Saviynt

Saviynt Identity Cloud Administration Guide

Copyright

saviynt.com

CONTENTS

Managing Actionable Analytics Controls

This section describes the allowed actions that can be used when creating a report.

Accept

When Accept is configured as an allowed action, it accepts the record as is and continues with non-conformance. You can do a careful evaluation later to check if the record is not a risk.

While accepting a record, provide a start date and an end date signifying the acceptance duration. Once the end date of the record is reached, the record's acceptance period completes and it reaches 'Open' state, indicating that you can take action on the record again.

Revoke

When Revoke is configured as an allowed action, it revokes the record. The record is a serious one, may cause a risk, and can become a drastic threat.

For example, a financial misuse by a certain employee, may result in revoking an employee's access to the financial application. Revoking the record needs a manual effort to revoke the access.

Further Review

When Further Review is configured as an allowed action, it allows you to assign the record to another person for review. You can also assign the record to multiple persons. The final assignee can review and take the configured action.

Note that the assignee must have access to the required controls through the assigned SAV Role.

> **Note**
>
> Action, Revoke and Further Review does not create any tasks.

Provision Access

When Provision Access is configured as an allowed action, it allows you to assign the account to a particular access. When this action is performed on a record, an Add Access task is created in the target application.

The Analytics query must have the columns given below:

`entvaluekey` - Entitlement value key of the entitlement to which the account will be assigned.

`acctKey` - Accountkey of the account to provision with the entitlement.

`accName` - Name of the account to provision with the entitlement.

`userKey` - Userkey of the user to whom the account is mapped.

Sample query:

SQL

```
select ev.entitlement_valuekey as entvaluekey, a.accountkey as acctKey, a.name as accName,u.userKey from entitlement_values ev, entitlement_types et, accounts a, user_accounts uwhere ev.entitlementtypekey=et.entitlementtypekey and et.endpointkey=6 and a.accountkey in(select accountkey from accounts where endpointkey=6) and u.ACCOUNTKEY=a.ACCOUNTKEY;
```

This action can also be configured as a default action, i.e., it will be automatically performed when the Analytics control is executed. Note that this default action is applicable when the Analytics control is scheduled for execution from the **Admin** > **Job Control Panel** > **Analytics jobs**.

Deprovision Access

When Deprovision Access is configured as an allowed action, it removes the entitlement from a particular account. When this action is performed on a record, a Remove Access task is created for removing access in the target application.

The Analytics query must have the columns given below:
`entvaluekey`- Entitlement value key of the entitlement which will be removed from the account.
`acctKey` - Accountkey of the account from which the entitlement access will be revoked.

Sample query:

- Schedule from Run All V2 Analytics Job

  SQL

```
select ae1.entitlement_valuekey as entvaluekey,ev.ENTITLEMENT_VALUE,ae1.accountkey as acctKey,
 a.name, 'Deprovision Access' as 'Default_Action_For_Analytics' from account_entitlements1 ae
1, accounts a , entitlement_values ev where a.accountkey = ae1.accountkey and ev.ENTITLEMENT_V
ALUEKEY=ae1.ENTITLEMENT_VALUEKEY and a.endpointkey=12;
```

- Schedule from Run All V1 Analytics Job

  SQL

```
select ae1.entitlement_valuekey as entvaluekey,ev.ENTITLEMENT_VALUE,ae1.accountkey as acctKey,
 a.name, 'Deprovision Access' as 'Default_Action_For_Analytics' from account_entitlements1 ae
1, accounts a , entitlement_values ev where a.accountkey = ae1.accountkey and ev.ENTITLEMENT_V
ALUEKEY=ae1.ENTITLEMENT_VALUEKEY and a.endpointkey=12;
```

This action can also be configured as a default action, i.e., it will be automatically performed when the Analytics control is executed.

Provision Roles

When the Provision Role is configured as an allowed action, the user role in EIC is added. This is applicable for enterprise, application and emergency access roles. When this action is performed on a record, a task is created to provide user access to the entitlements associated with the roles.

The Analytics query must have the columns given below:
`userkey` - Userkey of the user to whom the account is mapped.
`roleKey` - Rolekey of the role from which the entitlement access will be revoked.

Sample query:

SQL

```
select r.ROLE_NAME,u.username as name, r.ROLEKEY as rolekey, u.USERKEY as userkey,'Provision Rol
e' as 'Default_Action_For_Analytics' from roles r, users u where userkey =0000 and rolekey in (00
000)
```

Note

The columns names **rolekey** and **userkey** are case sensitive.

Deprovision Roles

When Deprovision Role is configured as an allowed action, it removes the role from user profile in Saviynt Identity Cloud. This is applicable for enterprise, application and emergency access roles. When this action is performed on a record, a task is created for removing user access for the entitlements associated with the roles.

The Analytics query must have the columns given below:

`userkey` - Userkey of the user to whom the account is mapped.

`acctKey` - Accountkey of the account from which the entitlement access will be revoked.

`roleKey` - Rolekey of the role from which the entitlement access will be revoked.

Sample query:

SQL

```
select r.ROLE_NAME,u.username as name, rua.ROLEKEY as roleKey,rua.ACCOUNTKEY as acctKey, u.USERKEY as userKey,'Deprovision Role' as 'Default_Action_For_Analytics' from role_user_account rua join user_accounts ua on ua.ACCOUNTKEY=rua.ACCOUNTKEY join users u on u.USERKEY = ua.USERKEY join roles r on r.ROLEKEY = rua.ROLEKEY where r.STATUS=1 ;
```

Password Expired

When Password Expired is configured as an allowed action, it expires user's password on the configured date. You can manually set or

schedule the password expiry. This action can be configured as a default action. This action does not create any tasks, instead it marks the password to be expired for the selected users.

The Analytics query must have the columns given below:
`userKey` - Stores the userkey of the user whose password has to be expired.

Sample query:

- Manual action (bulk/single row action)

  SQL

  ```
  select u.userkey, u.username , u.LASTPASSWORDUPDATEDATE, date_add(u.LASTPASSWORDUPDATEDATE,Interval p.expireafter day) as 'Expiry Date' from users u, policyrule p
  ```

  where `u.PASSWORDEXPIRED`=0
  and `p.SCOPE`='USER'
  and `p.EXPIREAFTER` > 0
  and `DATEDIFF(date_add(u.LASTPASSWORDUPDATEDATE,Interval p.expireafter day),sysdate())` = 0;

- Schedule from Run All V2 Analytics Job

  SQL

```
select u.userkey, u.username , u.LASTPASSWORDUPDATEDATE, date_add(u.LASTPASSWORDUPDATEDATE,Int
erval p.expireafter day) as 'Expiry Date', 'passwordExpired' as 'Default_Action_For_Analytic
s' from users u, policyrule p
```

where `u.PASSWORDEXPIRED`=0

and `p.SCOPE`='USER'and p.EXPIREAFTER > 0

and `DATEDIFF(date_add(u.LASTPASSWORDUPDATEDATE,Interval p.expireafter day),sysdate())` = 0;

- Schedule from Run All V1 Analytics Job

  SQL

```
select u.userkey, u.username , u.LASTPASSWORDUPDATEDATE, date_add(u.LASTPASSWORDUPDATEDATE,Int
erval p.expireafter day) as 'Expiry Date', 'Password Expired' as 'Default_Action_For_Analytic
s' from users u, policyrule p
```

  where `u.PASSWORDEXPIRED`=0

  and `p.SCOPE`='USER'and p.EXPIREAFTER > 0

  and `DATEDIFF(date_add(u.LASTPASSWORDUPDATEDATE,Interval p.expireafter day),sysdate())` = 0;

Deprovision Account

When Deprovision Account is configured as an allowed action, it deprovisions account of a user from Saviynt Identity Cloud. When this

action is performed on a record, a task is created for deprovisioning the account in the target application.

For example, a Box connector account of a user who is no longer with the organization is assigned to an admin or a service account.

The Analytics query must have the columns given below:
`acctKey` - Stores accountkey of the account which has to be deprovisioned.

Sample query:

- Schedule from Run All V2 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'deprovisionAccount' as Default_Action_For_An
alytics from accounts;
```

- Schedule from Run All V1 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'Deprovision Account' as Default_Action_For_A
nalytics from accounts;
```

Enable Account

When Enable Account is configured as an allowed action, it enables a user account that was disabled for security reasons. When this action is performed on a record, an Enable Account task is created for enabling the account in the target application.

The Analytics query must have the columns given below:
`acctKey` - Stores accountkey of the account which has to be enabled.

Sample query:

- Schedule from Run All V2 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'enableAccount' as Default_Action_For_Analyti
cs from accounts a;
```

- Schedule from Run All V1 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'Enable Account' as Default_Action_For_Analyt
ics from accounts a;
```

Disable Account

When Disable Account is configured as an allowed action, it disables a user account for security reasons. When this action is performed on a record, a Disable Account task is created for disabling the account in the target application.

The Analytics query must have the columns given below:
`acctKey` - Stores accountkey of the account which has to be disabled.

Sample query:

- Schedule from Run All V2 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'disableAccount' as Default_Action_For_Analyt
ics from accounts a;
```

- Schedule from Run All V1 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'Disable Account' as Default_Action_For_Analy
tics from accounts a;
```

Map Orphan Account

When Map Orphan Account is configured as an allowed action, it maps users to orphan accounts within Saviynt Identity Cloud.

> **Note**
>
> Default actions are not supported by the query. They must be performed manually.

For example, you can map an object in ExchangeOnline that does not have a corresponding object in Azure AD.

The Analytics query must have the columns given below:
`acctKey` - Stores account key of the orphan account which needs to be mapped to user account.

Sample query:

SQL

```
SELECT
  DISTINCT a.accountkey as acctKey,
  a.name 'Account Name',
  e.displayname 'Application Name',
  CASE WHEN a.status = 1 THEN 'Active' ELSE a.status END AS 'Account Status'
FROM
```

```
  accounts a,
  endpoints e,
  securitysystems s
WHERE
  a.endpointkey = e.endpointkey
  AND e.securitysystemkey = s.systemkey
  AND a.status IN (
    1, 'Active', 'Manually Provisioned'
  )
  and e.ACCOUNTTYPEFORSERVICEACCOUNT not like concat('%', a.ACCOUNTTYPE, '%')
  and a.accountkey not in (
    SELECT
      accountkey
    FROM
      user_accounts ua
  )
ORDER BY
  s.systemname;
```

For more information about viewing mapped orphan accounts, see *Sample Actions* in Reports in Creating Actionable Analytic Reports .

