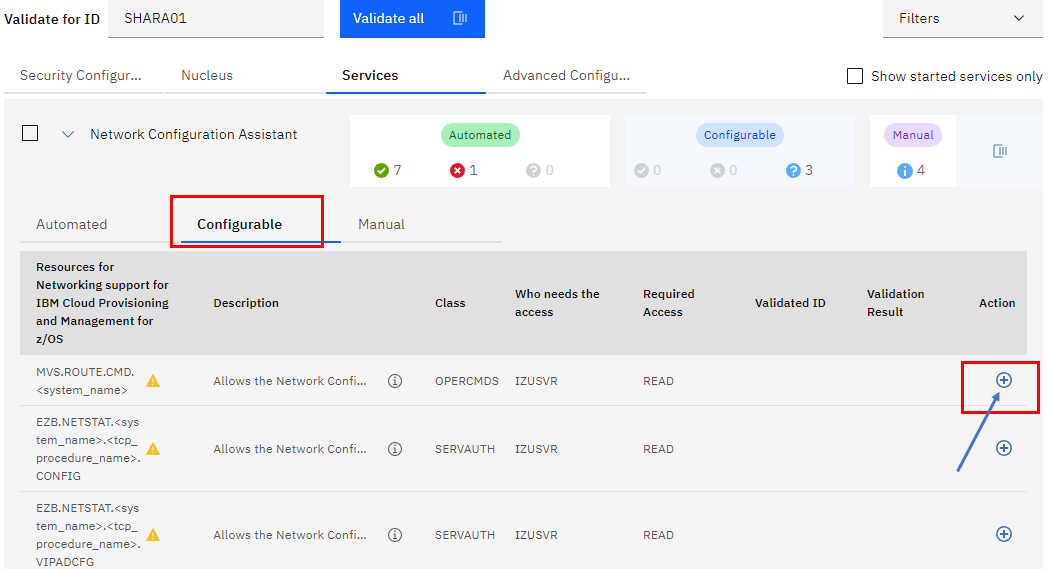
Click on the ***Validate*** icon on the same row with the category title “z/OSMF Sysplex Management”. This triggers validation for all the security requirements required by the service “z/OSMF Sysplex Management”. 

When the validation is completed, a message pops up and displays the result of validation. This operation is usually used to verify if a user can access a specific z/OSMF service. Depends on the different user you specified on the top, the validation result may vary. 

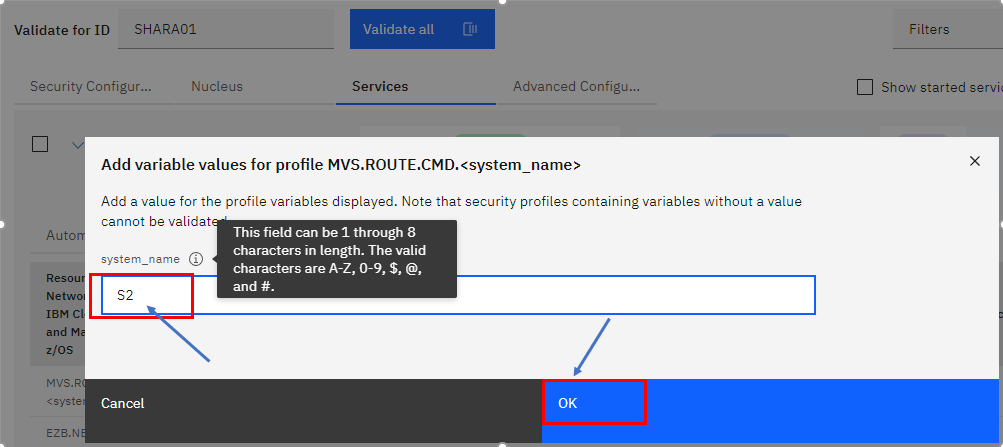
* 1. Validate Configurable security requirements

Component may have the **Configurable** security requirements which are placed in the “Configurable” tab. Specifically, those security resource names contain variables.

Extend “Network Configuration Assistant”, then click on “Configurable” tab. Click the + icon in **Action** column to add variable value for configurable requirements.

