One Identity Manager 9.1.3

LDAP Connector for IBM RACF Reference Guide

Legend

WARNING: A WARNING icon highlights a potential risk of bodily injury or property damage, for which industry-standard safety precautions are advised. This icon is often associated with electrical hazards related to hardware.

CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

One Identity Manager LDAP Connector for IBM RACF Reference Guide
Updated - 29 April 2024, 13:31

For the most recent documents and product information, see Online product documentation.

Contents

Initializing and configuring the LDAP connector for IBM RACF

This document describes how to initialize and configure the RACF LDAP connector into an existing One Identity Manager system. This allows a One Identity Manager system to access, read, and update data stored in a RACF database on an IBM mainframe.

Detailed information about this topic

Prerequisites

- The IBM mainframe must have the IBM Tivoli Directory Server for z/OS installed and configured.

- An LDAP service account must be created in your RACF database with the appropriate permissions to administer users and groups on this platform. To be able to administer everything in the RACF database, the user will need the RACF "special" privilege.
- If more than 4096 records need to be retrieved from the RACF database in any one search (e.g. if there are more than 4096 users defined on the system) then the Quest RACF TDS Exit must be installed and configured.
- If data set profile data is to be synchronized, then the Quest RACF TDS Exit must be installed and configured.

NOTE: You can find Quest RACF TDS Exit on the installation medium in the directory `MFR\dvd\AddOn\RacfTDSExit`.

NOTE: Before attempting to connect to the Tivoli Directory Server with the One Identity Manager connector, it is recommended to first check that the LDAP server is running correctly. This can be tested with any LDAP browser for example the `LDP.exe` tool from Microsoft. For more information, see your *LDAP browser documentation*.

Platform support

The RACF LDAP connector has been verified for synchronization against the IBM mainframe running z/OS 1.8 (and RACF 1.8) or later.

Operating constraints

- There is an eight-character limit for user and group names on RACF.
- There is an eight-character limit for passwords on RACF.
- If the Quest RACF TDS Exit has not been installed, there is a limit of 4,096 records that can be read from the RACF system in any one search operation.
- If the Quest RACF TDS Exit has not been installed, the RACF dataset LDAP object is not available to the connector.

Pre-installation information

Read the information in this section before you install the RACF LDAP connector.

Detailed information about this topic

- User and group identifier on page 7
- RACF system users on page 7

User and group identifier

The LDAP implementation for RACF uses the `racfid` attribute to store the user name in a user object and the group name in a group object. The object containing the attribute defines whether it is referring to a user or a group.

RACF system users

RACF creates three special or system users that can be listed with an LDAP call. They are `iicerta`, `iimulti`, and `iisitec`. These system users cannot be altered by the connector through an LDAP call, so they are filtered by the connector. For example, when returning a list of all users in the RACF database, these three users will not be listed.

How to initialize and configure the RACF LDAP connector

NOTE: The following sequence describes how you configure a synchronization project if the Synchronization Editor is in expert mode.

To set up initial synchronization project for RACF

1. Start the Synchronization Editor and log in.
2. From the start page, select **Start a new synchronization project**.

   This starts the Synchronization Editor project wizard.
3. On the **Choose target system** page, select **RACF LDAP Connector**.
4. On the **System access** page, click **Next**.
5. On the **Create system connection** page, select **Create new system connection**.
6. On the system connection wizard start page, click **Next.**
7. On the **Network** page:
   a. In the **Server** field, enter the DNS name or IP address of your mainframe server.
   b. In the **Port** field, enter the port number.
   c. Click **Test** to ensure the server is accessible.
   d. The Tivoli Directory Server for z/OS supports LDAP v3. Enter the number 3 in the **Protocol version**.
   e. If SSL is to be used, select the **Use SSL** check box.
8. On the **Authentication** page:

- For basic authentication, do the following:
    a. Set the **Authentication method** to **Basic**.
    b. In the **Credentials** section, enter the full DN and password of the administrator account on your RACF system.
    c. Click **Test** to check that the credentials are valid.
- For external (client certificate) authentication, do the following:
    a. Set the **Authentication method** to **External**.
    b. In the **Client Certificate** section, enter the 40 character SHA1 thumbprint of the locally stored client certificate to be used for authentication.

    This thumbprint can be obtained from the Microsoft Management Console snap-in for managing certificates.

    NOTE: The certificate must be installed in the **Personal** area of the **Current User** certificate store.

    c. Click **Test** to check that the credentials are valid.

The schema is loaded from the RACF system.

9. On the **Search options** page:
    a. In the **Base DN for searches** drop-down list, select the correct base DN for your system.
    b. Clear the **Use paged search** check box.
10. On the **System attributes** page, in the **Revision properties** section, clear the **createTimestamp** and **modifyTimestamp** entries by double-clicking them.
11. Click **Finish**.

    This takes you back to the Synchronization Editor project wizard.
12. On the **One Identity Manager connection** page, enter the database connection data.

    This loads the RACF schema into One Identity Manager. Wait for this to complete.
13. On the **Select project template** page, select **Create blank project**.
14. On the **General** page, enter a display name for your synchronization project and set a scripting language if required.
15. Click **Finish**.
16. Select **Activate project**.

Related topics

- Domain filter setting on page 9
- User mapping information on page 10
- Group mapping information on page 15
- Data set profile mapping information on page 21

System variables

The following system variables must be defined for the attribute mappings. For more detailed information about variables, see the *One Identity Manager Target System Synchronization Reference Guide*.

