Active Roles Synchronization Service 8.2

Administration Guide

Copyright 2025 One Identity LLC.

ALL RIGHTS RESERVED.

This guide contains proprietary information protected by copyright. The software described in this guide is furnished under a software license or nondisclosure agreement. This software may be used or copied only in accordance with the terms of the applicable agreement. No part of this guide may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of One Identity LLC .

The information in this document is provided in connection with One Identity products. No license, express or implied, by estoppel or otherwise, to any intellectual property right is granted by this document or in connection with the sale of One Identity LLC products. EXCEPT AS SET FORTH IN THE TERMS AND CONDITIONS AS SPECIFIED IN THE LICENSE AGREEMENT FOR THIS PRODUCT, ONE IDENTITY ASSUMES NO LIABILITY WHATSOEVER AND DISCLAIMS ANY EXPRESS, IMPLIED OR STATUTORY WARRANTY RELATING TO ITS PRODUCTS INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTY OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT. IN NO EVENT SHALL ONE IDENTITY BE LIABLE FOR ANY DIRECT, INDIRECT, CONSEQUENTIAL, PUNITIVE, SPECIAL OR INCIDENTAL DAMAGES (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF PROFITS, BUSINESS INTERRUPTION OR LOSS OF INFORMATION) ARISING OUT OF THE USE OR INABILITY TO USE THIS DOCUMENT, EVEN IF ONE IDENTITY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. One Identity makes no representations or warranties with respect to the accuracy or completeness of the contents of this document and reserves the right to make changes to specifications and product descriptions at any time without notice. One Identity does not make any commitment to update the information contained in this document.

If you have any questions regarding your potential use of this material, contact:

One Identity LLC.
Attn: LEGAL Dept
4 Polaris Way
Aliso Viejo, CA 92656

Refer to our website (http://www.OneIdentity.com) for regional and international office information.

Patents

One Identity is proud of our advanced technology. Patents and pending patents may apply to this product. For the most current information about applicable patents for this product, please visit our website at http://www.OneIdentity.com/legal/patents.aspx.

Trademarks

One Identity and the One Identity logo are trademarks and registered trademarks of One Identity LLC. in the U.S.A. and other countries. For a complete list of One Identity trademarks, please visit our website at www.OneIdentity.com/legal/trademark-information.aspx. All other trademarks are the property of their respective owners.

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.

Synchronization Service Administration Guide
Updated - 09 January 2025, 14:45

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

Contents

Synchronization Service overview **1**
Synchronization Service features and benefits 1
Bidirectional synchronization 2
Delta processing mode 2
Synchronization of group membership 2
Windows PowerShell scripting 2
Attribute synchronization rules 3
Rule-based generation of distinguished names 3
Scheduling capabilities 3
Extensibility 3
Azure BackSync configuration 4
Technical overview 5
Synchronization Service 5
Capture Agent 6
Connectors and connected data systems 6
Sync workflows 6

Deploying Synchronization Service **8**
Installing Synchronization Service 8
Configuring Synchronization Service 9
Configuring Azure BackSync 12
Configuring manual Azure BackSync 13
Configuring automatic Azure BackSync 15
Settings updated after Azure BackSync configuration operation 17
Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync 19
Upgrade from Quick Connect and Synchronization Service 19
Transferring sync workflows from Quick Connect 20
Communication ports used by Synchronization Service 21

Deploying Synchronization Service for use with AWS Managed Microsoft AD24
Supported AWS Managed Microsoft AD deployment configuration 24
Synchronization Service features and limitations when used with AWS Managed Microsoft AD 25

Main steps of configuring Active Roles Synchronization Service for AWS Managed Microsoft AD 26
Deployment requirements for AWS Managed Microsoft AD support 26
Creating the AWS Managed Microsoft AD instance 27
Creating the EC2 instance for Active Roles Synchronization Service 28
Joining the EC2 instance to AWS Managed Microsoft AD 29
Creating the RDS instance for the Active Roles Synchronization Service SQL Server 29
Verifying connectivity between the EC2 and RDS instances 30
Installing and configuring Synchronization Service for AWS Managed Microsoft AD 31

Getting started 33
Synchronization Service Console 33
Gear icon 34
Sync Workflows tab 35
Sync History tab 35
Connections tab 36
Mapping tab 36
Password Sync tab 37
Configuring diagnostic logging 38
How to synchronize identity data 39
Synchronization Service Management Shell 39
Cmdlet naming conventions 40
Getting help 40