Transfer Data

When Transfer Data is configured as an allowed action, it transfers access from one account to another. This action is available only for Box application accounts. Add the source account key in the analytics query and select the destination account while taking action. This action cannot be performed by the default action in Analytics Job. You must take this action manually.

The Analytics query must have the columns given below:
`acctKey` - Stores account key of the source account from which the access is to be transferred to the destination account.

Sample query:

- Schedule from Run All V2 Analytics Job

SQL

```
select NAME, ACCOUNTKEY as acctKey, endpointkey from accounts where status in(2,'inactive','Manua
lly Suspended');
```

- Schedule from Run All V1 Analytics Job

SQL

```
select NAME, ACCOUNTKEY as acctKey, endpointkey from accounts where status in(2,'inactive','Manua
lly Suspended');
```

Reopen Tasks

When Reopen Tasks is configured as an allowed action, it re-opens a closed task for re-execution, debugging or any other analysis. When you have exhausted the maximum number of retries for a task and the associated task is still not successfully processed, use this action to re-process those failed tasks.

The Analytics query must have the columns given below:
`tasks` - Stores taskkey of the task to be re-opened.

Sample query:

- Run all V2

SQL

```
select taskkey as tasks, STATUS , provisioningtries, 'reopenTasks' as 'Default_Action_For_Analytics' from arstasks where status=4 and PROVISIONINGTRIES>1;
```

- Run all V1

SQL

```
select taskkey as tasks, STATUS , provisioningtries, 'Reopen Tasks' as 'Default_Action_For_Analytics' from arstasks where status=4 and PROVISIONINGTRIES>1;
```

Update Account

When Update Account is configured as an allowed action, it updates an account in Saviynt Identity Cloud. When this action is performed on a record, an Update Account task is created for updating the account in the target application.

The Analytics query must have the columns given below:
`acctKey` - Stores accountkey of the account which has to be updated.

Sample query:

- Schedule from Run All V2 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'UpdateAccount' as Default_Action_For_Analyti
cs from accounts a;
```

- Schedule from Run All V1 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'Update Account' as Default_Action_For_Analyt
ics from accounts a;
```

Delete Account

When Delete Account is configured as an allowed action, it deletes an account from Saviynt Identity Cloud. When this action is performed on a record, a Delete Account task is created for deleting the account in the target application.

The Analytics query must have the columns given below:
`acctKey` - Stores accountkey of the account which has to be deleted.

Sample query:

- Schedule from Run All V2 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'deleteAccount' as Default_Action_For_Analyti
cs from accounts a;
```

- Schedule from Run All V1 Analytics Job

SQL

```
select a.name,a.accountkey as acctKey, endpointkey, 'Delete Account' as Default_Action_For_Analyt
ics from accounts a;
```

Map Entitlement Owner

When Map Entitlement Owner is configured as an allowed action, it maps owner(s) to entitlements in Saviynt Identity Cloud.

The Analytics query must have the columns given below:
`entvaluekey` - Stores entitlement key of the entitlement to which the owner needs to be added.

Sample query:

SQL

```
select e.endpointname AS APPLICATION, et.ENTITLEMENTNAME as 'ENTITLEMENT TYPE', ev.entitlement_value AS 'ENTITLEMENT NAME', ev.ENTITLEMENT_VALUEKEY AS 'entvaluekey' from entitlement_values ev, entitlement_owners eo, endpoints e, entitlement_types et where ev.ENTITLEMENTTYPEKEY= et.ENTITLEMENTTYPEKEY and et.ENDPOINTKEY = e.ENDPOINTKEY and ev.ENTITLEMENT_VALUEKEY not in (Select distinct ENTITLEMENT_VALUEKEY from entitlement_owners);
```

For more information about viewing mapped entotlement owners, see *Sample Actions* in Reports in Creating Actionable Analytic Reports.

Note

The default action via query is not available and these actions have to be performed manually.

Map Account Owners

You can map owners to your existing standard or service accounts via a new action named **Map Account Owner**. The accounts can be mapped to individual users or user groups. While mapping, specify the rank to which the owners (selected user or a user group) belongs to.

> **Note**
>
> Default actions are not supported by the query. They must be performed manually.

While creating or updating a control, you must specify the following column for both standard and service accounts.

- `acctKey` - This column stores the account keys that are used for mapping accounts with their owners.

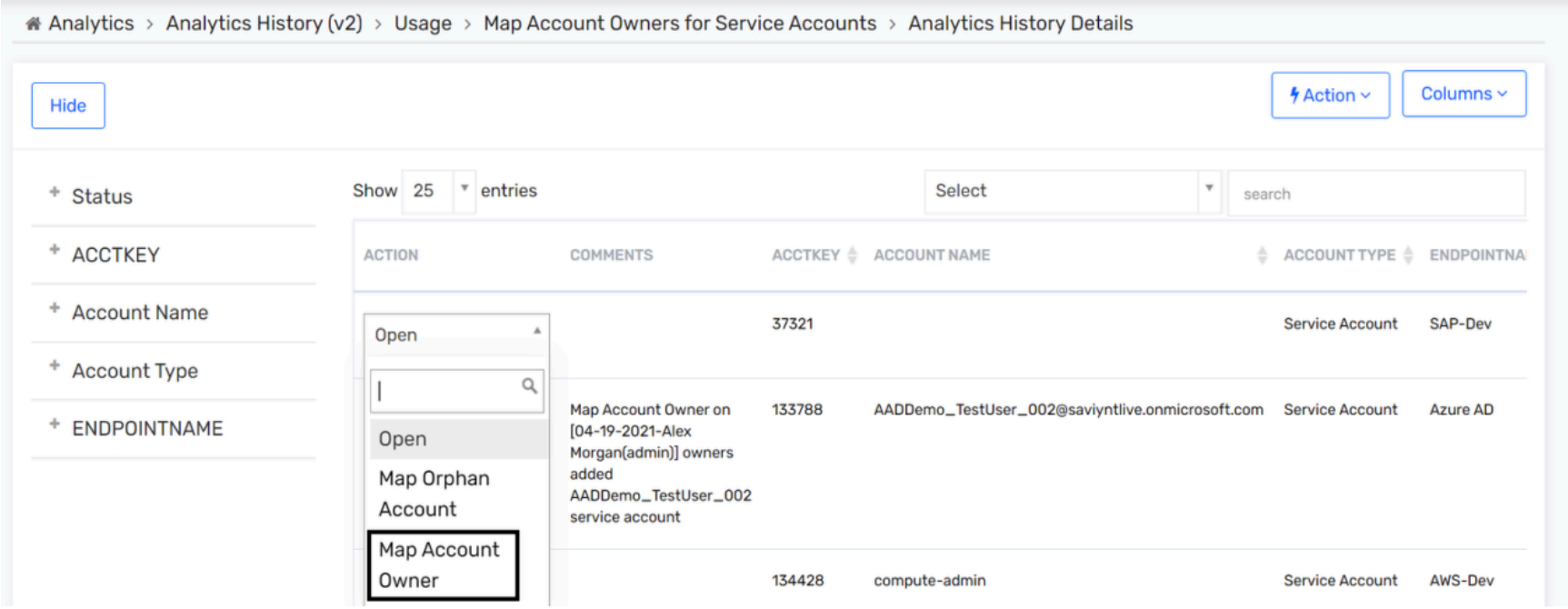
SQL