Table 1: System variables

Name	Value
IdentDomain	The name of your RACF domain, for example, `RACF_DOMAIN`
UserLocation	Parent DN of your RACF user container, for example, `profiletype=user,cn=mainframe1,o=mycompany,c=com`
GroupLocation	Parent DN of your RACF group container, for example, `profiletype=group,cn=mainframe1,o=mycompany,c=com`
DatasetLocation	Parent DN of your RACF dataset container, for example, `profiletype=dataset,cn=mainframe1,o=mycompany,c=com`

Related topics

- Domain filter setting on page 9
- Property mapping rules on page 11
- Property mapping rules on page 17
- Property mapping rules on page 23

Domain filter setting

A domain filter must be created to identify information that has been retrieved from the RACF database to keep it separate from other imported data.

1. Update the One Identity Manager schema so that all entries are included.
    a. In the Synchronization Editor, open your RACF project.
    b. Select **Configuration > One Identity Manager connection**.
    c. In the **General** section, click **Update schema**.
    d. Click **Yes** in the next two dialogs.
    e. Click **OK** when complete.
2. In Manager

a. Select **LDAP > Domains**.

b. In the result list toolbar, click .

c. On the **General** tab, enter the following general master data.

Table 2: Domain Master Data

Property	Description
Display name	Display name, for example, `RACF Domain`
Distinguished name	Distinguished name of the domain, for example, `cn=mainframe1,o=mycompany,c=com`
Domain	Domain name, for example, `RACF_DOMAIN`
Structural object class	Structural object class representing the object type; enter **DCOBJECT**

d. Save the changes.

3. In the Synchronization Editor, open your RACF project.

a. Select **Configuration > One Identity Manager connection**.

b. Select the **Scope view** and click **Edit scope**.

c. Select the object type `LDPDomain` in the **Scope hierarchy** list and set the **Object filter** to `Ident_Domain ='$IdentDomain$'`.

d. Save the changes.

For more detailed information about scopes, see the *One Identity Manager Target System Synchronization Reference Guide*.

Related topics

- System variables on page 9

User mapping information

This section describes a possible mapping between a user account in RACF and the standard One Identity Manager database table called `LDAPAccount`.

- Set up a new mapping from `LDAPAccount(all)` to `racfUser(all)`.

For more detailed information about setting up mappings, see the *One Identity Manager Target System Synchronization Reference Guide.*

Detailed information about this topic

- Mandatory RACF user attributes on page 11
- Property mapping rules on page 11
- Object matching rules on page 14
- Sample user mapping on page 15

Mandatory RACF user attributes

When creating a user in the RACF database, the following LDAP attributes must be defined:

- objectclass
- racfid

Related topics

- Property mapping rules on page 11
- Object matching rules on page 14

Property mapping rules

- CanonicalName ← vrtEntryCanonicalName

  `vrtEntryCanonicalName` is a virtual property, set to the canonical name of the object in the connector.

  Sample value:

  `COM/MYCOMPANY/MAINFRAME1/USER/USER1234`
- cn ←→ racfid

  On the RACF system, `racfid` is the user ID.

  Sample value:

  `USER1234`
- DistinguishedName ← vrtEntryDN

  `vrtEntryDN` is a virtual property, set to the DN of the object in the connector. Select the **Force mapping against direction of synchronization** check box.

  Sample value:

  `racfid=USER1234,profiletype=user,cn=mainframe1,o=mycompany,c=com`
- ObjectClass ←→ objectClass

The `objectClass` attribute (multi-valued) on the RACF system. Select the **Ignore case sensitivity** check box.

Sample value:

`TOP;RACFBASECOMMON;RACFUSER`

- StructuralObjectClass ← vrtStructuralObjectClass

  `vrtStructuralObjectClass` on the RACF system defines the single object class for the object type. Select the **Ignore case sensitivity** check box.

  Sample value:

  `RACFUSER`

- UID_LDPDomain ← vrtIdentDomain

  Create a fixed value property variable on the RACF side called `vrtIdentDomain` that is set to the value `$IdentDomain$`. Map this to `UID_LDPDomain`. This causes a conflict, and the Property Mapping Rule Conflict Wizard opens automatically.

  To resolve the conflict

  1. In the Property Mapping Rule Conflict Wizard, select the first option and click **OK**.
  2. On the **Select an element** page, select **Ident_Domain** and click **OK**.
  3. Confirm the security prompt with **OK**.
  4. On the **Edit property** page:
     a. Clear **Save unresolvable keys**.
     b. Select **Handle failure to resolve as error**.

     To close the Property Mapping Rule Conflict Wizard, click **OK**.
  5. Select the **Force mapping against direction of synchronization** check box.

  Sample value:

  `RACF_DOMAIN`

- vrtParentDN → vrtEntryParentDN

  Create a fixed value property variable on the One Identity Manager side called `vrtParentDN` equal to a fixed string with the value `$UserLocation$`. Map this to `vrtEntryParentDN` on the RACF side.

  Sample value:

  `profiletype=user,cn=mainframe1,o=mycompany,c=com`

- vrtRDN → vrtEntryRDN

  Create a new variable on the One Identity Manager side of type **Script Property** with the name `vrtRDN` and a data type of **String**. In the **Scripts** section, enter one of the following scripts in the **Read script** section, depending on whether your project is configured for C# or Visual Basic.

C# Script

```
references VI.TSUtils.dll;

return (VI.TargetSystem.Base.Utils.LDAP.RDN.Create("cn", useOldValues ? $cn
[o]$ : $cn$).ToString()).Replace("cn=","racfid=");
```

VB Script

```
References VI.TSUtils.dll

Imports VI.TargetSystem.Base.Utils.LDAP

Dim name as String = ""

If useOldValues Then

     name = $cn[o]$

Else

     name = $cn$

End If

return RDN.Create("cn",name).ToString().Replace("cn=","racfid=")
```

Then map this to `vrtEntryRDN` on the RACF side.

Sample value:

```
USER1234
```

- userPassword → racfPassword

  Used to change a user's RACF password. A condition must be set on this rule to map the password only when there is a value to be copied.