Connections to external data systems 42
External data systems supported with built-in connectors 43
Working with Active Directory 44
Creating an Active Directory connection 45
Modifying an Active Directory connection 46
Communication ports required to synchronize data between two Active Directory domains 47
Synchronizing user passwords between two Active Directory domains 48
Synchronizing SID history of users or groups 48
Working with an AD LDS (ADAM) instance 49
Creating an AD LDS (ADAM) instance connection 50
Modifying an existing AD LDS (ADAM) instance connection 51
Working with Skype for Business Server 52

Creating a new Skype for Business Server connection 52
Modifying an existing Skype for Business Server connection 53
Supported Skype for Business Server data 54
Attributes required to create a Skype for Business Server user 58
Getting or setting the Telephony option value in Skype for Business Server 58
Working with Oracle Database 59
Creating an Oracle Database connection 60
Modifying an existing Oracle Database connection 61
Sample SQL queries for working with an Oracle Database 63
Working with Oracle Database user accounts 64
Creating an Oracle Database user accounts connection 65
Modifying an existing Oracle Database user account connection 66
Sample SQL queries for working with Oracle Database user accounts 68
Working with Exchange Server 68
Creating a new connection to Exchange Server 69
Modifying an existing connection to Exchange Server 71
Exchange Server data supported out of the box 72
Scenario: Migrate mailboxes from one Exchange Server to another 87
Working with Active Roles 88
Creating an Active Roles connection 89
Modifying an Active Roles connection 90
Working with One Identity Manager 91
Creating a One Identity Manager connection 92
Modifying a One Identity Manager connection 92
One Identity Manager Connector configuration file 93
Working with a delimited text file 94
Creating a delimited text file connection 95
Modifying an existing delimited text file connection 96
Working with Microsoft SQL Server 99
Creating a Microsoft SQL Server connection 99
Modifying an existing Microsoft SQL Server connection 101
Sample queries to modify SQL Server data 103
Working with Micro Focus NetIQ Directory 104
Creating a Micro Focus NetIQ Directory connection 104
Modifying an existing Micro Focus NetIQ Directory connection 106

Working with Salesforce 108
Creating a Salesforce connection 109
Modifying an existing Salesforce connection 109
Salesforce data supported for synchronization 110
Scenario: Provisioning users from an Active Directory domain to Salesforce 114
Working with ServiceNow 115
Creating a ServiceNow connection 116
Modifying an existing ServiceNow connection 117
ServiceNow data supported for synchronization 118
Working with Oracle Unified Directory 119
Creating an Oracle Unified Directory connection 119
Modifying an existing Oracle Unified Directory Server connection 121
Working with an LDAP directory service 123
Creating an LDAP directory service connection 124
Modifying an existing Generic LDAP directory service connection 126
Specify password sync parameters for LDAP directory service 129
Working with an OpenLDAP directory service 129
Creating an OpenLDAP directory service connection 130
Modifying an existing OpenLDAP directory service connection 131
Working with IBM DB2 133
Creating an IBM DB2 connection 134
Modifying an existing IBM DB2 connection 135
Working with IBM AS/400 137
Creating an IBM AS/400 connection 138
Modifying an existing IBM AS/400 connection 139
Additional considerations for an IBM AS/400 connection 139
Working with IBM RACF 140
Creating an IBM RACF connection 141
Modifying an IBM RACF connection 142
Example of mapping for dataset information 142
Creating SQL database and table 142
Povisioning datasets 143
Updating datasets 144
Deprovisioning datasets 145
Working with TSO command 145