****

Input value **S2** for the variable name <system\_name>, and click **OK.**

****

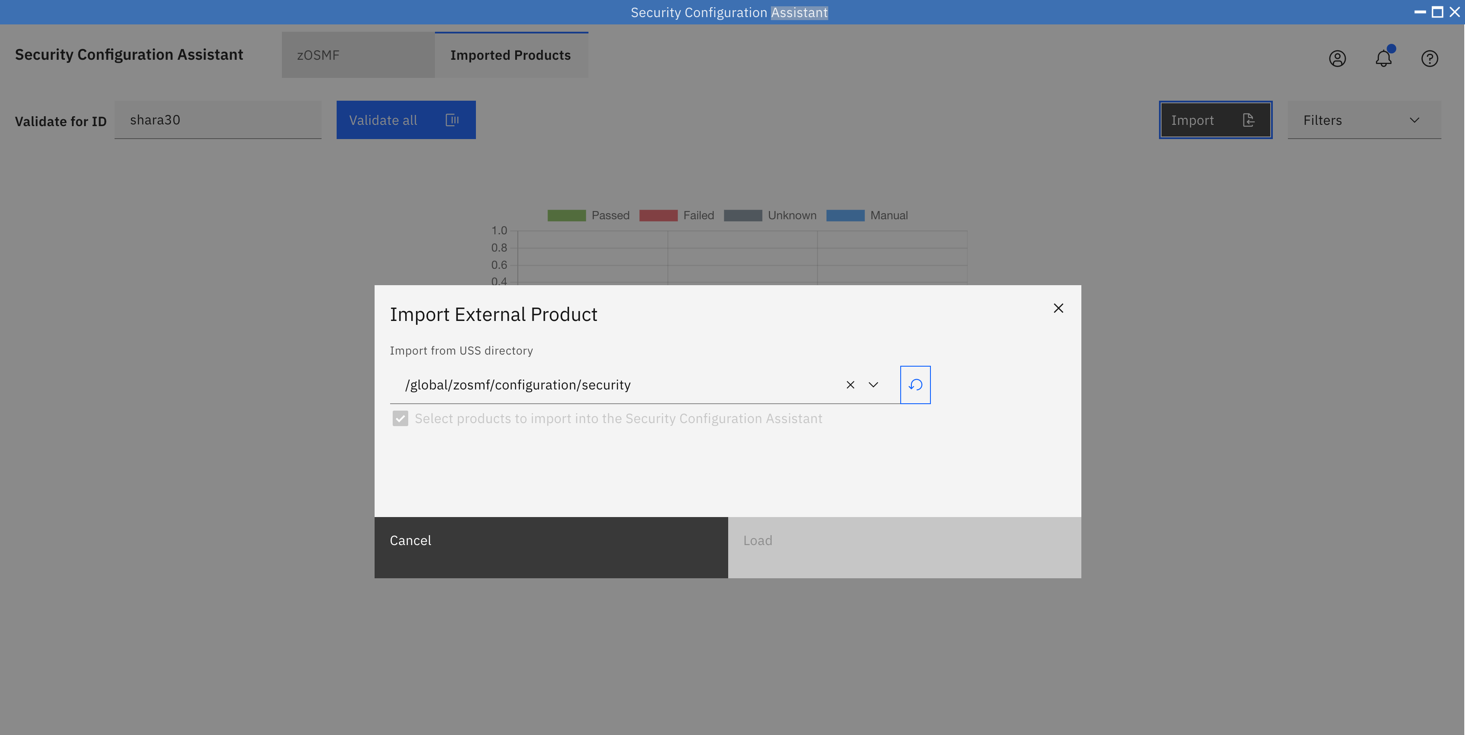
The added resource will be validated automatically.