  To add a condition

  1. Create the mapping.
  2. Edit the property mapping rule.
  3. Expand the **Condition for execution** section at the bottom of the dialog.
  4. Click **Add condition** and set the following condition (a blank password is indicated by using two apostrophe characters).

     ```
     Left.UserPassword<>''
     ```

- UID_LDAPContainer ← vrLDAPContainerDN

  This is a workaround needed to support group mappings. Create a new fixed value variable on the RACF side of type **String** with no value called `vrtLDAPContainerDN` with the value set to `$UserLocation$`. This generates a property mapping rule conflict.

  To resolve the conflict

  1. In the Property Mapping Rule Conflict Wizard, select the first option and click **OK**.
  2. On the **Select an element** page, select **DistinguishedName** and click **OK**.

3. Confirm the security prompt with **OK**.
4. On the **Edit property** page:
    a. Clear **Save unresolvable keys**.
    b. Select **Handle failure to resolve as error**.
    c. Select **Ignore case**.
5. To close the Property Mapping Rule Conflict Wizard, click **OK**.

Related topics

- Mandatory RACF user attributes on page 11
- System variables on page 9
- Object matching rules on page 14
- Sample user mapping on page 15

Object matching rules

- DistinguishedName (primary rule) vrtEntryDN

  `vrtEntryDN` is a virtual property, set to the DN of the object in the connector. This forms a unique ID to distinguish individual user objects on the RACF system.

  To convert this mapping into an object matching rule

  1. Select the property mapping rule in the rule window.
  2. Click in the rule view toolbar.

     A message appears.
  3. Click **Yes** to convert the property mapping rule into an object matching rule and save a copy of the property mapping rule.
  4. Edit the object mapping rule and ensure that the **Case sensitive** check box is not selected.

  Sample value:

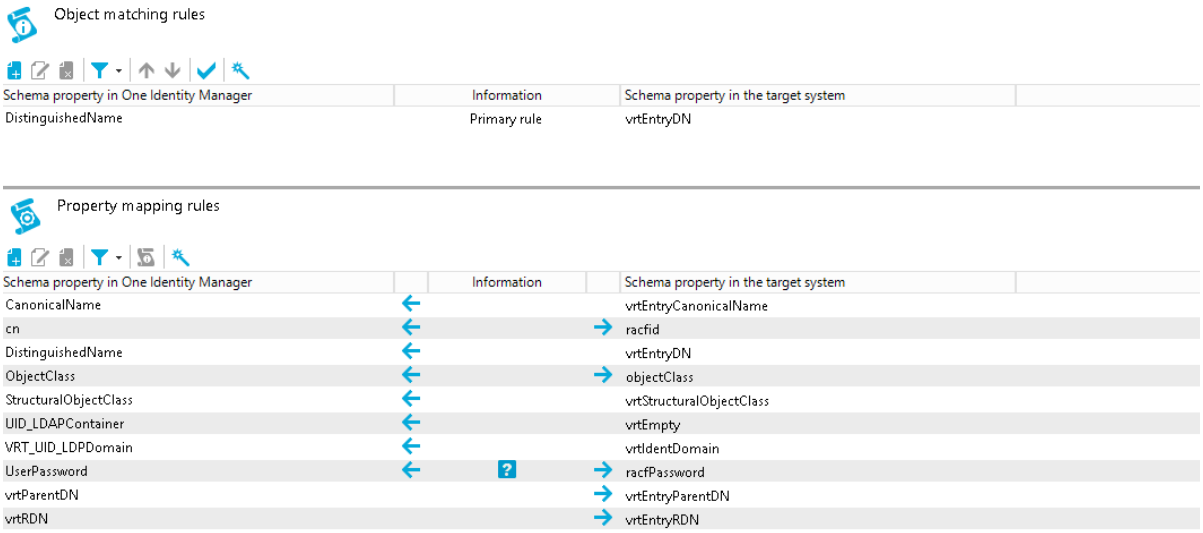
  `racfid=USER1234,profiletype=user,cn=mainframe1,o=mycompany,c=com`

Related topics

- Mandatory RACF user attributes on page 11
- Property mapping rules on page 11
- Sample user mapping on page 15

Sample user mapping

The following figure shows the user mapping in operation.



Group mapping information

This section shows a possible mapping between a user account in RACF and the standard One Identity Manager database table called `LDAPGroup`. The data set profile mapping used later also maps to `LDAPGroup`, so a filter must be applied in order to tell these apart.

- When creating the group mapping, add a new schema class as follows.

  Table 3: Schema class settings

Property	Value
Schema type	LDAPGroup
Display name	LDAPGroup (RACFGroup)
Class name	LDAPGroup_racfgroup
Select objects: Condition	StructuralObjectClass='racfgroup'
Select objects: Ignore case	Activated

- Select this new schema class, `LDAPGroup (RACF Group)`, for this mapping to `racfGroup (all)` on the RACFside.

For more detailed information about setting up mappings, see the *One Identity Manager Target System Synchronization Reference Guide*.

Detailed information about this topic

- Mandatory RACF group attributes on page 16
- Property mapping rules on page 17
- Object matching rules on page 20
- Data set profile mapping information on page 21
- Sample group mapping on page 20

Mandatory RACF group attributes

When creating a group in the RACF database, the following LDAP attributes must be defined:

- objectclass
- racfid

Related topics

- Property mapping rules on page 17
- Object matching rules on page 20

Property mapping rules

- CanonicalName ← vrtEntryCanonicalName

  `vrtEntryCanonicalName` is a virtual property, set to the canonical name of the object in the connector.

  Sample value:

  `COM/MYCOMPANY/MAINFRAME1/GROUP/USERGRP`

- cn ← → racfid

  On the RACF system, `racfid` is the group ID.