```
select a.accountkey as acctKey, a.name as 'AccountName' , a.ACCOUNTTYPE ,e.endpointname as 'Endpo
intName'
from accounts a, endpoints e
where
a.ENDPOINTKEY=e.ENDPOINTKEY
and a.ACCOUNTKEY not in (select distinct ACCOUNTKEY from accountowners)
```

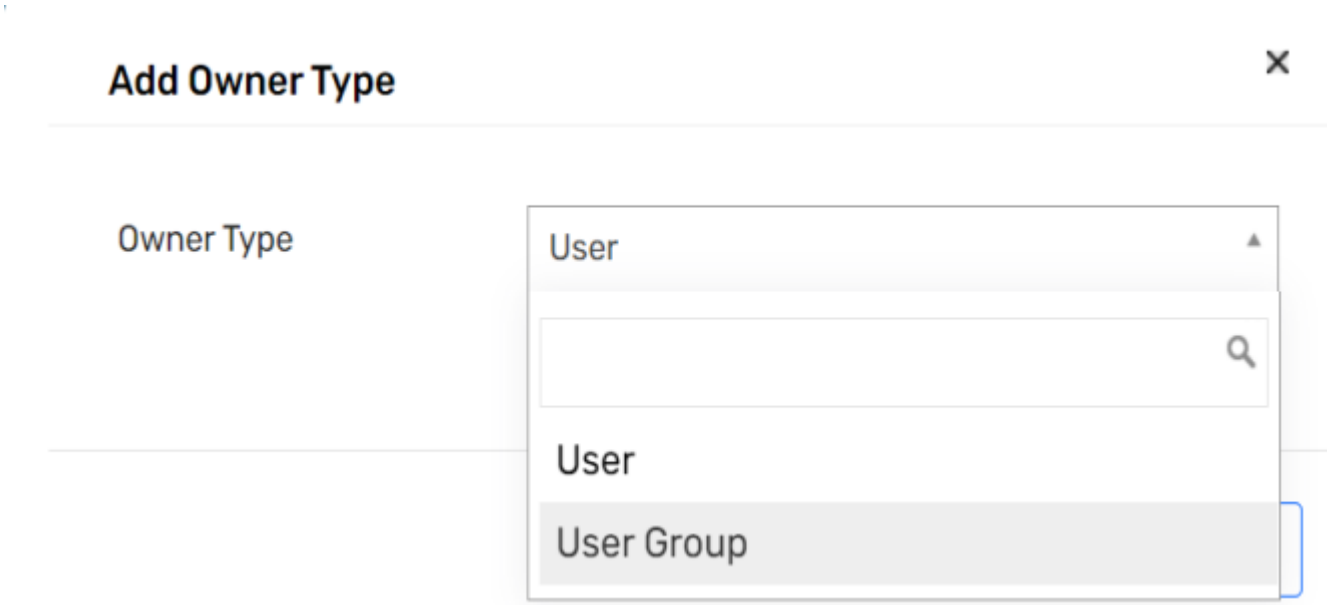
```
and a.ACCOUNTTYPE = 'Service Account';
```

To map an account owner, perform the following steps:

1. In the **Analytics History Details** page, navigate to the **Action** drop down against an account that has to be mapped to an owner, and click **Map Account Owner**.

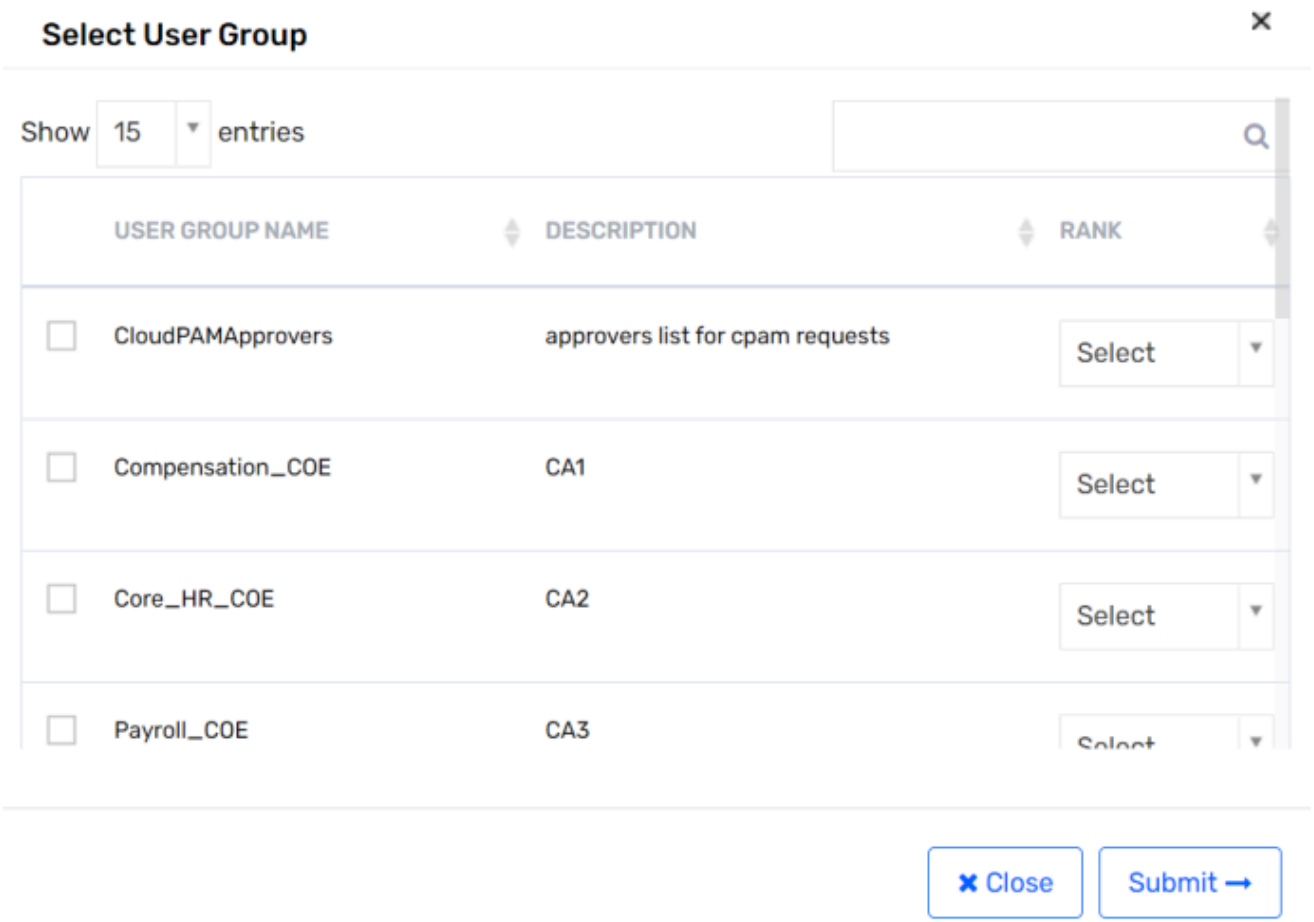


a. In the **Add Owner Type** window, select the owner type as **User** or **User Group**, and click **Next**.

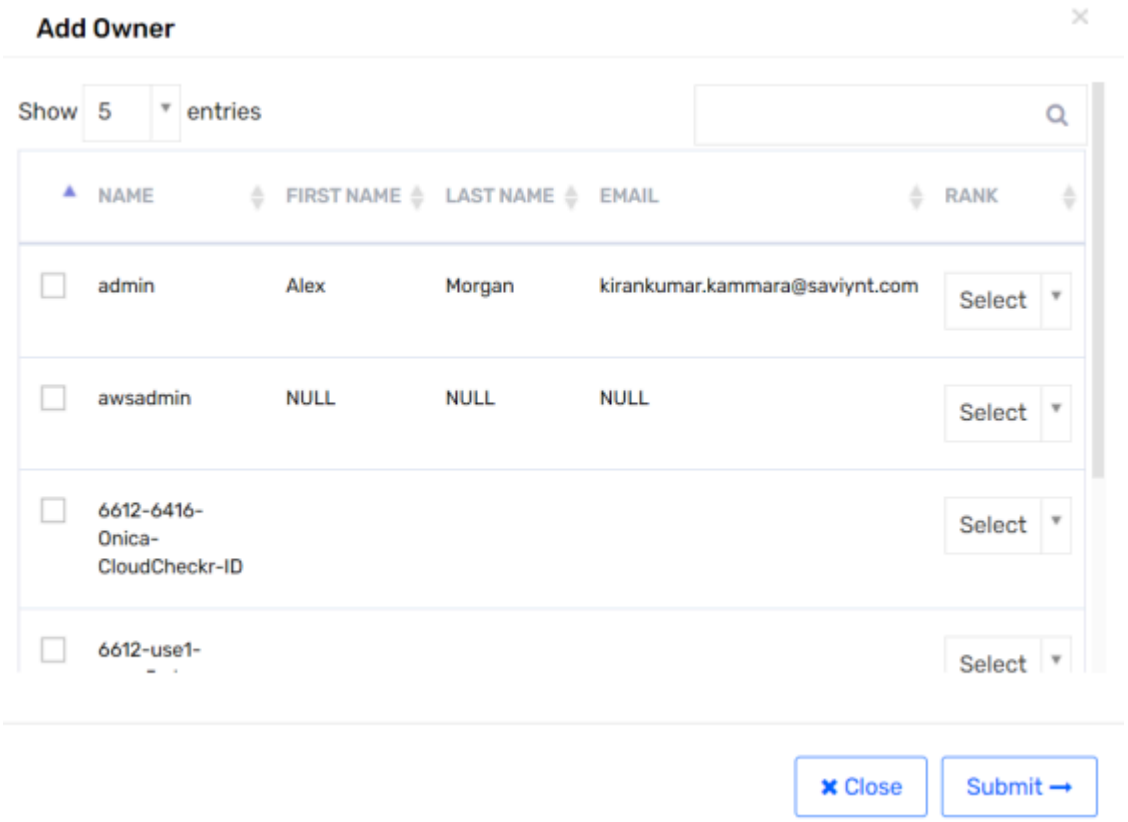


1. Select **User Group** if you want to assign multiple users as the owners of the analytics control. A user group consists of single or multiple users and provides a convenient way to assign multiple users as owners of analytics controls.

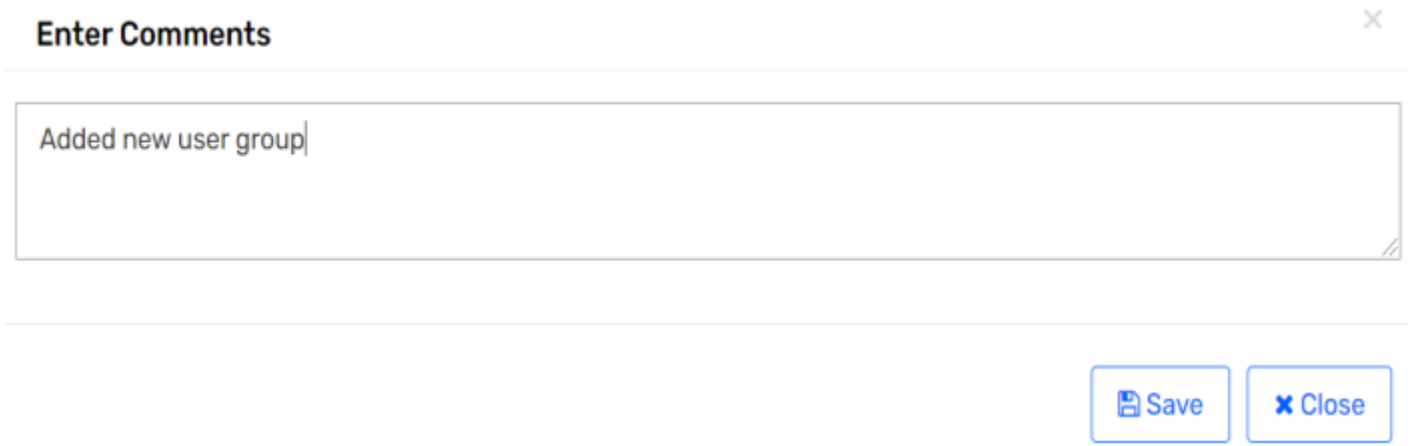
    1. If you select **User Group**, select one or more user groups and their ranks from the **Select User Group** page and click **Submit**.



2. If you select **User**, select a user and the user's rank from the **Add Owner** page and click **Submit**.



2. In the **Enter Comments** window, specify the comments and click **Save**.



3. Navigate to **Admin > Identity Repository > Accounts > Owner** to view the newly added owner details.

Note

Default actions are not supported via queries. You need to perform them manually.

Quarantine File

When Quarantine File is configured as an allowed action, it moves the file with sensitive data (for example, driving license number, social security number, etc) from its original location to the Quarantine folder.

A dummy file is created in the original location stating that the file was moved since it contained sensitive content. Entitlement

Customproperty should have a value to denote that the file has been quarantined. Original file location is maintained in the entitlements_value table.

Sample query:

SQL