* 1. Import external security descriptor file

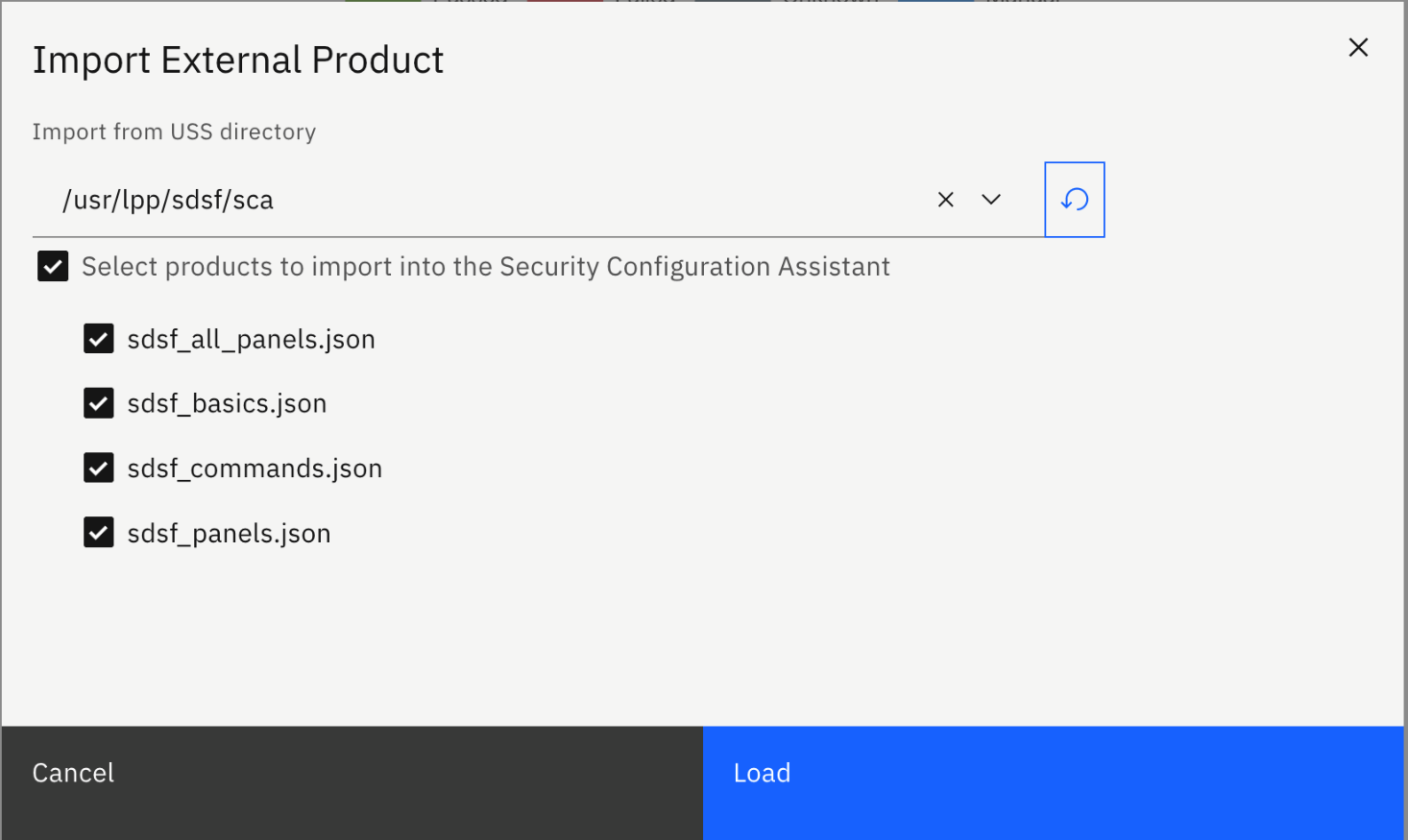
SCA supports other external products. Once you have a security descriptor file, you can use “Import” function of SCA to import security requirements of other products into SCA so that you can use SCA to organize, display, validate and even fix security requirements. SDSF exploits SCA via APAR PH53477. Let’s use SDSF as an example to see how to import external security descriptor file into SCA.

Please note, since the imported security requirements are visible to every SCA user, we have already imported SDSF security requirements into SCA. Below steps are just for your reference and you don’t need to perform them.

Click **Imported Products** tab, and then click **Import** button. The “Import External Product” dialog will be popped up.



You can then specify “/usr/lpp/sdsf/sca” in the input line because that’s the path SDSF ship its SCA security descriptor file. Once you entered the path, SCA automatically displays the list of security descriptor files for your selection.



When you click on “Load” button, the security requirements will then be loaded into SCA like below:



* 1. Review & Fix security failures

**Step 10a. Validate security requirements**

Ensure you are on the tab of “zOSMF” and select “Advanced Configuration” sub tab. Then extend “z/OSMF Discover CPC” category. Click on the 3 dot icon on the right side and then select Validate menu item.

A screenshot of a computer

Description automatically generated

You will see that 1 out of 5 security requirements is failed. Click on the 3 dot icon on the first row and now select “Review & fix” action like below:

A screenshot of a computer

Description automatically generated

A dialog will be popped up to display the generated commands for fixing this failure:

A screenshot of a computer

Description automatically generated

You can then review the commands and either send those commands to Security Administrator for reference or click on “Submit” to submit those commands to z/OS if you are authorized. For our lab, since you are not authorized, you will see below error:

A screenshot of a computer

Description automatically generated

* 1. SCA RESTful API

This step does not require your actions. It’s only for your reference.