Working with MySQL database 147
Creating a MySQL database connection 147
Modifying an existing MySQL database connection 149
Working with an OLE DB-compliant relational database 151
Creating an OLE DB-compliant relational database connection 152
Modifying an existing OLE DB-compliant data source connection 153
Working with SharePoint 154
Creating a SharePoint connection 155
SharePoint data supported for data synchronization 155
Considerations for creating objects in SharePoint 210
Working with Microsoft 365 211
Creating a Microsoft 365 connection 211
Modifying a Microsoft 365 connection 214
Microsoft 365 data supported out of the box 217
Objects and attributes specific to Microsoft 365 services 306
How the Microsoft 365 Connector works with data 307
Working with Microsoft Azure Active Directory 308
Creating a Microsoft Azure Active Directory connection 308
Modifying a Microsoft Azure Active Directory connection 313
Microsoft Azure Active Directory data supported for synchronization 316
Configuring data synchronization with the SCIM Connector 320
Objects and operations supported by the SCIM Connector 321
Creating a SCIM connection with the SCIM Connector 322
Viewing or modifying the settings of a SCIM Connector 324
Configuring data synchronization with the Generic SCIM Connector 324
Configuring the Generic SCIM Connector for Starling Connect connections 325
Viewing or modifying the settings of a Generic SCIM Connector connection 330
Using connectors installed remotely 333
Installing Synchronization Service and built-in connectors remotely 333
Creating a connection using a remotely installed connector 334
Creating a connection 334
Renaming a connection 335
Deleting a connection 335
Modifying synchronization scope for a connection 335
Using connection handlers 336

Specifying password synchronization settings for a connection 338

Synchronizing identity data 340

Creating a sync workflow 341

Running a sync workflow 342

Running a sync workflow manually 342

Running a sync workflow on a recurring schedule 342

Disabling a sync workflow run schedule 343

Renaming a sync workflow 343

Deleting a sync workflow 344

Adding a creating step 344

Creating an update step 345

Creating a deprovisioning step 347

Modifying an existing sync workflow step 348

General Options 349

Source 349

Target 350

Creation Rules 350

Deprovisioning Rules 351

Updating Rules 352

Step Handlers 352

Deleting a sync workflow step 353

Changing the order of steps in a sync workflow 353

Generating object names by using rules 353

Modifying attribute values by using rules 355

Configuring a forward sync rule 355

Forward sync rule source item 356

Forward sync rule target item 356

Configuring a reverse sync rule 357

Reverse sync rule source item 357

Reverse sync rule target item 357

Configuring a merge sync rule 358

Using value generation rules 359

Configuring a rule entry 360

Using sync workflow step handlers 361

Example: Synchronizing group memberships 362

Example: Synchronizing multivalued attributes ... 362
Using sync workflow alerts ... 363
Creating or editing a sync workflow alert ... 363
Deleting a sync workflow alert ... 365
Managing outgoing mail profiles ... 365
Outgoing mail profile settings ... 366

Mapping objects ... 367
How to map objects ... 368
Creating mapping pairs ... 368
Creating mapping rules ... 369
Change scope for mapping rules ... 369
Running map operation ... 370
How to unmap objects ... 372

Automated password synchronization ... 373
How to automate password synchronization ... 374
Managing Capture Agent ... 375
Installing Capture Agent manually ... 375
Using Group Policy to install Capture Agent ... 376
Uninstalling Capture Agent ... 378
Managing password sync rules ... 378
Creating a password sync rule ... 379
Deleting a password sync rule ... 380
Modifying settings for a password sync rule ... 380
Fine-tuning automated password synchronization ... 381
Configuring Capture Agent ... 381
Creating and linking a Group Policy object ... 382
Adding an administrative template to Group Policy object ... 383
Using Group Policy object to modify Capture Agent settings ... 383
Modifying Synchronization Service parameters ... 383
Specifying a custom certificate for encrypting password sync traffic ... 385
Obtaining and installing a certificate ... 385
Exporting the custom certificate to a file ... 386
Importing certificate into certificates store ... 387
Copying the certificate's thumbprint ... 387

Providing the certificate's thumbprint to Capture Agent ... 388
Providing the certificate's thumbprint to Synchronization Service ... 389
Using PowerShell scripts with password synchronization ... 389

Synchronization history ... 391
Viewing sync workflow history ... 391
View mapping history ... 392
Searching synchronization history ... 393
Cleaning up synchronization history ... 394