  Sample value:

  `USERGRP`

- DistinguishedName ← vrtEntryDN

  `vrtEntryDN` is a virtual property, set to the DN of the object in the connector. Select the **Force mapping against direction of synchronization** check box.

  Sample value:

  `racfid=USERGRP,profiletype=group,cn=mainframe1,o=mycompany,c=com`

- ObjectClass ← → objectClass

  The `objectClass` attribute (multi-valued) on the RACF system. Select the **Ignore case sensitivity** check box.

  Sample value:

  `TOP;RACFBASECOMMON;RACFGROUP`

- StructuralObjectClass ← vrtStructuralObjectClass

  `vrtStructuralObjectClass` on the RACF system defines the single object class for the object type.

  Sample value:

  `RACFGROUP`

- UID_LDPDomain ← vrtIdentDomain

  Create a fixed value property variable on the RACF side called `vrtIdentDomain` that is set to the value `$IdentDomain$`. Map this to `UID_LDPDomain`. This will cause a conflict and the Property Mapping Rule Conflict Wizard opens automatically.

  To resolve the conflict

  1. In the Property Mapping Rule Conflict Wizard, select the first option and click **OK**.
  2. On the **Select an element** page, select **Ident_Domain** and click **OK**.
  3. Confirm the security prompt with **OK**.
  4. On the **Edit property** page:

a. Clear **Save unresolvable keys**.

b. Select **Handle failure to resolve as error**.

5. To close the Property Mapping Rule Conflict Wizard, click **OK**.
6. Select the **Force mapping against direction of synchronization** check box.

Sample value:

```
RACF_DOMAIN
```

- vrtParentDN → vrtEntryParentDN

  Create a fixed value property variable on the One Identity Manager side called `vrtParentDN` equal to a fixed string with value `$GroupLocation$`. Map this to `vrtEntryParentDN` on the RACF side. Select the **Ignore case sensitivity** check box.

  Sample value:

  ```
  profiletype=group,cn=mainframe1,o=mycompany,c=com
  ```

- vrtRDN → vrtEntryRDN

  Create a new variable on the One Identity Manager side of type **Script Property** with the name `vrtRDN` and a data type of **String**. In the **Scripts** section, enter one of the following scripts in the **Read script** section, depending on whether your project is configured for C# or Visual Basic.

  C# Script

  ```
  references VI.TSUtils.dll;

  return (VI.TargetSystem.Base.Utils.LDAP.RDN.Create("cn", useOldValues ? $cn
  [o]$ : $cn$).ToString()).Replace("cn=","racfid=");
  ```

  VB Script

  ```
  References VI.TSUtils.dll

  Imports VI.TargetSystem.Base.Utils.LDAP

  Dim name as String = ""

  If useOldValues Then

       name = $cn[o]$

  Else

       name = $cn$

  End If

  return RDN.Create("cn",name).ToString().Replace("cn=","racfid=")
  ```

  Then map this to `vrtEntryRDN` on the RACF side.

  Sample value:

  ```
  USERGRP
  ```

- UID_LDAPContainer ← vrLDAPContainerDN

This is a workaround needed to support group mappings. Create a new fixed value variable on the RACF side of type **String** with no value called `vrtLDAPContainerDN` with the value set to `$GroupLocation$`. This generates a property mapping rule conflict.

To resolve the conflict

1. In the Property Mapping Rule Conflict Wizard, select the first option and click **OK**.
2. On the **Select an element** page, select **DistinguishedName** and click **OK**.
3. Confirm the security prompt with **OK**.
4. On the **Edit property** page:
    a. Clear **Save unresolvable keys**.
    b. Select **Handle failure to resolve as error**.
    c. Select **Ignore case**.
5. To close the Property Mapping Rule Conflict Wizard, click **OK**.

- vrtMember ← → racfGroupUserids

  This mapping is used to synchronize group membership information.

  1. Create a new virtual entry on the One Identity Manager side of type **Members of M:N schema types** with the name `vrtMember`. Select the **Ignore case** and **Enable relative component handling** check boxes.
  2. Add the following M:N schema types:
      a. Add an entry for `LDAPAccountInLDAPGroup`. Set the left box to `UID_LDAPGroup` and the right box to `UID_LDAPAccount`. Set the **Primary Key Property** to `DistinguishedName`.
      b. Add an entry for `LDAPGroupInLDAPGroup`. Set the left box to `UID_LDAPGroupParent` and the right box to `UID_LDAPGroupChild`. Set the **Primary Key Property** to `DistinguishedName`.
  3. Create a new mapping rule of type **Multi-reference mapping rule**. Set the rule name to **Member** and the mapping direction to **Both directions**. Set the One Identity Manager schema property to `vrtMember` and the RACF schema property to `racfGroupUserids`.

Related topics

- Mandatory RACF group attributes on page 16
- System variables on page 9
- Object matching rules on page 20
- Sample group mapping on page 20

Object matching rules

- DistinguishedName (primary rule) vrtEntryDN

  `vrtEntryDN` is a virtual property, set to the DN of the object in the connector. This forms a unique ID to distinguish individual group objects on the RACF system.

  To convert this mapping into an object matching rule

  1. Select the property mapping rule in the rule window.
  2. Click in the rule view toolbar.

     A message appears.
  3. Click **Yes** to convert the property mapping rule into an object matching rule and save a copy of the property mapping rule.
  4. Edit the object mapping rule and select the **Case sensitive** check box.

