Technical Document: ServiceNow to SailPoint Role Request Workflow

## May 14, 2025

**Contents**

1. [Overview](#_bookmark0) 2
2. [Process Flow](#_bookmark1) 2
   1. [ServiceNow Initiates Role Request](#_bookmark2) 2
      1. [JSON Payload Example](#_bookmark3) 2
   2. [SailPoint Workflow: ESD Role Request Workflow](#_bookmark4) 2
      1. [Key Steps in the Workflow](#_bookmark5) 3
   3. [Rule: ESD-12 Role Request Plan Generation](#_bookmark6) 3
      1. [Logic](#_bookmark7) 3
      2. [Key Code Snippet](#_bookmark8) 4
   4. [Rule: ESD-12 Z Account Selection](#_bookmark9) 5
      1. [Logic](#_bookmark10) 5
      2. [Key Code Snippet](#_bookmark11) 5
   5. [Custom Object: ESD-12 Custom](#_bookmark12) 6
      1. [Key Attributes](#_bookmark13) 6
   6. [Account Provisioning](#_bookmark14) 6
      1. [Z Account (Active Directory)](#_bookmark15) 6
      2. [ZC Account (Azure)](#_bookmark16) 7
   7. [Notify ServiceNow](#_bookmark17) 7
      1. [Key Code Snippet](#_bookmark18) 7
3. [Summary](#_bookmark19) 7

# Overview

This document outlines the process of handling role requests initiated from ServiceNow to SailPoint IdentityIQ (IIQ) for provisioning Z accounts (privileged accounts in Active Directory) and ZC accounts (Azure cloud accounts). The workflow involves creating or modifying accounts, assigning roles, and managing AD groups or Azure cloud groups based on the request type. The process leverages SailPoint workflows, rules, and custom objects to automate account provisioning and role assignment.

# Process Flow

## ServiceNow Initiates Role Request

ServiceNow sends a role request to SailPoint via a REST API call. The request is a POST request to the SailPoint endpoint [http://silvmspiqd01.ent.dir.labor.gov:8080/identityiq.](http://silvmspiqd01.ent.dir.labor.gov:8080/identityiq) The JSON payload contains the necessary parameters for the role request.

* + 1. JSON Payload Example

1

{

" work Flow Args ": { " ritm ": " RITM 0540493 ",

" identity Name ": " May\_User4 ", " email ": " May\_User4 @ dev . dol. gov ", " accountType ": " ZC",

" role ": " Desktop - Tier 3 Support", " requestType ": " Add / Modify ", " launcher ": " spadmin "

}

}

2

3

4

5

6

7

8

9

10

11

* **ritm**: Request ticket number (e.g., RITM0540493).
* **identityName**: The name of the user (e.g., May\_User4).
* **email**: User’s email address (e.g., [May\_User4@dev.dol.gov).](mailto:May_User4@dev.dol.gov)
* **accountType**: Type of account (e.g., Z for AD privileged accounts, ZC for Azure accounts).
* **role**: The business role to assign (e.g., Desktop - Tier 3 Support).
* **requestType**: Type of request (e.g., Add/Modify, New Account Request, Delete Account).
* **launcher**: The user or system initiating the request (e.g., spadmin).

## SailPoint Workflow: ESD Role Request Workflow

The SailPoint workflow ESD Role Request Workflow (filename: ESD Role Request Workflow.xml) is triggered upon receiving the REST API call. This workflow processes the JSON payload and orches- trates the provisioning process.

* + 1. Key Steps in the Workflow
       1. **Extract Parameters**: The workflow extracts the parameters from the JSON payload (e.g., identityName,

role, accountType, requestType).

### Generate Provisioning Plan:

* + - * + The workflow references the rule ESD-12 Role Request Plan Generation (filename:

ESD-12 Role Request Plan Generation.xml) to create a provisioning plan.

* + - * + The rule takes inputs such as roleIds, identityName, requestType, and accountType

to generate a provisioning plan for role assignment or account creation.

### Account Selection:

* + - * + The workflow uses the rule ESD-12 Z Account Selection (filename: ESD-12 Z Account Selection.xml) to determine if the user already has a Z account in AD or a ZC account in Azure.
        + If no account exists, a new account is created with a specific naming convention.

### Provision Account and Assign Roles:

* + - * + Based on the provisioning plan, SailPoint provisions the account in AD (for Z accounts) or Azure (for ZC accounts) and assigns the appropriate roles.

### Post-Provisioning:

* + - * + For Z accounts, a PAM (Privileged Access Management) account is created (covered in a separate document).
        + For ZC accounts, no PAM account creation is required.
      1. **Notify ServiceNow**: The workflow notifies ServiceNow of the provisioning result.