Scenarios of use ... 395
Scenario: Create users from a .csv file to an Active Directory domain ... 396
Creating a sync workflow ... 396
Adding a creating step ... 397
Running the configured creating step ... 399
Committing changes to Active Directory ... 399
Scenario: Using a .csv file to update user accounts in an Active Directory domain ... 399
Creating an updating step ... 400
Running the configured updating step ... 401
Committing changes to Active Directory ... 401
Scenario: Synchronizing data between One Identity Manager Custom Target Systems and an Active Directory domain ... 401
Creating a connection to One Identity Manager ... 402
Configuring One Identity Manager modules, Custom Target System and Container Information ... 402
Creating a workflow for provisioning ... 403
Creating a provisioning step ... 403
Specifying synchronization rules ... 403
Running the workflow ... 404
Committing changes to One Identity Manager ... 404
Verify on One Identity Manager ... 404
Scenario: Deprovisioning between One Identity Manager Custom Target Systems and an Active Directory domain ... 405
Scenario: Provisioning of groups between One Identity Manager Custom Target Systems and an Active Directory domain ... 406
Scenario: Enabling Delta Sync mode between One Identity Manager Custom Target Systems and an Active Directory domain ... 407
Example of using the Generic SCIM Connector for data synchronization ... 408

Creating object mapping between a SCIM connection and an SQL connection 409
Creating a sync workflow for synchronizing data from a SCIM-based Starling Connect connector 412
Synchronizing complex multi-value objects from a SCIM source system 419

Appendix: Developing PowerShell scripts for attribute synchronization rules 423
Accessing source and target objects using built-in hash tables 423

Using PowerShell script to transform passwords 425
Accessing source object password 425

About us 427
Contacting us 427
Technical support resources 427

Synchronization Service overview

Within the same organization, identity information can be stored in many different data systems, such as directories, databases, or formatted dump files. Managing identity information and synchronizing it between these data systems can take a lot of time and effort for administrators. In addition, performing data synchronization manually is error-prone and can lead to duplicate information or incompatible data formats.

With Active Roles Synchronization Service, you can completely automate the process of identity data synchronization between the data systems used in your enterprise environment.

Synchronization Service increases data management efficiency by automating the creation, deprovision and update operations between your data systems. For example, if an employee joins or leaves the organization, Synchronization Service can automatically update the related information in all data systems, reducing your administrative workload and getting new users up and running faster.

The use of scripting capabilities provides a flexible way to:

- Automate day-to-day administration tasks.
- Integrate the administration of managed data systems with other business processes.

To start synchronizing identity data, you must connect Synchronization Service to your data systems with so-called "connectors". Connectors allow Synchronization Service to access specific data systems, then read and synchronize data in that system according to your settings.

Synchronization Service includes several built-in connectors that do not require any license file. For the list of these connectors and more information on configuring them, see External data systems supported with built-in connectors.

Synchronization Service features and benefits

Synchronization Service offers a wide range of features to synchronize identity data between your data systems.

Bidirectional synchronization

Bidirectional synchronization allows you to synchronize all changes occurred to identity information between your data systems. Using this type of synchronization, you can proactively prevent potential identity information conflicts between different data sources.

> NOTE: Bidirectional synchronization is unavailable for some of the supported data systems. For more information, see External data systems supported with built-in connectors.

Delta processing mode

Delta processing mode allows you to synchronize identities more quickly by processing only the data that has changed in the source and target connected systems since their last synchronization.

Both the full mode and the delta mode provide you with the flexibility of choosing the appropriate method for your synchronization tasks.

> NOTE: Delta processing mode is unavailable for some of the supported data systems. For more information, see External data systems supported with built-in connectors

Synchronization of group membership

Synchronization Service allows you to ensure that group membership information is in sync in all connected data systems. For example, when creating a group object from an Active Directory domain to an AD LDS (ADAM) instance, you can configure rules to synchronize the **Member** attribute from the Active Directory domain to the AD LDS (ADAM) instance.

Windows PowerShell scripting

The Management Shell component of Synchronization Service is an automation and scripting shell that provides a command-line management interface for synchronizing data between connected systems via the Synchronization Service.

The Management Shell is implemented as a Windows PowerShell snap-in that extends the standard Windows PowerShell functionality. The cmdlets provided by the Management Shell conform to the Windows PowerShell standards and are fully compatible with the default command-line tools that come with Windows PowerShell.

The Management Shell allows administrators to perform attribute or password synchronization operations by using Windows PowerShell scripts. For example, you can compose and run a Windows PowerShell script that assigns values to the target object attributes using the values of the source object attributes. For more information, see Using PowerShell script to transform passwords.