  Sample value:

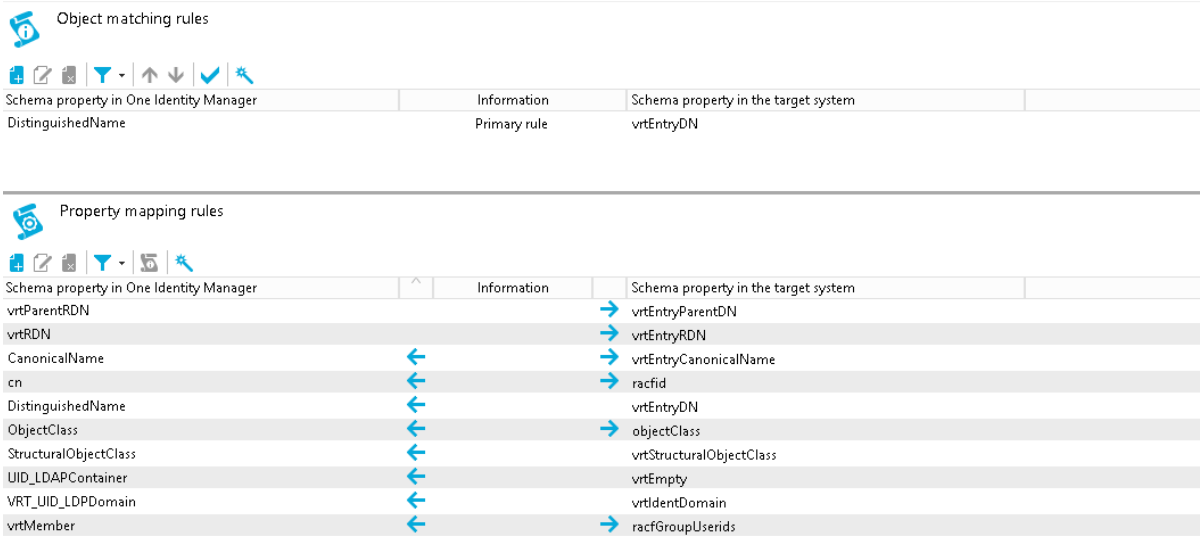
  `racfid=USERGRP,profiletype=group,cn=mainframe1,o=mycompany,c=com`

Related topics

- Mandatory RACF group attributes on page 16
- Property mapping rules on page 17
- Sample group mapping on page 20

Sample group mapping

The following figure shows the group mapping in operation.



Object matching rules

Schema property in One Identity Manager	Information	Schema property in the target system
DistinguishedName	Primary rule	vrtEntryDN

Property mapping rules

Schema property in One Identity Manager		Information		Schema property in the target system
vrtParentRDN			→	vrtEntryParentDN
vrtRDN			→	vrtEntryRDN
CanonicalName	←		→	vrtEntryCanonicalName
cn	←		→	racfid
DistinguishedName	←			vrtEntryDN
ObjectClass	←		→	objectClass
StructuralObjectClass	←			vrtStructuralObjectClass
UID_LDAPContainer	←			vrtEmpty
VRT_UID_LDPDomain	←			vrtIdentDomain
vrtMember	←		→	racfGroupUserids

System filtering on users and groups

The IBM Tivoli Directory Server does not support standard LDAP filtering but a limited level of functionality is supported. The only attribute that can be filtered is `racfid`, which can apply to both user and group names. This means it is possible to filter by the names of both users and groups.

This is done by applying a system filter to either the `racfuser` or `racfgroup` objects of the form `(racfid=<variable>*)` where `<variable>` applies to a common prefix.

For example, to import only users that start with **ABC**, the following system filter should be applied to the `racfuser` object:

```
(racfid=ABC*)
```

To import only groups beginning with **#1**, the following system filter should be applied to the `racfgroup` object:

```
(racfid=#1*)
```

Data set profile mapping information

This section shows a possible mapping between a user account in RACF and the standard One Identity Manager database table called `LDAPGroup` (a group is the closest equivalent in One Identity Manager to a data set profile). A mapping for RACF group already exists, so a filter needs to be applied in order to tell these apart.

- When creating the data set profile mapping, add a new schema class as follows.

  Table 4: Schema class settings

Property	Value
Schema type	LDAPGroup
Display name	LDAPGroup (Data set profile)
Class name	LDAPGroup_datasetprofile
Select objects: Condition	StructuralObjectClass='RACFDATASET'
Select objects: Ignore case	Activated

- Select this new schema class, `LDAPGroup (Data set profile)` for this mapping to `racfDataset(all)` on the RACF side.

For more detailed information about setting up mappings, see the *One Identity Manager Target System Synchronization Reference Guide*.

Detailed information about this topic

- Mandatory RACF data set profile attributes on page 22
- Property mapping rules on page 23
- Object matching rules on page 25
- Group mapping information on page 15
- Sample data set profile mapping on page 26

Mandatory RACF data set profile attributes

When creating a data set profile in the RACF database, the following LDAP attributes must be defined:

- objectclass
- racfDataset

Related topics

- Property mapping rules on page 23
- Object matching rules on page 25

Property mapping rules

- CanonicalName ← vrtEntryCanonicalName

  `vrtEntryCanonicalName` is a virtual property, set to the canonical name of the object in the connector.

  Sample value:

  `COM/MYCOMPANY/MAINFRAME1/DATASET/ABCDB.*.**`

- cn ← → racfDataset

  On the RACF system, this refers to the dataset profile ID.

  Sample value:

  `ABCDB.*.**`

- DistinguishedName ← vrtEntryDN

  `vrtEntryDN` is a virtual property, set to the DN of the object in the connector.

  Sample value:

  `racfdataset=ABCDB.*.**,profiletype=dataset,cn=mainframe1,o=mycompany,c=com`

- ObjectClass ← → objectClass

  The `objectClass` attribute (multi-valued) on the RACF system. Select the **Ignore case sensitivity** check box.

  Sample value:

  `TOP;RACFBASECOMMON;RACFDATASET`

- StructuralObjectClass ← vrtStructuralObjectClass

  `vrtStructuralObjectClass` on the RACF system defines the single object class for the object type.