```
SELECT DISTINCTexc.CONNECTIONNAME AS 'GApps Account',ev.customproperty9 AS 'File Name',ev.CUSTOMP
ROPERTY5 AS 'File Path',ev.customproperty4 AS 'File Owner',GROUP_CONCAT(DISTINCT sr.RulenameORDER
 BY sr.Rulename ASCSEPARATOR ',') AS 'Rule Name',sre.times_matched AS 'Violations Count',CASEWHEN
 (ev.CUSTOMPROPERTY11 = 'anyone') THEN 'External'WHEN (ev.CUSTOMPROPERTY11 = 'domain') THEN 'Inte
rnal'WHEN (ev.CUSTOMPROPERTY11 = 'group') THEN 'Group'WHEN(ev.CUSTOMPROPERTY11 = 'Specific Peopl
e'AND ac.ACCOUNTTYPE = 'Internal')THEN'Internal Collaborator'WHEN(ev.CUSTOMPROPERTY11 = 'Specific
 People'AND ac.ACCOUNTTYPE = 'External')THEN'External Collaborator'ELSE 'Other shared type'END A
S 'Share Type',CASEWHEN(ev.CUSTOMPROPERTY11 = 'anyone'AND sre.times_matched >= 1)THEN'Critical -
Publicly Shared'WHEN((ev.CUSTOMPROPERTY11 = 'group'OR ev.CUSTOMPROPERTY11 = 'domain'OR ev.CUSTOMP
ROPERTY11 = 'Specific People')AND (sre.times_matched >= 1AND sre.times_matched <= 999))THEN'Low'W
HEN((ev.CUSTOMPROPERTY11 = 'domain'OR ev.CUSTOMPROPERTY11 = 'group'OR ev.CUSTOMPROPERTY11 = 'Spec
ific People')AND (sre.times_matched >= 1000AND sre.times_matched <= 4999))THEN'Medium'WHEN((ev.CU
STOMPROPERTY11 = 'domain'OR ev.CUSTOMPROPERTY11 = 'group'OR ev.CUSTOMPROPERTY11 = 'Specific Peopl
e')AND (sre.times_matched >= 5000AND sre.times_matched <= 9999))THEN'High'WHEN((ev.CUSTOMPROPERTY
11 = 'domain'OR ev.CUSTOMPROPERTY11 = 'group'OR ev.CUSTOMPROPERTY11 = 'Specific People')AND (sr
e.times_matched >= 10000))THEN'Critical'ELSE 'Undefined'END AS Severity,ev.entitlement_valuekey A
```

```
S entvaluekey,exc.externalconnectionkey AS Connectionkey,ep.ENDPOINTKEY AS endpointKeyFROMentitle
ment_values evINNER JOINentitlement_types et ON ev.entitlementtypekey = et.entitlementtypekeyAND
ev.Status = 1AND et.ENTITLEMENTNAME = 'GoogleFile'INNER JOINscanrules_entitlements sre ON sre.ENT
ITLEMENT_VALUEKEY = ev.ENTITLEMENT_VALUEKEYAND ev.CustomProperty18 IS NULLAND sre.VIOLATION_PROBA
BILITY = 'HIGH'AND ev.customproperty11 IS NOT NULLINNER JOINscanrules sr ON sre.SCANRULENAME = s
r.SCANRULEKEYAND sr.rulename LIKE '%License%'INNER JOINsecuritysystems ss ON et.systemkey = ss.sy
stemkeyINNER JOINendpoints ep ON ep.SECURITYSYSTEMKEY = ss.systemkeyINNER JOINexternalconnection
exc ON ss.externalconnection = exc.externalconnectionkeyINNER JOINexternalconnectiontype exct ON
exc.externalconnectiontype = exct.externalconnectiontypekeyAND exct.connectiontype = 'GoogleApp
s'INNER JOINaccounts ac ON ac.name = ev.customproperty4AND ac.status = 1AND ac.endpointkey IN (2
3 , 26, 27)
```

Unquarantine File

When Unquarantine File is configured as an allowed action, it moves the file from the quarantined folder to the original location.

If the file does not have sensitive data and has been incorrectly moved to the quarantine folder, delete the dummy file created during the quarantine operation and reset the value of Customproperty that denotes the file is quarantined.

Delete File

When Delete File is configured as an allowed action, it deletes the file from the Box application.

False Positive

When False Positive is configured as an allowed action, it marks a file as False Positive, if a file is identified as sensitive content during a scan but its actually not is marked as False-positive.

Certain files may be identified as containing sensitive data (such as driving license number, Social security number (SSN), etc) however, on further review it appears that the file does not have such data. Files marked as False Positive are not scanned by Saviynt Identity Cloud.

Sample query:

SQL

```
SELECT DISTINCT ev.entitlement_value AS 'Sensitive File Name', ev.customproperty9 AS 'File Name',
 ev.CUSTOMPROPERTY5 AS 'File Path', ev.customproperty4 AS 'File Owner', ev.entitlement_valuekey A
S entvaluekey, exc.externalconnectionkey AS Connectionkey FROM entitlement_values ev INNER JOIN e
ntitlement_types et ON ev.entitlementtypekey = et.entitlementtypekey AND ev.Status = 1 AND et.ENT
ITLEMENTNAME = 'GoogleFile' and ev.customproperty18 = 2 INNER JOIN scanrules_entitlements sre ON
sre.entitlement_valuekey = ev.entitlement_valuekey INNER JOIN scanrules sr ON sr.scanrulekey = sr
e.scanrulename INNER JOIN securitysystems ss ON et.systemkey = ss.systemkey INNER JOIN endpoints
ep ON ep.SECURITYSYSTEMKEY = ss.systemkey INNER JOIN externalconnection exc ON ss.externalconnect
ion = exc.externalconnectionkey INNER JOIN externalconnectiontype exct ON exc.externalconnectiont
ype = exct.externalconnectiontypekey AND exct.connectiontype = 'GoogleApps'
```

Non-False Positive

When Non-False Positive is configured as an allowed action, it marks a file as Non-False Positive, if a file marked as False Positive is rolled back and marked as Non-False Positive.

Certain files may be identified as containing sensitive data when the file is updated. Files marked as Non-False Positive are now scanned by Saviynt Identity Cloud.

Remove All Collaborators Access

When Remove All Collaborators Access is configured as an allowed action, it removes access of all the collaborators on the file.

White List

When White List is configured as an allowed action, it marks a file as whitelisted, if files/folders contains sensitive data but updated by a legitimate genuine user. Files marked as whitelisted are not scanned by Saviynt Identity Cloud.

For example, a sensitive file with the bank account name and salary details of employees is crucial, however, if uploaded by a finance CFO working in the organization is permissible and can be marked as a whitelisted.

Sample query:

SQL

```
SELECT DISTINCTexc.CONNECTIONNAME AS 'GApps Account',ev.entitlement_value AS 'Sensitive FileNam
e',ev.customproperty9 AS 'GApps File Name',ev.CUSTOMPROPERTY5 AS 'GApps File Path',ev.customprope
rty4 AS 'File Owner',GROUP_CONCAT(DISTINCT sr.RulenameORDER BY sr.Rulename ASCSEPARATOR ', ') AS
'Rule Name',sr.category AS 'ViolationType',ev.entitlement_valuekey AS entvaluekey,exc.externalcon
nectionkey AS Connectionkey,ep.ENDPOINTKEY AS endpointKeyFROMentitlement_values evINNER JOINentit
lement_types et ON ev.entitlementtypekey = et.entitlementtypekeyAND ev.Status = 1AND et.ENTITLEME
NTNAME = 'GoogleFile'AND ev.Status = 1AND ev.CUSTOMPROPERTY18 = 1INNER JOINscanrules_entitlements
 sre ON sre.entitlement_valuekey = ev.entitlement_valuekeyINNER JOINscanrules sr ON sr.scanruleke
y = sre.scanrulenameAND sr.category IN ('PCI' OR 'PII')INNER JOINsecuritysystems ss ON et.systemk
ey = ss.systemkeyINNER JOINendpoints ep ON ep.SECURITYSYSTEMKEY = ss.systemkeyINNER JOINexternalc
onnection exc ON ss.externalconnection = exc.externalconnectionkeyINNER JOINexternalconnectiontyp
e exct ON exc.externalconnectiontype = exct.externalconnectiontypekeyAND exct.connectiontype = 'G
oogleApps'
```

Un WhiteList

When Un WhiteList is configured as an allowed action, it removes the whitelist flag on files or folders. Files marked as whitelisted are now scanned by Saviynt Identity Cloud.

For example, a file previously whitelisted is updated and uploaded by someone other than legitimate user.

Stop EC2 Instance

When Stop EC2 Instance is configured as an allowed action, it stops the EC2 instance in the target AWS account.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name' , Ec2.ent
itlement_value as 'EC2 Instance',evat1.attribute_Value as 'Name',Ec2.customproperty9 as 'Regio
n',If(ec2.customproperty14 = 'Windows', 'Windows','Linux') as 'Platform',if(Ec2.customproperty12
is NULL,'Not Assigned',Ec2.customproperty12) as 'Public IP',scg.entitlement_value as 'Security Gr
oup', 'Inbound' as 'Rule', ev.customproperty1 as 'Source' ,ev.customproperty2 as 'Protocol',ev.cu
stomproperty3 as 'From Port',ev.customproperty4 as 'To Port',Ec2.customproperty3 as 'Tags',Ec2.EN
TITLEMENT_VALUEKEY as entvaluekey ,et.endpointkey as externalConnectionKey ,ep.ENDPOINTKEY AS end
pointKey from entitlement_values ev INNER JOIN entitlement_types et ON ev.entitlementtypekey = e
t.entitlementtypekey And et.Entitlementname = 'SGIBRules' INNER JOIN securitysystems sc ON et.sys
temkey = sc.systemkey INNER JOIN endpoints ep on ep.SECURITYSYSTEMKEY=sc.systemkey INNER JOIN ext
ernalconnection exc ON sc.externalconnection = exc.externalconnectionkey INNER JOIN externalconne
ctiontype exct ON exc.externalconnectiontype = exct.externalconnectiontypekey and exct.connection
type = 'AWS' INNER JOIN externalconnattvalue excv ON excv.connectiontype = exc.externalconnection
key and excv.attributekey = 'AWS_ACCOUNT_ID' INNER JOIN entitlements2 e2 ON e2.entitlement_value2
```

```
key = ev.entitlement_valuekey INNER JOIN entitlement_values scg ON e2.entitlement_value1key = sc
g.entitlement_valuekey INNER JOIN entitlement_types et1 ON et1.entitlementtypekey = scg.entitleme
nttypekey and et1.entitlementname = 'AWSSecurityGroup' LEFT OUTER JOIN entitlementmap emap ON e
2.entitlement_value1key = emap.entitlement_value2key INNER JOIN entitlement_values Ec2 ON emap.en
titlement_value1key = Ec2.entitlement_valuekey INNER JOIN entitlement_types et2 ON et2.entitlemen
ttypekey = Ec2.entitlementtypekey and et2.entitlementname = 'EC2Instance' Left Outer Join entitle
ment_value_attrs evat1 on evat1.ENTITLEMENT_VALUE_KEY = ec2.entitlement_Valuekey and evat1.ATTRIB
UTE_NAME = 'Name' Where ev.customproperty1 = '0.0.0.0/0' and ev.customproperty2 not in ('icmp','u
dp') and ((CAST(ev.customproperty3 AS UNSIGNED) <= 3389 and CAST(ev.customproperty4 AS UNSIGNED)
>= 3389) or ((ev.customproperty3 = 'ALL') and (ev.customproperty4 = 'ALL'))) and ev.status = 1 an
d COALESCE(scg.status,0) < 2 and ec2.status = 1 and Ec2.entitlement_value != 'null' and ev.custom
property1 not like '%terminated%'
```