Attribute synchronization rules

With Synchronization Service, you can create and configure synchronization rules to generate values of target object attributes. These rules support the following types of synchronization:

- **Direct synchronization**: Assigns the value of a source object attribute to the target object attribute you specify.
- **Script-based synchronization**: Allows you to use a Windows PowerShell script to generate the target object attribute value.
- **Rule-based synchronization**: Allows you to create and use rules to generate the target object attribute value you want.

Rule-based generation of distinguished names

Synchronization Service lets you create flexible rules for generating the distinguished names (DNs) of objects being created. These rules allow you to ensure that created objects are named in full compliance with the naming conventions existing in your organization.

Scheduling capabilities

You can schedule running synchronization operations and automatically perform them on a regular basis to satisfy your company's policy and save time and effort.

Extensibility

To access external data systems, Synchronization Service employs special connectors. A connector allows Synchronization Service to read and synchronize the identity data contained in a particular data system. Out of the box, Synchronization Service includes connectors that allow you to connect to the following data systems:

- Microsoft Active Directory Domain Services
- Microsoft Active Directory Lightweight Directory Services
- Microsoft Exchange Server
- Microsoft Skype for Business Server
- Microsoft Azure Active Directory
- Microsoft 365

- Microsoft SQL Server
- Microsoft SharePoint
- Active Roles version 7.4.x, 7.3, 7.2, 7.1, 7.0, or 6.9
- One Identity Manager version 8.1, 8.0, or 7.0
- Data sources accessible through an OLE DB provider
- Delimited text files
- Generic LDAP Directory service
- MYSQL Database
- OpenLDAP Directory service
- Salesforce
- ServiceNow
- IBM DB2 Database
- IBM RACF Connector
- IBM AS/400 Connector
- Oracle Database connector
- Oracle Database User Accounts connector
- Micro Focus NetIQ Directory connector
- Oracle Unified Directory connector

Azure BackSync configuration

In any hybrid environment, on-premises Active Directory objects are synchronized to Azure AD, for example with Azure AD Connect. When Active Roles Synchronization Service is deployed in such a hybrid environment, to ensure data synchronization between the two systems, the existing user and group information (such as IDs) must be synchronized back from Azure AD to the on-premises AD deployment. To synchronize existing AD users and groups from Azure AD with Active Roles Synchronization Service, use the Azure back synchronization operation, known as "Azure BackSync".

For an Azure BackSync operation, you configure Active Roles Synchronization Service sync workflows to identify the unique Azure AD users or groups, then and map them to the on-premises AD users or groups. After the back synchronization operation is completed, Active Roles Synchronization Service displays the configured Azure attributes for the synchronized objects.

Azure BackSync allows you to configure the back synchronization operation in Azure with on-premises Active Directory objects through the Synchronization Service Console. The required connections, mappings, and sync workflow steps are created automatically.

When you configure back synchronization, the Azure application registration is done automatically with the default app `ActiveRoles_AutocreatedAzureBackSyncApp_V2`.

NOTE: Consider the following when configuring Azure BackSync:

- If you receive an `Application not found` error, try configuring back synchronization again later. The error may occur because Azure application synchronization may take some time.
- If you use existing back synchronization configuration settings, then the existing default app `ActiveRoles_AutocreatedAzureBackSyncApp` is used to run the back synchronization workflow. However, One Identity recommends using the default app `ActiveRoles_AutocreatedAzureBackSyncApp_V2` since it requires reduced administrator privileges. To use the latest Azure application, configure back synchronization again as described in Configuring Azure BackSync.
- To ensure that back synchronization works as expected, you must have:
  - Write permissions for `edsvaAzureOffice365Enabled`, `edsaAzureContactObjectId`, `edsvaAzureObjectID`, and `edsvaAzureAssociatedTenantId` attributes.
  - Local administrator privileges where Active Roles Synchronization Service is running.

Technical overview

The following illustration shows how Synchronization Service synchronizes data between connected data systems.

Figure 1: Synchronization of data between connected systems

Synchronization Service uses Capture Agents, connected data systems, connectors, connections, and sync workflows to synchronize identity data.