  Sample value:

  `RACFDATASET`

- VRT_UID_LDPDomain ← vrtIdentDomain

  Create a fixed value property variable on the RACF side called `vrtIdentDomain` that is set to the value `$IdentDomain$`. Map this to `VRT_UID_LDPDomain`, the attribute created by One Identity Manager when this step was performed for a group mapping above.

  Sample value:

  `RACF_DOMAIN`

- vrtDatasetParentDN → vrtEntryParentDN

  Create a fixed value property variable on the One Identity Manager side called `vrtDatasetParentDN` equal to a fixed string with value `$DatasetLocation$`. Map this to `vrtEntryParentDN` on the RACF side.

  Sample value:

  `profiletype=dataset,cn=mainframe1,o=mycompany,c=com`

- vrtDatasetRDN → vrtEntryRDN

  Create a new variable on the One Identity Manager side of type **Script Property** with the name `vrtDatasetRDN` and a data type of **String**. In the **Scripts** section, enter one of the following scripts in the **Read script** section, depending whether your project is configured for C# or Visual Basic.

  C# Script

  ```
  references VI.TSUtils.dll;

  return (VI.TargetSystem.Base.Utils.LDAP.RDN.Create("cn", useOldValues ? $cn
  [o]$ : $cn$).ToString()).Replace("cn=","racfDataset=");
  ```

  VB Script

  ```
  References VI.TSUtils.dll

  Imports VI.TargetSystem.Base.Utils.LDAP

  Dim name as String = ""

  If useOldValues Then

       name = $cn[o]$

  Else

       name = $cn$

  End If

  return RDN.Create("cn",name).ToString().Replace("cn=","racfDataset=")
  ```

  Then map this to `vrtEntryRDN` on the RACF side.

  Sample value:

  ```
  ABCDB.*.**
  ```

- BusinessCategory ← → uid

  This is a multi-valued string that contains the RACF user IDs and the rights they are granted for a particular data set profile. Changes to this list on the RACF side can be performed by synchronizing the necessary changes from the One Identity Manager side. `BusinessCategory` was chosen for the mapping as it was a pre-existing multi-valued string.

  Sample value:

  ```
  USER001(READ); USER002(ALTER); USER003(READ)
  ```

- vrtDatasetMember ← → racfPermitId

  This mapping is used to synchronize data set membership information.

  1. Create a new virtual entry on the One Identity Manager side of type **Members of M:N schema types** with the name `vrtDatasetMember`. Select the **Ignore case** and **Enable relative component handling** check boxes.
  2. Add the following M:N schema types:

a. Add an entry for `LDAPAccountInLDAPGroup`. Set the left box to `UID_LDAPGroup` and the right box to `UID_LDAPAccount`. Set the **Primary Key Property** to `DistinguishedName`.

b. Add an entry for `LDAPGroupInLDAPGroup`. Set the left box to `UID_LDAPGroupParent` and the right box to `UID_LDAPGroupChild`. Set the **Primary Key Property** to `DistinguishedName`.

3. Create a new mapping rule of type **Multi-reference mapping rule**. Set the rule name to **Member** and the mapping direction to **Both directions**. Set the One Identity Manager schema property to `vrtDatasetMember` and the RACF schema property to `racfPermitId`.

   NOTE: When this membership mapping is set up at the same time as a mapping for groups (vrtMember <-> racfGroupUserids in the group mapping), the data set synchronization populates both the `vrtDatasetMember` and `vrtMember` attributes with the same values. The values stored in `vrtMember` can be ignored.

Related topics

- Mandatory RACF data set profile attributes on page 22
- System variables on page 9
- Object matching rules on page 25
- Sample data set profile mapping on page 26

Object matching rules

- DistinguishedName (primary rule) vrtEntryDN

  `vrtEntryDN` is a virtual property, set to the DN of the object in the connector. This forms a unique ID to distinguish individual dataset objects on the RACF system.

  To convert this mapping into an object matching rule

  1. Select the property mapping rule in the rule window.
  2. Click in the rule view toolbar.

     A message appears.
  3. Click **Yes** to convert the property mapping rule into an object matching rule and save a copy of the property mapping rule.

  Sample value:

  `racfdataset=ABCDB.*.**,profileType=dataset,cn=mainframe1,o=mycompany,c=com`

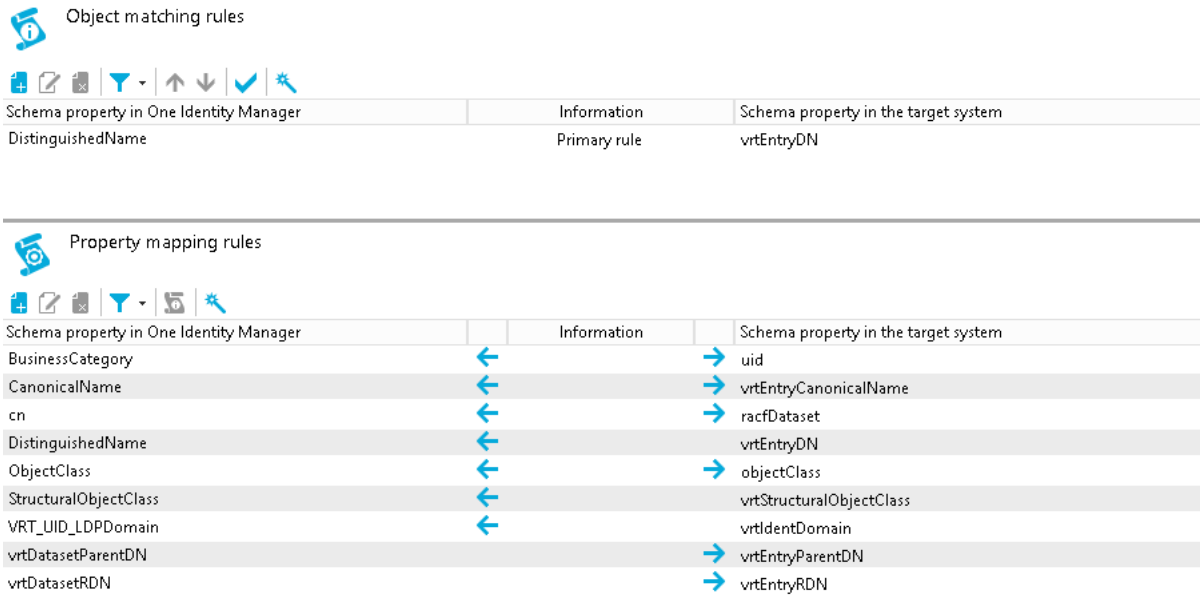
Related topics

- Mandatory RACF data set profile attributes on page 22
- Property mapping rules on page 23