Enable Cloud Trail

When Enable Cloud Trail is configured as an allowed action, it enables the log file integrity validation for various cloud trails in AWS.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name',ev.entitl
```

```
ement_value as 'Trail Name', ev1.entitlement_value as 'Bucket Name', ev1.customproperty8 as 'Crea
tion Time',ev1.customproperty3 as 'Tags',et.endpointkey as externalConnectionKey ,ep.ENDPOINTKEY
AS endpointKey,ev.entitlement_valuekey AS entvaluekey from entitlement_values ev Inner Join entit
lement_types et on ev.entitlementtypekey = et.entitlementtypekey and et.Entitlementname = 'AWSClo
udTrail' and ev.status = 1 inner Join entitlement_values ev1 inner join entitlement_types et1 on
ev1.entitlementtypekey = et1.entitlementtypekey and et1.entitlementname = 's3bucket' and ev.CUSTO
MPROPERTY1 = ev1.entitlement_value and ev1.status = 1 inner join securitysystems sc on et.systemk
ey = sc.systemkey INNER JOIN endpoints ep on ep.SECURITYSYSTEMKEY=sc.systemkey INNER JOIN externa
lconnection exc on exc.externalconnectionkey = sc.externalconnection INNER JOIN externalconnattva
lue excv on excv.connectiontype = exc.externalconnectionkey and excv.attributekey = 'AWS_ACCOUN
T_ID' INNER JOIN entitlements2 e2 on e2.entitlement_value1key = ev1.Entitlement_valuekey INNER JO
IN entitlement_values ev2 on ev2.Entitlement_valuekey = e2.entitlement_value2key and ev2.Entitlem
ent_value = 'AllUsers' INNER JOIN entitlements2_privilege e2p on e2p.entitlement2uid = e2.entitle
ment2uid
```

Enable Rotation

When Enable Rotation is configured as an allowed action, it enables rotation of the KMS Key for specific AWS region(s).

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name',ep.displa
yname as 'Account Name' ,ev.entitlement_value as 'Volume ID', ev.customproperty11 as 'Encrypte
d',ev.customproperty10 as 'Region',ev.customProperty2 as 'Creation Time',ev.customProperty1 as 'S
ize (In GB)',ev.customProperty13 as 'State',ev.customProperty16 as 'Attachment Device',ev1.custom
property13 as 'Key Alias',ev1.customProperty11 as 'Key ARN',ev.customProperty3 as 'Tags',ev.ENTIT
LEMENT_VALUEKEY as entvaluekey ,et.endpointkey as externalConnectionKey ,ep.ENDPOINTKEY AS endpoi
ntKey from entitlement_values ev Inner Join entitlement_types et on ev.entitlementtypekey = et.en
titlementtypekey and et.Entitlementname = 'EBSVolume' and ev.status = 1 inner join entitlement_va
lues ev1 on ev.customproperty12=ev1.customproperty11 and IFNULL(ev1.customproperty13, '') LIKE 'a
ws/%' Inner Join entitlement_types et1 on ev1.entitlementtypekey = et1.entitlementtypekey and et
1.Entitlementname = 'encryptionkeys' and ev1.status = 1 inner join securitysystems sc on et.syste
mkey = sc.systemkey INNER JOIN endpoints ep on ep.SECURITYSYSTEMKEY=sc.systemkey INNER JOIN exter
nalconnection exc on exc.externalconnectionkey = sc.externalconnection INNER JOIN externalconnatt
value excv on excv.connectiontype = exc.externalconnectionkey and excv.attributekey = 'AWS_ACCOUN
T_ID'
```

Detach Volumes

When Detach Volumes is configured as an allowed action, it detaches explicitly or terminates an Amazon EBS volume from an instance. But, if the instance is running, you must first unmount the volume from the instance.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name' , ev.enti
tlement_value as 'Volume ID', ev.customproperty13 as 'State',ev.customproperty10 as 'Region',ev.c
ustomProperty1 as 'Size (In GB)', ev.customProperty2 as 'Creation Time',ev.ENTITLEMENT_VALUEKEY a
s entvaluekey ,et.endpointkey as externalConnectionKey ,ep.ENDPOINTKEY AS endpointKey from entitl
ement_values ev Inner Join entitlement_types et on ev.entitlementtypekey = et.entitlementtypekey
and et.Entitlementname = 'EBSVolume' and ev.status = 1 and ev.customproperty13 not like '%In-us
e%' inner join securitysystems sc on et.systemkey = sc.systemkey INNER JOIN endpoints ep on ep.SE
CURITYSYSTEMKEY=sc.systemkey INNER JOIN externalconnection exc on exc.externalconnectionkey = s
c.externalconnection INNER JOIN externalconnattvalue excv on excv.connectiontype = exc.externalco
nnectionkey and excv.attributekey = 'AWS_ACCOUNT_ID'
```

Delete Security Group

When Delete Security Group is configured as an allowed action, it deletes a security group. If the security group is associated with an AWS instance, an error message is displayed.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name' , Ec2.ent
itlement_value as 'EC2 Instance',evat1.attribute_Value as 'Name',Ec2.customproperty9 as 'Regio
n',If(ec2.customproperty14 = 'Windows', 'Windows','Linux') as 'Platform',if(Ec2.customproperty12
is NULL,'Not Assigned',Ec2.customproperty12) as 'Public IP',scg.entitlement_value as 'Security Gr
oup', 'Inbound' as 'Rule', ev.customproperty1 as 'Source' ,ev.customproperty2 as 'Protocol',ev.cu
stomproperty3 as 'From Port',ev.customproperty4 as 'To Port',Ec2.customproperty3 as 'Tags',Ec2.EN
TITLEMENT_VALUEKEY as entvaluekey ,et.endpointkey as externalConnectionKey ,ep.ENDPOINTKEY AS end
pointKey from entitlement_values ev INNER JOIN entitlement_types et ON ev.entitlementtypekey = e
t.entitlementtypekey And et.Entitlementname = 'SGIBRules' INNER JOIN securitysystems sc ON et.sys
temkey = sc.systemkey INNER JOIN endpoints ep on ep.SECURITYSYSTEMKEY=sc.systemkey INNER JOIN ext
ernalconnection exc ON sc.externalconnection = exc.externalconnectionkey INNER JOIN externalconne
ctiontype exct ON exc.externalconnectiontype = exct.externalconnectiontypekey and exct.connection
type = 'AWS' INNER JOIN externalconnattvalue excv ON excv.connectiontype = exc.externalconnection
key and excv.attributekey = 'AWS_ACCOUNT_ID' INNER JOIN entitlements2 e2 ON e2.entitlement_value2
key = ev.entitlement_valuekey INNER JOIN entitlement_values scg ON e2.entitlement_value1key = sc
g.entitlement_valuekey INNER JOIN entitlement_types et1 ON et1.entitlementtypekey = scg.entitleme
nttypekey and et1.entitlementname = 'AWSSecurityGroup' LEFT OUTER JOIN entitlementmap emap ON e
2.entitlement_value1key = emap.entitlement_value2key INNER JOIN entitlement_values Ec2 ON emap.en
titlement_value1key = Ec2.entitlement_valuekey INNER JOIN entitlement_types et2 ON et2.entitlemen
ttypekey = Ec2.entitlementtypekey and et2.entitlementname = 'EC2Instance' Left Outer Join entitle
ment_value_attrs evat1 on evat1.ENTITLEMENT_VALUE_KEY = ec2.entitlement_Valuekey and evat1.ATTRIB
```

```
UTE_NAME = 'Name' Where ev.customproperty1 = '0.0.0.0/0' and (ev.customproperty3 = 'ALL' and ev.c
ustomproperty4 = 'ALL') and ev.status = 1 and COALESCE(scg.status,0) < 2 and ec2.status = 1 and E
c2.entitlement_value != 'null' and ev.customproperty1 not like '%terminated%'
```

Enable EC2 Termination

When Enable EC2 Termination is configured as an allowed action, it enables the EC2 termination protection for an AWS instance. If the AWS instance is already protected, you cannot update the EC2 termination.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name', ev.entit
lement_value as 'EC2 Instance', ev.customproperty9 as 'Region', ev.customproperty5 as 'VPC ID', e
v.customProperty3 as Tags ,et.endpointkey as externalConnectionKey ,ep.ENDPOINTKEY AS endpointKey
 ,ev.entitlement_valuekey AS entvaluekey from entitlement_values ev inner join entitlement_types
et on et.entitlementtypekey = ev.entitlementtypekey And et.entitlementname = 'ec2instance' And e
v.customproperty5 is null And ev.status = 1 and ev.customproperty1 not like '%terminated%' inner
join securitysystems sc on et.systemkey = sc.systemkey inner join endpoints ep on ep.SECURITYSYST
EMKEY=sc.systemkey inner join externalconnection exc on sc.externalconnection = exc.externalconne
ctionkey inner join externalconnectiontype exct on exc.externalconnectiontype = exct.externalconn
```

```
ectiontypekey and exct.connectiontype = 'AWS' inner join externalconnattvalue excv on excv.connec
tiontype = exc.externalconnectionkey and excv.attributekey = 'AWS_ACCOUNT_ID'
```

Delete Elasticsearch IP

When Delete Elasticsearch IP is configured as an allowed action, it deletes an Elastic IP (EIP) from an AWS instance. If you delete an EIP of a running instance, it will remain without that until AWS assigns a new EIP to it.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name', ev.ENTIT
LEMENT_VALUE as 'Domain',ev.customproperty9 as 'Region',ev.customproperty5 as 'Elasticsearch Vers
ion',ev.customproperty7 as 'Number of Nodes',ev.customproperty12 as 'Zone Awareness',ev.custompro
perty3 as 'Tags',ev.ENTITLEMENT_VALUEKEY as entvaluekey ,et.endpointkey as externalConnectionKey
,ep.ENDPOINTKEY AS endpointKey from entitlement_values ev inner join entitlement_types et on et.e
ntitlementtypekey = ev.entitlementtypekey and et.entitlementname = 'Elasticsearch' And ev.status
= 1 and ev.customproperty12 = 'False' INNER JOIN securitysystems sc on et.systemkey = sc.systemke
y inner join endpoints ep on ep.SECURITYSYSTEMKEY=sc.systemkey inner join externalconnection exc
on sc.externalconnection = exc.externalconnectionkey inner join externalconnectiontype exct on ex
c.externalconnectiontype = exct.externalconnectiontypekey and exct.connectiontype = 'AWS' inner j
```

```
oin externalconnattvalue excv on excv.connectiontype = exc.externalconnectionkey and excv.attribu
tekey = 'AWS_ACCOUNT_ID';
```

Delete ELB

When Delete ELB is configured as an allowed action, it deletes an AWS Elastic Load Balancer (ELB) which is not in use.