With APAR PH41248 and PH39327, SCA now exposes its capability of automatic security validation and provisioning via REST API. Since REST API is easy to be consumed by many programming languages either locally or remotely, it’s now easy to consume SCA capability without having to open SCA UI.

Here is a example for using SCA REST API to do validation. The security requirements to be validated is directly included in the REST API request body:

Request:

|  |
| --- |
| Post 'https://share.centers.ihost.com/zosmf/config/security/v1/validate?userid=ibmuser'  {  "resourceItems": [  {  "resourceProfile": "IZUDFLT.ZOSMF.CONFIGURATION.SECURITY\_ASSISTANT",  "resourceClass": "ZMFAPLA",  "access": "READ"  }  ]  }' |

Response:

|  |
| --- |
| {  "resourceItems": [  {  "resourceProfile": "IZUDFLT.ZOSMF.CONFIGURATION.SECURITY\_ASSISTANT",  "resourceClass": "ZMFAPLA",  "access": "READ",  "action": "validate",  "validatedId": "ibmuser",  "status": "Passed"  }  ]  } |

Here is another example in which security requirements to be validated are specified in a standalone Security Descriptor file:

**R**equest:

|  |
| --- |
| Post 'https://share.centers.ihost.com/zosmf/config/security/v1/validate/descriptor?userid=ibmuser'  {  "path": "/usr/lpp/zosmf/configuration/izu5655S28SM01.json"  } |

Response:

|  |
| --- |
| {  "serviceId": "5655S28SM01",  "serviceName": "z/OSMF Security Configuration Assistant",  "version": "1.0",  "vendor": "IBM",  "resourceItems": [  {  "itemId": "5655S28SM01I00001000",  "itemType": "PROGRAMMABLE",  "itemCategory": "z/OSMF Security Configuration Assistant",  "itemDescription": "Allow the user to verify resources in the SERVER class.",  "resourceProfile": "BBG.SECCLASS.SERVER",  "resourceClass": "SERVER",  "whoNeedsAccess": "<IZU\_STARTED\_TASK\_USERID\_NAME>",  "access": "READ",  "action": "validate",  "validatedId": "ibmuser",  "status": "Passed"  }  ]  } |

Here is an example about using SCA REST API to provision security configuration. The security requirements are specified directly in the REST API request body:

Request:

|  |
| --- |
| Post 'https://share.centers.ihost.com/zosmf/config/security/v1/provision?userid=ibmuser'  {  "resourceItems": [  {  "resourceProfile": "IZUDFLT.ZOSMF.CONFIGURATION.SECURITY\_ASSISTANT",  "resourceClass": "ZMFAPLA",  "access": "READ"  }  ]  }' |

Response:

|  |
| --- |
| {  "resourceItems": [  {  "resourceProfile": "IZUDFLT.ZOSMF.CONFIGURATION.SECURITY\_ASSISTANT",  "resourceClass": "ZMFAPLA",  "access": "READ",  "actionObjectId": "ibmuser",  "status": "passed",  "action": "provision",  "validatedId": "ibmuser"  }  ]  } |

The security requirements can also be saved in a standalone file:

**R**equest:

|  |
| --- |
| Post 'https://share.centers.ihost.com/zosmf/config/security/v1/provision/descriptor?userid=ibmuser'  {  "path": "/usr/lpp/zosmf/configuration/izu5655S28SM01.json"  } |

Response:

|  |
| --- |
| {  "serviceId": "5655S28SM01",  "serviceName": "z/OSMF Security Configuration Assistant",  "version": "1.0",  "vendor": "IBM",  "resourceItems": [  {  "resourceProfile": "BBG.SECCLASS.SERVER",  "resourceClass": "SERVER",  "access": "READ",  "actionObjectId": "ibmuser",  "status": "passed",  "itemId": "5655S28SM01I00001000",  "itemType": "PROGRAMMABLE",  "itemCategory": "z/OSMF Security Configuration Assistant",  "itemDescription": "Allow the user to verify resources in the SERVER class.",  "whoNeedsAccess": "<IZU\_STARTED\_TASK\_USERID\_NAME>",  "action": "validate",  "validatedId": "ibmuser"  } ]  } |

#### End of exercise