## Rule: ESD-12 Role Request Plan Generation

This rule is responsible for generating the provisioning plan based on the request type (New Account Request, Add/Modify, Delete Account).

* + 1. Logic

### Inputs:

* + roleIds: List of role IDs to assign.
  + identityName: Name of the identity (e.g., May\_User4).
  + requestType: Type of request (e.g., Add/Modify).
  + nativeId: Native identity for the account (if applicable).

### Custom Object Reference:

* + The rule references the custom object ESD-12 Custom (filename: ESD-12 Custom.xml) to retrieve configuration values such as Z\_FORMAT, Z\_AD\_OU, ZC\_ROLE\_NAME\_FILTER, and Z\_ROLE\_NAME\_FILTER.

### Provisioning Plan Creation:

* + For New Account Request:
    - Creates a new account request for the application IIQ.
    - Assigns roles based on the roleIds.
    - Differentiates between Z roles (on-premises) and ZC roles (cloud) using filters (Z\_ROLE\_NAME\_FILTER and ZC\_ROLE\_NAME\_FILTER).
  + For Add/Modify:
    - Modifies an existing account by adding or removing roles.
  + For Delete Account:
    - Deletes the specified account.

### Output:

* + Returns a ProvisioningPlan object that SailPoint uses to execute the provisioning.
    1. Key Code Snippet

1 if ( requestType . equalsIgnore Case ( new AccoutValue )) {

2 AccountRequest on PremacctReq = null;

3 AccountRequest on CloudacctReq = null;

4 attrOpr = Provisioning Plan . Operation . Add ;

5

6 for ( String roleId : role Ids) {

7 Attribute Request attrReq = new Attribute Request ();

8 attrReq . setOp ( attrOpr);

9 Attributes attr = new Attributes ();

10 attr. put(" assignment", true );

11 attrReq . setArgs( attr);

12 attrReq . setName (" assigned Roles ");

13 Bundle role Obj = context. getObjectBy Name ( Bundle . class , roleId );

14 attrReq . setValue ( role Obj. getName ());

15

16 if ( roleObj. getName (). to LowerCase (). endsWith ( zRole Name Filter )) {

17 if ( on PremacctReq == null) {

18 on PremacctReq = new AccountRequest ();

19 on PremacctReq . setOperation ( AccountRequest . Operation . Modify );

20 on PremacctReq . setApplication (" IIQ ");

21 on PremacctReq . add ( attrReq );

22 } else {

23 on PremacctReq . add ( attrReq );

24 }

25 } else if ( roleObj. getName (). to LowerCase (). endsWith ( zcRole Name Filter )) {

26 if ( on CloudacctReq == null) {

27 on CloudacctReq = new AccountRequest ();

28 on CloudacctReq . setOperation ( AccountRequest . Operation . Modify );

29 on CloudacctReq . setApplication (" IIQ ");

30 on CloudacctReq . add ( attrReq );

31 } else {

32 on CloudacctReq . add ( attrReq );

33 }

34 }

35 }

36 plan . setSource (" LCM ");

37 if ( on CloudacctReq != null) {

38 plan . add ( on CloudacctReq );

39 }

40

if ( on PremacctReq != null) {

plan . add ( on PremacctReq );

}

}

41

42

43

## Rule: ESD-12 Z Account Selection

This rule determines whether the user already has a Z account in AD or a ZC account in Azure. If no account exists, it creates a new account with a specific naming convention.

* + 1. Logic

### Inputs:

* + identity: The identity object of the user.
  + links: List of existing accounts (links) associated with the identity.

### Custom Object Reference:

* + Retrieves Z\_FORMAT (e.g., "Z-") and Z\_AD\_OU (organizational unit in AD) from the custom object ESD-12 Custom.

### Account Selection:

* + Checks if the user has an existing Z account by looking for an account with sAMAccountName starting with Z\_FORMAT.
  + If no Z account exists, creates a new account with:
    - sAMAccountName: Z-<firstname><lastname> (e.g., Z-MayUser4).
    - Distinguished Name (DN): cn=Z-<firstname><lastname>,<Z\_AD\_OU> (e.g., cn=Z-MayUser4,ou=

### Output:

* + Returns a Link object representing the selected or newly created account.
    1. Key Code Snippet

1

Link selected Link = null;

Custom custObj = context. getObjectBy Name ( Custom . class , " ESD -12 Custom ")

;

Map app Map = custObj. getAttributes (); String zFormat = app Map . get(" Z\_FORMAT "); String zOu = app Map . get(" Z\_AD\_OU ");

if ( links != null) {

for ( Link link : links) {

String id = link . getAttribute (" sAMAccountName "). to String (); if ( id. to LowerCase (). startsWith ( zFormat)) {

selected Link = link ;

}

}

}

if ( selected Link == null) {

String iden Name = identity . getName (); iden Name = zFormat + iden Name ;

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

String natId = " cn=" + iden Name + "," + zOu ;

Link \_nlink = new Link ();

\_nlink . setNative Identity ( natId ); selected Link = \_nlink ;

}

return selected Link ;

20

21

22

23

24

25

## Custom Object: ESD-12 Custom

The custom object ESD-12 Custom (filename: ESD-12 Custom.xml) stores configuration values used by the rules.