Sample query:

SQL

```
select ev_elb.entitlement_value as 'ELB', ev_listner.entitlement_value as 'Listener',ev_elb.custo
mproperty9 as 'ELB Region', ev_listner.custom property1 as 'Protocol', substr(ev_listner.custompro
perty5,1,locate(',',ev_listner.custom property5)-1) as 'Certificate', concat(trim(substr(ev_listne
r.custom property5,locate(',',ev_listner.custom property5)+1,20)),substr(ev_listner.custom property
5,(locate(',', ev_listner.custom property5)+24),LENGTH(ev_listner.custom property5)) ) as 'Certific
ate Expiry Date', excv.attributevalue as 'Root Account ID', et_listner.endpointkey as 'Endpoint K
ey' ,ep.ENDPOINTKEY AS endpointKey, ev.entitlement_valuekey AS entvaluekey,exc.externalconnection
key AS Connectionkey from entitlement_values ev_listner inner join entitlement_types et_listner o
n et_listner.entitlementtypekey = ev_listner.entitlementtypekey And lower(et_listner.Entitlementn
ame) = lower('elblistener') and COALESCE(ev_listner.status,0) < 2 and ev_listner.custom property1
in ('HTTPS', 'SSL') and str_to_date(concat(trim(substr(ev_listner.custom property5,locate(',',ev_l
```

```
istner.customproperty5)+1,20)),substr(ev_listner.customproperty5,(locate(',', ev_listner.custompr
operty5)+24),LENGTH(ev_listner.customproperty5))),'%Y-%c-%d %k:%i:%S') < CURRENT_TIMESTAMP() inne
r join entitlements2 listen_elb on ev_listner.entitlement_valuekey = listen_elb.entitlement_value
2key inner join entitlement_values ev_elb on listen_elb.entitlement_value1key = ev_elb.entitlemen
t_valuekey and COALESCE(ev_elb.status,0) < 2 inner join securitysystems sc on et_listner.systemke
y = sc.systemkey inner join endpoints ep on ep.SECURITYSYSTEMKEY=sc.systemkey inner join external
connection exc on sc.externalconnection = exc.externalconnectionkey inner join externalconnection
type exct on exc.externalconnectiontype = exct.externalconnectiontypekey and exct.connectiontype
= 'AWS' inner join externalconnattvalue excv on excv.connectiontype = exc.externalconnectionkey a
nd excv.attributekey = 'AWS_ACCOUNT_ID'
```

Dissociate User From Policy

When Dissociate User From Policy is configured as an allowed action, it disassociates the user in AWS from the policy.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name', ac.NAME
as 'IAM User', acr.user_creation_time as 'Creation Time',if(acr.mfa_active='true', 'Enabled', 'No
t Enabled' ) as 'MFA Status',ev.entitlement_value as 'Attached Policy Name',e2p.attribute_value a
```

```
s 'Allowed Action',ev.ENTITLEMENT_VALUEKEY as entvaluekey,et.endpointkey as externalConnectionKe
y,ac.ACCOUNTKEY as acctKey ,ep.ENDPOINTKEY AS endpointKey from aws_credreport acr inner join acco
unts ac on lower(ac.customproperty4) = lower(acr.arn) and acr.mfa_active = 'false' and acr.passwo
rd_enabled = 'true' inner join account_entitlements1 ae1 on ac.accountkey = ae1.accountkey inner
join entitlement_values ev on ae1.ENTITLEMENT_VALUEKEY = ev.ENTITLEMENT_VALUEKEY inner join entit
lement_types et on et.entitlementtypekey = ev.entitlementtypekey And et.entitlementname = 'AWSPol
icy' inner join entitlements2 e2 on ev.entitlement_valuekey = e2.entitlement_value1key inner Joi
n entitlement_values ev1 on e2.entitlement_value2key = ev1.entitlement_valuekey inner join entitl
ement_types et1 on et1.entitlementtypekey = ev1.entitlementtypekey And et1.entitlementname = 'AWS
PolicyStatement' inner join entitlements2_privilege e2p on e2p.entitlement2uid = e2.entitlement2u
id and e2p.attribute_name = 'AllowAction' And e2p.attribute_value like '%:*' inner join securitys
ystems sc on ac.SYSTEMID = sc.systemkey inner join endpoints ep on ep.SECURITYSYSTEMKEY=sc.system
key inner join externalconnection exc on sc.externalconnection = exc.externalconnectionkey inner
join externalconnectiontype exct on exc.externalconnectiontype = exct.externalconnectiontypekey a
nd exct.connectiontype = 'AWS' inner join externalconnattvalue excv on excv.connectiontype = ex
c.externalconnectionkey and excv.attributekey = 'AWS_ACCOUNT_ID'
```

Delete Access Key Of User

Each user has an Access Key and a Secret Access Key for user activation in AWS. When Delete Access Key Of User is configured as an allowed action, it deletes the Access Key of the user. The user will not be able to access AWS.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name', ac.name
as 'IAM User',acr.user_creation_time 'Creation Time', acr.password_enabled as 'Password Enabled',
 acr.password_last_used as 'Password Last used', acr.access_key_1_active as 'Access Key 1 Activ
e', acr.access_key_1_last_used_date as 'Access Key 1 Last Used', acr.access_key_2_active as 'Acce
ss Key 2 Active', acr.access_key_2_last_used_date as 'Access Key 2 Last Used' , ev.entitlement_va
luekey AS entvaluekey,exc.externalconnectionkey AS Connectionkey,ep.ENDPOINTKEY AS endpointKey fr
om accounts ac INNER JOIN securitysystems sc ON ac.systemid = sc.systemkey inner join endpoints e
p on ep.SECURITYSYSTEMKEY=sc.systemkey INNER JOIN externalconnection exc on sc.externalconnection
 = exc.externalconnectionkey INNER JOIN externalconnectiontype exct on exc.externalconnectiontype
 = exct.externalconnectiontypekey and exct.connectiontype = 'AWS' INNER JOIN externalconnattvalue
 excv on excv.connectiontype = exc.externalconnectionkey and excv.attributekey = 'AWS_ACCOUNT_I
D' INNER JOIN aws_credreport acr on acr.aws_account_id = excv.attributevalue and ac.custompropert
y4 = acr.arn and ac.status = 1 and acr.password_enabled = 'true' and (acr.access_key_1_active =
'true' or acr.access_key_2_active = 'true')
```

Revoke S3 Bucket Anonymous Access

When Revoke S3 Bucket Anonymous Access is configured as an allowed action, it revokes an anonymous public access on the S3

Bucket.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name',ev.entitl
ement_value as 'Trail Name', ev1.entitlement_value as 'Bucket Name', ev1.customproperty8 as 'Crea
tion Time',ev1.customproperty3 as 'Tags',et.endpointkey as externalConnectionKey ,ep.ENDPOINTKEY
AS endpointKey,ev.entitlement_valuekey AS entvaluekey from entitlement_values ev Inner Join entit
lement_types et on ev.entitlementtypekey = et.entitlementtypekey and et.Entitlementname = 'AWSClo
udTrail' and ev.status = 1 inner Join entitlement_values ev1 inner join entitlement_types et1 on
ev1.entitlementtypekey = et1.entitlementtypekey and et1.entitlementname = 's3bucket' and ev.CUSTO
MPROPERTY1 = ev1.entitlement_value and ev1.status = 1 inner join securitysystems sc on et.systemk
ey = sc.systemkey INNER JOIN endpoints ep on ep.SECURITYSYSTEMKEY=sc.systemkey INNER JOIN externa
lconnection exc on exc.externalconnectionkey = sc.externalconnection INNER JOIN externalconnattva
lue excv on excv.connectiontype = exc.externalconnectionkey and excv.attributekey = 'AWS_ACCOUN
T_ID' INNER JOIN entitlements2 e2 on e2.entitlement_value1key = ev1.Entitlement_valuekey INNER JO
IN entitlement_values ev2 on ev2.Entitlement_valuekey = e2.entitlement_value2key and ev2.Entitlem
ent_value = 'AllUsers' INNER JOIN entitlements2_privilege e2p on e2p.entitlement2uid = e2.entitle
ment2uid
```

Delete RDS

Amazon Relational Database Service (RDS) makes it easy to setup, operate, and scale a relational database in the cloud. When Delete RDS is configured as an allowed action, it deletes an existing RDS.

It is recommended to take a snapshot of the RDS before deleting an RDS.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name' , rds.ent
itlement_value as 'DB Instance',rds.customproperty1 as 'Region',rds.customproperty2 as 'Engine',r
ds.customProperty18 as 'Publicly Accessible' , rds.customproperty14 as 'Resource Status',scg.enti
tlement_value as 'Security Group', 'Inbound' as 'Rule', ev.customproperty1 as 'Source' ,ev.custom
property2 as 'Protocol',ev.customproperty3 as 'From Port',ev.customproperty4 as 'To Port',rds.cus
tomproperty3 as 'Tags',rds.ENTITLEMENT_VALUEKEY as entvaluekey ,et.endpointkey as externalConnect
ionKey ,ep.ENDPOINTKEY AS endpointKey from entitlement_values ev INNER JOIN entitlement_types et
ON ev.entitlementtypekey = et.entitlementtypekey And et.Entitlementname = 'SGIBRules' INNER JOIN
securitysystems sc ON et.systemkey = sc.systemkey INNER JOIN endpoints ep on ep.SECURITYSYSTEMKE
Y=sc.systemkey INNER JOIN externalconnection exc ON sc.externalconnection = exc.externalconnectio
nkey INNER JOIN externalconnectiontype exct ON exc.externalconnectiontype = exct.externalconnecti
ontypekey and exct.connectiontype = 'AWS' INNER JOIN externalconnattvalue excv ON excv.connection
```

```
type = exc.externalconnectionkey and excv.attributekey = 'AWS_ACCOUNT_ID' INNER JOIN entitlements
2 e2 ON e2.entitlement_value2key = ev.entitlement_valuekey INNER JOIN entitlement_values scg ON e
2.entitlement_value1key = scg.entitlement_valuekey INNER JOIN entitlement_types et1 ON et1.entitl
ementtypekey = scg.entitlementtypekey and et1.entitlementname = 'AWSSecurityGroup' LEFT OUTER JOI
N entitlementmap emap ON e2.entitlement_value1key = emap.entitlement_value2key INNER JOIN entitle
ment_values rds ON emap.entitlement_value1key = rds.entitlement_valuekey INNER JOIN entitlement_t
ypes et2 ON et2.entitlementtypekey = rds.entitlementtypekey and et2.entitlementname = 'RdsDbInsta
nce' And rds.status = 1 and rds.customproperty18 = 'true' Where ev.customproperty1 = '0.0.0.0/0'
and ev.customproperty2 not in ('icmp','udp') and ev.status = 1 and COALESCE(scg.status,0) < 2 an
d rds.entitlement_value != 'null'
```

Remove Vpc Peering

VPC Peering is a networking connection between 2 Virtual Private Clouds, that enables you to route traffic between them using IPv4/ IPv6 addresses. It is mainly used for high-availability and avoids single point failure. When Remove Vpc Peering is configured as an allowed action, it removes a VPC network connection between two VPCs following which data replication would not occur.