- Sample data set profile mapping on page 26

Sample data set profile mapping

The following figure shows the data set profile mapping in operation.



Run TSO command

The RACF LDAP connector can be used to run any TSO command on the connected system if the Quest RACF TDS Exit has been installed and configured. This TSO command execution needs to be configured manually for the connector made available with One Identity Manager.

Create a custom defined process using the `MFRComponent` process component. Use the **RACF LDAP connector** server function to specify the execution server. The One Identity Manager Service is installed on this server with the RACF LDAP connector.

For more detailed information about configuring the server and creating processes, see the *One Identity Manager Configuration Guide*.

Auxiliary classes

The RACF user and group objects have a number of auxiliary classes available to add extra attributes. There are 13 of these auxiliary classes in total.

Auxiliary classes that can extend the RACF user object:

- SAFTSOSegment
- SAFDfpSegment
- racfCicsSegment
- racfLanguageSegment
- racfOperparmSegment
- racfWorkAttrSegment
- racfUserOmvsSegment
- racfUserOvmSegment
- racfNetviewSegment
- racfDCESegment

Auxiliary classes that can extend the RACF group object:

- racfGroupOmvsSegment
- racfGroupOvmSegment
- SAFDfpSegment

The additional attributes that each of these makes available is listed in Auxiliary classes on page 35.

When the RACF user or group object is viewed in the Synchronization Editor, all of the attributes made available by all of the above auxiliary classes are listed by default and can be used in user or group mappings. To make use of additional attributes during a synchronization to RACF, the user or group object must contain the corresponding object class for each additional attribute or the attribute will be discarded. The object class attribute for a user is multi-valued and must contain the full list of all object classes needed for the user.

For example, the auxiliary class `racfUserOvmSegment` contains an attribute called `racfOvmUid`.

To successfully synchronize a value to this attribute for a user, the user object must contain the value `racfUserOvmSegment` in its object class attribute.

RACF groups and RACF universal groups

A standard RACF group keeps track of its members in an attribute called `racfGroupUserIds`. This imposes a limit on the number of members a group can have because there is a fixed

amount of space in a group's profile to store this information. The limit is approximately 6,000 users.

To get around this, IBM introduced universal groups. Universal group profiles do not list user members whose group authority is set to **USE**, and since most users have this as their group authority, the number of possible user members is increased well over the 6,000 limit.

Creating a universal group

A universal group is created in the the same way as a standard group except that the `racfAttributes` attribute for the group must be set to `UNIVERSAL` when the group is created. This must be done when the group is created; a standard group cannot be converted to a universal group after it has been created.

Group authority

When a user is connected to a group, the user's group authority level must be specified. The default level is `USE`, but it is possible to set this to a different value. To do this, a virtual attribute called `vrtGroupPermission` must be enabled for user mappings. This is done in the RACF connection configuration wizard on the **Search Options** panel. Select the box next to **Use vrtGroupPermission** to enable this virtual attribute in user searches and mappings.

Synchronizing group members

There are a number of ways to synchronize group memberships. The method used depends on whether the group is universal group and whether the group authority level needs to be a different value from the default of **USE**. There are three options available, but note that only one of the three options should be used with any one group:

- Standard group and all users have default authority

  In this case, the list of group members must be synchronized to the `racfGroupUserIds` group attribute. Entries to be synchronized take the form of the DN of each user member. For more information, see Sample group mapping on page 20.

- Universal group and all users have default authority

  In this case, the group memberships must be synchronized on a per-user basis using the `racfConnectGroupName` user attribute. Entries to be synchronized take the form of the DN of each of the groups that the user is to be connected to.

- Any group type and some users have non-default authority

  In this case, the group memberships must be synchronized on a per-user basis using the virtual `vrtGroupPermission` user attribute. The values to be synchronized must take the form:

  ```
  <group ID> (<Authority level>)
  ```

RACF pass phrase support

Password values in RACF are eight characters or fewer in length. IBM has added support for longer passwords in RACF by implementing pass phrases. These longer values need to be stored differently to passwords.

When synchronising a user's One Identity Manager password to RACF, the length of the password determines where the password should be stored. If it is eight characters or fewer in length it must be synchronised to the `racfPassword` attribute. If it is longer than eight characters it must be synchronised to the `racfPassPhrase` attribute. This can be achieved as follows.

First, create a new variable on the One Identity Manager side of type **Script Property** with name `vrtsIsLongPassword` and a data type of **Boolean** – logical value. In the **Read script** section for this variable, enter the following script depending on the script language defined for the connector:

C# Script

```
if( $UserPassword$.ToString().Length < 9)
return false;
return true;
```

VB Script

```
if Len($UserPassword$)<9 Then
Return False
End If
Return True
```

Then set up the password mapping as follows:

- UserPassword → racfPassPhrase

  A condition needs to be set on this rule to map the password only when there is a value to be copied and it is more than eight characters in length.