* + 1. Key Attributes
* Z\_FORMAT: Prefix for Z accounts (e.g., "Z-").
* Z\_AD\_OU: Organizational unit for Z accounts in AD (e.g., ou=PrivilegedAccounts,dc=ent,dc=dir,dc=labo
* ZC\_ROLE\_NAME\_FILTER: Filter for ZC roles (e.g., "\_zc").
* Z\_ROLE\_NAME\_FILTER: Filter for Z roles (e.g., "\_z").
* NEW\_ACC\_REQ\_FORM\_STR: Value for new account requests (e.g., "New Account Request").
* ADD\_ROLES\_FORM\_STR: Value for adding roles (e.g., "Add/Modify").
* REMOVE\_ROLES\_FORM\_STR: Value for removing roles (e.g., "Remove Roles").
* DELETE\_ACC\_FORM\_STR: Value for deleting accounts (e.g., "Delete Account").

## Account Provisioning

* + 1. Z Account (Active Directory)

### Account Creation:

* + If the user does not have a Z account, a new AD account is created with:
    - sAMAccountName: Z-<firstname><lastname> (e.g., Z-MayUser4).
    - DN: Based on the Z\_AD\_OU from the custom object.
  + The AD provisioning policy assigns the user to the appropriate AD groups tied to the IT roles associated with the business role (e.g., Desktop - Tier 3 Support).

### Role Assignment:

* + For Add/Modify requests, the provisioning plan adds or removes AD groups as needed.

### Post-Provisioning:

* + A PAM (Privileged Access Management) account is created (details in a separate document).
    1. ZC Account (Azure)

### Account Creation:

* + If the user does not have a ZC account, a new Azure account is created with:

\* UPN (User Principal Name): zc-<firstname><lastname>@domain (e.g., [zc-MayUser4@dev.dol.gov](mailto:zc-MayUser4@dev.dol.gov)

* + The provisioning plan assigns the user to Azure cloud groups tied to the IT roles associated with the ZC business role.

### Role Assignment:

* + For Add/Modify requests, the provisioning plan adds or removes Azure groups as needed.

### Post-Provisioning:

* + No PAM account creation is required for ZC accounts.

## Notify ServiceNow

The workflow checks the provisioning result using the Check Provisioning Success step (from ESD Role Request Workflow.xml). If successful, it notifies ServiceNow of the outcome.

* + 1. Key Code Snippet

1 if ( project != null) {

2 logger. debug (" Project Executed in identity :\ n" + project. to Xml ());

3 List < Provisioning Plan > plans = project. getPlans ();

4 if ( plans != null) {

5 for ( Provisioning Plan plan1 : plans) {

6 List < AccountRequest > accReqsts = plan1 . getAccountRequests ();

7 if ( accReqsts != null) {

8 for ( AccountRequest accReq : accReqsts) {

9 Provisioning Result res = ( Provisioning Result ) accReq . getResult ();

10 String prov Status = res. getStatus ();

11 logger. debug ( accReq . getApplication () + " Provisioning status is: " + res. getStatus ());

12 if ( prov Status != null && ( prov Status. equalsIgnore Case ( Provisioning Result . STATUS\_COMMITTED ) || prov Status. equalsIgnore Case ( Provisioning Result . STATUS\_QUEUED ))) {

13 prov Successful = " true ";

14 } else {

15 prov Successful = " false ";

16 return prov Successful ;

17 }

18 }

19 }

20 }

21 }

22 }

23 return prov Successful ;

# Summary

* **ServiceNow** initiates a role request via a REST API call to SailPoint with a JSON payload.
* The ESD Role Request Workflow processes the request, generates a provisioning plan using the ESD-12 Role Request Plan Generation rule, and selects or creates an account using the ESD-12 Z Account Selection rule.
* **Z Accounts** are provisioned in AD with a Z-<firstname><lastname> naming convention, as- signed AD groups, and followed by PAM account creation.
* **ZC Accounts** are provisioned in Azure with a zc-<firstname><lastname> UPN, assigned cloud groups, and do not require PAM account creation.
* The provisioning result is logged and communicated back to ServiceNow.

This process ensures automated and secure provisioning of privileged accounts while adhering to organizational policies and role-based access control.