You can establish VPC peering in your account, in another VPC account, or in another AWS region.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS Account', ev.entitlement_value as 'Violated Peering V
PC IDs', ev.customproperty8 as 'Central VPC',ev.customproperty9 as 'vpc_region',ev.customproperty
3 as 'vpc_Tags',ev.ENTITLEMENT_VALUEKEY as entvaluekey ,et.endpointkey as externalConnectionKey f
rom entitlement_values ev inner join entitlement_types et on et.entitlementtypekey = ev.entitleme
nttypekey and et.entitlementname = 'VpcPeering' and ev.ENTITLEMENT_VALUEKEY not in (select distin
ct ev.ENTITLEMENT_VALUEKEY from entitlement_values ev inner join entitlement_types et on et.entit
lementtypekey = ev.entitlementtypekey and et.entitlementname = 'VpcPeering' )INNER JOIN securitys
ystems sc ON et.systemkey = sc.systemkey INNER JOIN endpoints ep on ep.SECURITYSYSTEMKEY=sc.syste
mkey INNER JOIN externalconnection exc ON sc.externalconnection = exc.externalconnectionkey INNE
R JOIN externalconnectiontype exct ON exc.externalconnectiontype = exct.externalconnectiontypekey
 and exct.connectiontype = 'AWS' INNER JOIN externalconnattvalue excv ON excv.connectiontype = ex
c.externalconnectionkey and excv.attributekey = 'AWS_ACCOUNT_ID' and ev.status = 1
```

Enable S3 Logging

When Enable S3 Logging is configured as an allowed action, it enables logging of requests made to a S3 bucket.
Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name', ev.ENTIT
LEMENT_VALUE as 'Cluster ID', ev.customproperty1 as 'Cluster Name',ev.customproperty9 as 'Regio
```

```
n',ev.customProperty11 as 'Creation Time',If(ev.customproperty2 is NULL, 'No', 'Yes') as 'Logging
 Enabled',ev.customProperty3 as 'Tags',ev.ENTITLEMENT_VALUEKEY as entvaluekey ,et.endpointkey as
externalConnectionKey,ev.entitlement_valuekey AS entvaluekey,ep.ENDPOINTKEY AS endpointKey from e
ntitlement_values ev Inner Join entitlement_types et on ev.entitlementtypekey = et.entitlementtyp
ekey and et.Entitlementname = 'EMR' and ev.customproperty2 is NULL and ev.status = 1 inner join s
ecuritysystems sc on et.systemkey = sc.systemkey INNER JOIN endpoints ep on ep.SECURITYSYSTEMKE
Y=sc.systemkey INNER JOIN externalconnection exc on exc.externalconnectionkey = sc.externalconnec
tion INNER JOIN externalconnattvalue excv on excv.connectiontype = exc.externalconnectionkey and
excv.attributekey = 'AWS_ACCOUNT_ID';
```

Create Entitlement Value Certification

When Create Entitlement Value Certification is configured as an allowed action, it creates and launches the certification for the entitlement of the entitlement owner.

Enable Cloud Trail Validation

When Enable Cloud Trail Validation is configured as an allowed action, it enables the log file integrity validation for various cloud trails in AWS.

Sample query:

SQL

```
select distinct excv.attributevalue as 'AWS AccountID',ep.displayname as 'Account Name', ev.ENTIT
LEMENT_VALUE as 'Cluster ID', ev.customproperty1 as 'Cluster Name',ev.customproperty9 as 'Regio
n',ev.customProperty11 as 'Creation Time',If(ev.customproperty2 is NULL, 'No', 'Yes') as 'Logging
 Enabled',ev.customProperty3 as 'Tags',ev.ENTITLEMENT_VALUEKEY as entvaluekey ,et.endpointkey as
externalConnectionKey,ev.entitlement_valuekey AS entvaluekey,ev.entitlement_valuekey AS entvaluek
ey,ep.ENDPOINTKEY AS endpointKey from entitlement_values ev Inner Join entitlement_types et on e
v.entitlementtypekey = et.entitlementtypekey and et.Entitlementname = 'EMR' and ev.customproperty
2 is NULL and ev.status = 1 inner join securitysystems sc on et.systemkey = sc.systemkey INNER JO
IN endpoints ep on ep.SECURITYSYSTEMKEY=sc.systemkey INNER JOIN externalconnection exc on exc.ext
ernalconnectionkey = sc.externalconnection INNER JOIN externalconnattvalue excv on excv.connectio
ntype = exc.externalconnectionkey and excv.attributekey = 'AWS_ACCOUNT_ID';
```

Execute Lambda Function

When Execute Lambda Function is configured as an allowed action, it executes the Lambda function. Ensure that the Lambda function is already created.

The AWS Lambda executes your function on your behalf.

The Analytics query must have the following columns:

- **entvaluekey** - Entitlement value key of the entitlement to which the account will be assigned.
- **externalConnectionKey** - Connection key of the AWS Connection.
- **lambdaName** - Name of the Lambda function to be executed.
- **region** - Region where the Lambda Function is present.
- **inputJson** - JSON body to be sent as a test event to the Lambda Function.
- **lambdaAccountId** - AWS Account ID of the AWS account where the Lambda function is present.

Sample query:

SQL

```
Select excv.attributevalue as AccountID,ac.CUSTOMPROPERTY4 as 'Account ARN',ac.name as 'IAM Use
r',ac.CREATED_ON as 'AccountCreateDate',substring_index(acat.ATTRIBUTE_VALUE,',',1) as 'AccessKe
y',substring_index(substring_index(acat.ATTRIBUTE_VALUE,',',2),',',-1) as 'AccessKey CreateDate',
 NULL as entvaluekey,'us-east-1' as region,'Execute Lambda Function' as 'Default_Action_For_Analy
tics', 'AcmePOC-deleteaccesskey' as lambdaName, exc.EXTERNALCONNECTIONKEY as externalConnectionKe
y,CONCAT('{"awsiamuser":"', ac.name ,'","accesskeyid":"',substring_index(acat.ATTRIBUTE_VALU
E,',',1),'","crossaccountrole":','"arn:aws:iam::661222050851:role/saviynt-analyzer-poc-SaviyntAWS
Role-1M7ICRQQT66I0"}') as inputJson, '533811351211' as lambdaAccountId from accounts ac Inner joi
n securitysystems sc on ac.SYSTEMID = sc.systemkey Inner Join externalconnection exc on sc.extern
alconnection = exc.externalconnectionkey Inner Join externalconnectiontype exct on exc.externalco
```

```
nnectiontype = exct.externalconnectiontypekey and exct.connectiontype = 'AWS' Inner join external
connattvalue excv on excv.connectiontype = exc.externalconnectionkey and excv.attributekey ='AW
S_ACCOUNT_ID' inner join account_attributes acat on ac.accountkey = acat.accountkey and acat.attr
ibute_name = 'accessKeyMetaData_Event' where substring_index(substring_index(acat.ATTRIBUTE_VALU
E,',',2),',',-1) > (Select ah.UPDATEDate from analytics_analyticshistory as ah,analyticsconfig a
s aconf where ah.ANALYTICSCONFIG=aconf.ANALYTICSKEY and aconf.ANALYTICSNAME = 'Access keys Creati
on' order by ah.analyticshistory desc limit 1);
```

Customizing Lambda Function

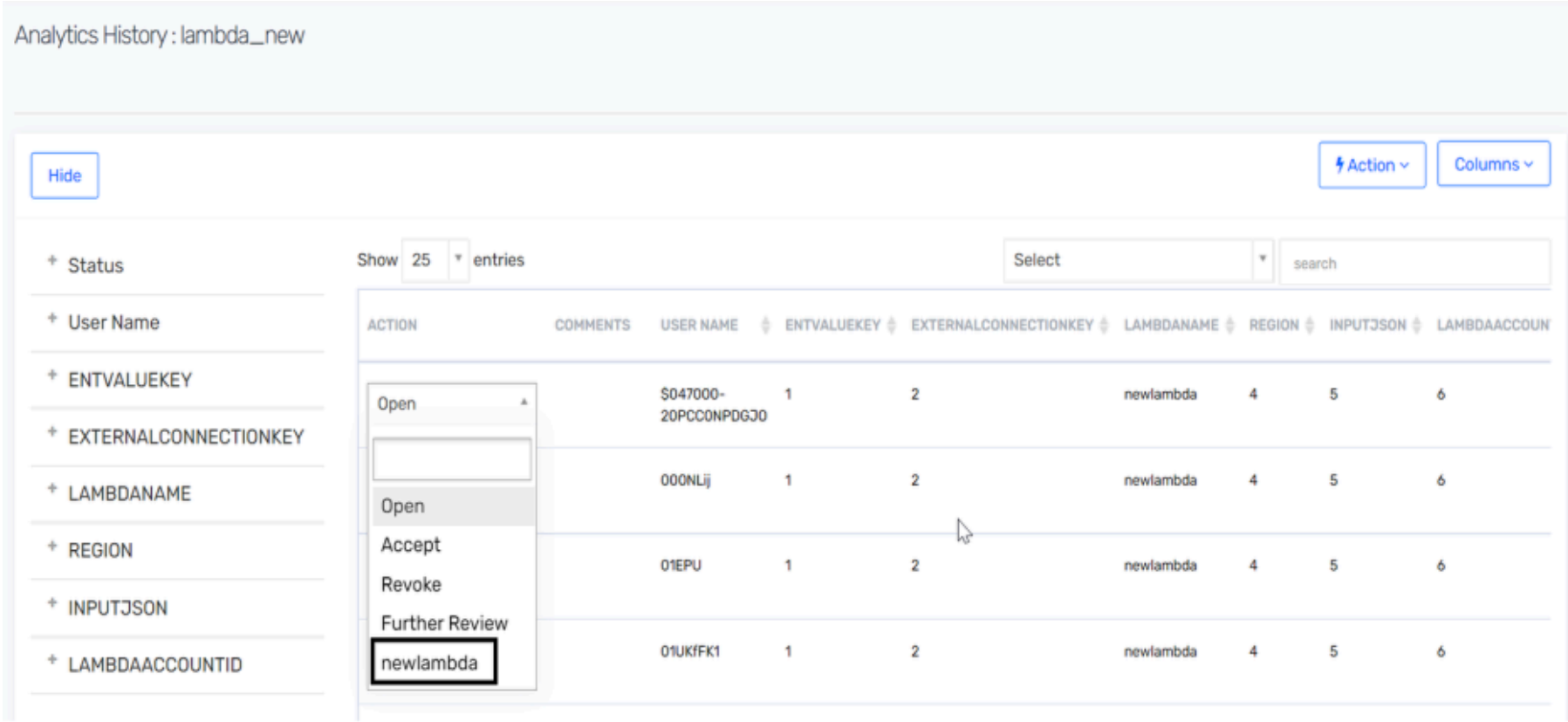
You can customize the name of the **Execute Lambda Function** option by creating an analytics with the SQL query that supports Lambda function. After executing the query, the Lambda function name configured in the query is displayed in the Action drop down.