  To add a condition

  - Create the mapping.
  - Edit the property mapping rule.
  - Expand the **Condition for execution** section at the bottom of the dialog.
  - Click **Add condition** and set the following condition (a blank password is indicated by using two apostrophe characters).

    ```
    Left.UserPassword<>'' and Left.vrtsIsLongPassword='1'
    ```

- UserPassword → racfPassword

  A condition needs to be set on this rule to map the password only when there is a value to be copied and it is eight characters or fewer in length.

To add a condition

- Create the mapping.
- Edit the property mapping rule.
- Expand the **Condition for execution** section at the bottom of the dialog.
- Click **Add condition** and set the following condition (a blank password is indicated by using two apostrophe characters).

  ```
  Left.UserPassword<>'' and Left.vrtsIsLongPassword='0'
  ```

Appendix A

RACF user attributes

The following table lists the RACF user attributes that are made available to One Identity Manager by the RACF LDAP connector.

Table 5: List of RACF user attributes

Attribute name
racfAttributes
racfAuthorizationDate
racfClassName
racfConnectGroupAuthority
racfConnectGroupName
racfConnectGroupUACC
racfDatasetModel
racfDefaultGroup
racfHavePassPhraseEnvelope
racfHavePasswordEnvelope
racfid
racfInstallationData
racfLastAccess
racfLogonDays
racfLogonTime
racfOwner
racfPassPhrase
racfPassPhraseChangeDate

Attribute name
racfPassPhraseEnvelope
racfPassword
racfPasswordChangeDate
racfPasswordEnvelope
racfPasswordInterval
racfProgrammerName
racfResumeDate
racfRevokeDate
racfSecurityLabel
racfSecurityLevel

Appendix B

RACF group attributes

The following table lists the RACF group attributes that are made available to One Identity Manager by the RACF LDAP connector.

Table 6: List of RACF group attributes

Attribute name
racfAuthorizationDate
racfDatasetModel
racfGroupNoTermUAC
racfGroupUniversal
racfGroupUserids
racfid
racfInstallationData
racfOwner
racfSubGroupName
racfSuperiorGroup

Appendix C

RACF data set profile attributes

If the Quest RACF TDS Exit has been installed and enabled, the following RACF data set profile attributes are made available to One Identity Manager by the RACF LDAP connector.

Table 7: List of RACF data set profile attributes

Attribute name
racfAccess
racfAudit
racfCreateGroup
racfDataset
racfErase
racfGlobalAudit
racfNotify
racfOwner
racfPermitid
racfUacc
racfWarning
uid

Appendix D

Auxiliary classes

The following list defines all of the auxiliary classes for RACF user and group classes, along with their associated attributes.

Auxiliary class `SAFDfpSegment` for RACF user and RACF group

- SAFDfpDataApplication
- SAFDfpDataClass
- SAFDfpManagementClass
- SAFDfpStorageClass

Auxiliary class `racfGroupOmvsSegment` for RACF group

- racfOmvsGroupId

Auxiliary class `racfGroupOvmSegment` for RACF group

- racfOvmUserId

Auxiliary class `SAFTsoSegment` for RACF user

- SAFAccountNumber
- SAFDefaultCommand
- SAFDestination
- SAFHoldClass
- SAFJobClass
- SAFMessageClass
- SAFDefaultLoginProc
- SAFLogonSize
- SAFMaximumRegionSize
- SAFDefaultSysoutClass
- SAFUserdata
- SAFDefaultUnit
- SAFTsoSecurityLabel

Auxiliary class `racfCicsSegment` for RACF user

- racfOperatorIdentification
- racfOperatorClass
- racfOperatorPriority
- racfOperatorReSignon
- racfTerminalTimeout

Auxiliary class `racfLanguageSegment` for RACF user

- racfPrimaryLanguage
- racfSecondaryLanguage

Auxiliary class `racfOperparmSegment` for RACF user

- racfStorageKeyword
- racfAuthKeyword
- racfMformKeyword
- racfLevelKeyword
- racfMonitorKeyword
- racfRoutcodeKeyword
- racfLogCommandResponseKeyword
- racfMGIDKeyword
- racfDOMKeyword
- racfKEYKeyword
- racfCMDSYSKeyword
- racfUDKeyword
- racfMscopeSystems
- racfAltGroupKeyword
- racfAutoKeyword

Auxiliary class `racfWorkAttrSegment` for RACF user

- racfWorkAttrUserName
- racfBuilding
- racfDepartment
- racfRoom
- racfAddressLine1
- racfAddressLine2
- racfAddressLine3
- racfAddressLine4
- racfWorkAttrAccountNumber

Auxiliary class `racfUserOmvsSegment` for RACF user

- racfOmvsUid
- racfOmvsHome
- racfOmvsInitialProgram

Auxiliary class `racfNetviewSegment` for RACF user

- racfNetviewInitialCommand
- racfDefaultConsoleName
- racfCTLKeyword
- racfMSGRCVRKeyword
- racfNetviewOperatorClass
- racfDomains
- racfNGMFADMKeyword

Auxiliary class `racfDCESegment` for RACF user

- racfDCEUUID
- racfDCEPrincipal
- racfDCEHomeCell
- racfDCEHomeCellUUID
- racfDCEAutoLogin

Auxiliary class `racfUserOvmSegment` for RACF user

- racfOvmUid
- racfOvmHome
- racfOvmInitialProgram
- racfOvmFileSystemRoot
- racfOvmHomeUUID

About us

One Identity solutions eliminate the complexities and time-consuming processes often required to govern identities, manage privileged accounts and control access. Our solutions enhance business agility while addressing your IAM challenges with on-premises, cloud and hybrid environments.

Contacting us

For sales and other inquiries, such as licensing, support, and renewals, visit https://www.oneidentity.com/company/contact-us.aspx.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at https://support.oneidentity.com/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to videos at www.YouTube.com/OneIdentity
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product