Sample query:

SQL

```
select username,'1' as entvaluekey,'2' as externalConnectionKey,'newlambda' as lambdaName,'4' as
region,'5' as inputJson,'6' as lambdaAccountId from users limit 5
```

After running the analytics by executing the sample query, the customized lambda name appears in the Action dropdown. The following screenshot illustrates the same:



Generate Token

When Generate Token is configured as an allowed action, it allows you to generate the access token stored in the HashiCorp Vault connector and periodically rotate SECRET_ID.

Sample query:

SQL

```
select exc.externalconnectionkey as externalConnectionKey, exc.CONNECTIONNAME as CONNECTIONNAME, exc.status as connStatus, exc.statusForEnableDisable, sysdate() as sysdate, case when exc.statusForEnableDisable = '1' then 'Generate Token' end as Default_Action_For_Analytics from externalconnection exc, externalconnectiontype exct WHERE exc.externalconnectionType = exct.externalconnectiontypekey AND exct.connectiontype = 'Hashicorp';
```

Below are few examples where you can automate actions on all the records of a report:

- Deprovision users once the Term Date is Current Date
- Disable accounts with expiring passwords
- Close, or power off instances with incorrectly configured network ports

Example: Create an analytics control with a default action to disable accounts with expired passwords. To create this control, use the below sample SQL query.

SQL

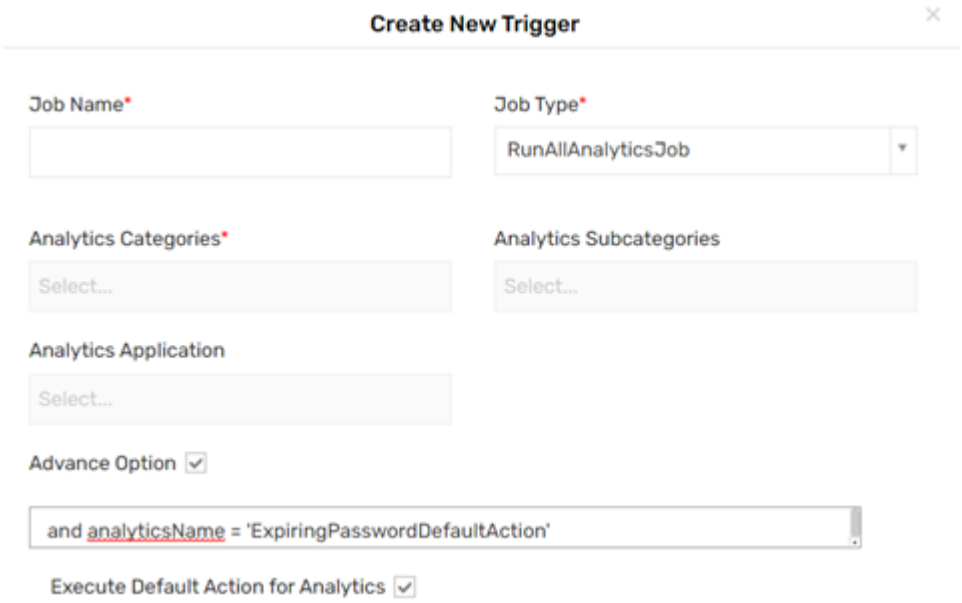
```
select
  u.userKey,
```

```
  u.username,
  u.LASTPASSWORDUPDATEDATE,
  date_add(
    u.LASTPASSWORDUPDATEDATE, Interval p.expireafter day
  ) as 'Expiry Date',
  'passwordExpired' as 'Default_Action_For_Analytics'
from
  users u,
  policyrule p
where
  u.PASSWORDEXPIRED = 0
  and p.SCOPE = 'USER'
  and p.EXPIREAFTER > 0
  and DATEDIFF(
    date_add(
      u.LASTPASSWORDUPDATEDATE, Interval p.expireafter day
    ),
    sysdate()
  ) = 0;
```

In the above query, Default_Action_For_Analytics is mandatory and a case-sensitive column for using default actions. Make sure to set

default actions as part of the allowed action. You must also write the same action aliasing as ‘Default_Action_For_Analytics’ in the SELECT clause of the query.

After creating the analytics control, run **RunAllAnalyticsJob** by creating a new trigger in Job Control Panel and ensure that **Execute Default Action for Analytics** is selected and other parameters are specified as needed. For more information, see Adding a New Job.



You can use the sample query for deprovisioning users.

SQL

```
select
```

```
  ae1.entitlement_valuekey as entvaluekey,
  ev.ENTITLEMENT_VALUE,
  ae1.accountkey as acctKey,
  a.name,
  'Deprovision Access' as 'Default_Action_For_Analytics'
from
  account_entitlements1 ae1,
  accounts a,
  entitlement_values ev
where
  a.accountkey = ae1.accountkey
  and ev.ENTITLEMENT_VALUEKEY = ae1.ENTITLEMENT_VALUEKEY
  and a.endpointkey = 12;
```

Sample query for closing an open network port

SQL

```
select
  distinct excv.attributevalue as 'AWS AccountID',
  ep.displayname as 'Account Name',
  Ec2.entitlement_value as 'EC2 Instance',
```

```
  evat1.attribute_Value as 'Name',
  Ec2.customproperty9 as 'Region',
  If(
    ec2.customproperty14 = 'Windows',
    'Windows', 'Linux'
  ) as 'Platform',
  if(
    Ec2.customproperty12 is NULL, 'Not Assigned',
    Ec2.customproperty12
  ) as 'Public IP',
  scg.entitlement_value as 'Security Group',
  'Inbound' as 'Rule',
  ev.customproperty1 as 'Source',
  ev.customproperty2 as 'Protocol',
  ev.customproperty3 as 'From Port',
  ev.customproperty4 as 'To Port',
  Ec2.customproperty3 as 'Tags',
  Ec2.ENTITLEMENT_VALUEKEY as entvaluekey,
  et.endpointkey as externalConnectionKey,
  ep.ENDPOINTKEY AS endpointKey
from
```

```
  entitlement_values ev
  INNER JOIN entitlement_types et ON ev.entitlementtypekey = et.entitlementtypekey
  And et.Entitlementname = 'SGIBRules'
  INNER JOIN securitysystems sc ON et.systemkey = sc.systemkey
  INNER JOIN endpoints ep on ep.SECURITYSYSTEMKEY = sc.systemkey
  INNER JOIN externalconnection exc ON sc.externalconnection = exc.externalconnectionkey
  INNER JOIN externalconnectiontype exct ON exc.externalconnectiontype = exct.externalconnectiont
ypekey
  and exct.connectiontype = 'AWS'
  INNER JOIN externalconnattvalue excv ON excv.connectiontype = exc.externalconnectionkey
  and excv.attributekey = 'AWS_ACCOUNT_ID'
  INNER JOIN entitlements2 e2 ON e2.entitlement_value2key = ev.entitlement_valuekey
  INNER JOIN entitlement_values scg ON e2.entitlement_value1key = scg.entitlement_valuekey
  INNER JOIN entitlement_types et1 ON et1.entitlementtypekey = scg.entitlementtypekey
  and et1.entitlementname = 'AWSSecurityGroup'
  LEFT OUTER JOIN entitlementmap emap ON e2.entitlement_value1key = emap.entitlement_value2key
  INNER JOIN entitlement_values Ec2 ON emap.entitlement_value1key = Ec2.entitlement_valuekey
  INNER JOIN entitlement_types et2 ON et2.entitlementtypekey = Ec2.entitlementtypekey
  and et2.entitlementname = 'EC2Instance'
  Left Outer Join entitlement_value_attrs evat1 on evat1.ENTITLEMENT_VALUE_KEY = ec2.entitlemen
t_Valuekey
```

```
  and evat1.ATTRIBUTE_NAME = 'Name'
Where
  ev.customproperty1 = '0.0.0.0/0'
  and ev.customproperty2 not in ('icmp', 'udp')
  and (
    (
      CAST(ev.customproperty3 AS UNSIGNED) <= 3389
      and CAST(ev.customproperty4 AS UNSIGNED) >= 3389
    )
    or (
      (ev.customproperty3 = 'ALL')
      and (ev.customproperty4 = 'ALL')
    )
  )
  and ev.status = 1
  and COALESCE(scg.status, 0) < 2
  and ec2.status = 1
  and Ec2.entitlement_value != 'null'
  and ev.customproperty1 not like '%terminated%'
```

Personally Identifiable Information Erasure

When Personally Identifiable Information (PII) Erasure action is configured as a default allowed action for V2 analytics controls, it allows you to erase the PII data of inactive users. This action removes the PII information and anonymizes the user id within Saviynt Identity Cloud. It also removes information like profile picture.

The Analytics query must have the columns given below:

userKey - Stores userkey of the user whose PII has to be erased.

Sample query:

SQL

```
SELECT userKey, PIIERASURESTATUS,'personallyIdentifiableInformationErasure' as Default_Action_Fo
r_Analytics FROM users where username='acme';
```

Note

- You must use the PII Erasure action carefully as there is no rollback of the erased data.
- You must configure this action as a Default action as manual action is not allowed.
- You cannot club this action with any other analytics action.
- This action is not supported for V1 Analytics controls.

Related Topics

Upgrading to MySQL 8