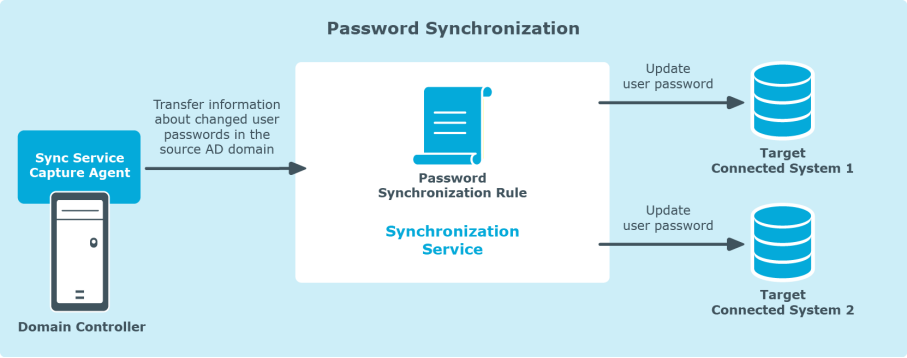
Synchronization Service

Synchronization Service performs data synchronization operations and include the Synchronization Service Console that provides a graphical user interface for managing connections to data systems and data synchronization operations.

Capture Agent

Synchronization Service Capture Agent allows you to synchronize user passwords between Active Directory domains managed by Synchronization Service and other connected data systems. The following diagram shows how password synchronization works with Synchronization Service Capture Agent:

Figure 2: Password synchronization



Capture Agent tracks changes to user passwords in the source Active Directory domain and provides that information to Synchronization Service, which then synchronizes the changes to the target connected data systems by using the password synchronization rules you specified.

To synchronize passwords, install Capture Agent on each domain controller in the Active Directory domain you want to use as a source for the password synchronization operations.

Connectors and connected data systems

Synchronization Service lets you synchronize identity information between a wide variety of external data systems. To synchronize identities, you must connect Synchronization Service to your data systems through special connectors. A connector enables Synchronization Service to access a specific data system and read and synchronize identity data in that system.

For the list of supported data systems, see Extensibility.

Sync workflows

A sync workflow is a set of synchronization steps (or synchronization operations) that define how to synchronize objects between two connected data systems. A sync workflow can comprise one or more synchronization steps. You can use the Synchronization Service Console, a component of Synchronization Service, to configure as many sync workflows as needed.

You can configure a synchronization step to perform one of the following operations:

- **Creation**: Creates objects in the target connected data systems based on the changes made to specific objects in the source connected system. When creating a new object, Synchronization Service assigns initial values to the object attributes based on the attribute population rules you have configured.
- **Update**: Changes the attributes of objects in the target connected data systems based on the changes made to specific objects in the source connected system. To define the objects that will participate in the update operation you can use object mapping rules. For more information, see Mapping objects.
- **Deprovision**: Modifies or removes objects in the target connected data systems after their counterparts have been disconnected from the source connected system. Synchronization Service can be configured to remove objects permanently or change them to a specific state.

2

Deploying Synchronization Service

This section describes how to:

- Install and configure Active Roles Synchronization Service.
- Configure Azure BackSync.
- Upgrade from supported versions of One Identity Quick Connect.

It also lists the communication ports used by Synchronization Service.

Installing Synchronization Service

To install all features and components of Active Roles Synchronization Service, use the installation media downloaded from the One Identity Support Portal. Alternatively, you can also install the Synchronization Service Management Shell only.

Prerequisites

Make sure the system on which you want to install Synchronization Service meets the system requirements described in the *Active Roles Synchronization Service Release Notes*.

NOTE:

After installing a supported version of the Az.Accounts PowerShell module, to prevent the module from using a non-supported authentication method later, run the following command in Windows PowerShell:

```
Update-AzConfig -EnableLoginByWam $false
```

Running this command ensures that the module uses an authentication method supported by Synchronization Service, even if the Az.Accounts module is updated later to a newer version (for example, because of upgrading another PowerShell module) that uses a newer, unsupported authentication method by default.

To install Synchronization Service and all its components

1. From the Active Roles Synchronization Service installation package, run the Active Roles Synchronization Service setup.
2. Follow the instructions in the setup wizard.
3. On the **Ready to Install** page, click **Install**. The wizard will then install the following components:
   - Synchronization Service Console: The graphical user interface of Active Roles Synchronization Service.
   - Management Shell: A command-line interface to synchronize data between external data systems with Active Roles Synchronization Service. For more information, see Synchronization Service Management Shell.
   - All built-in connectors to connect Synchronization Service to external data systems.
4. To exit the wizard, click **Finish**.

To install Synchronization Service Management Shell only

1. In Windows Explorer, navigate to the following folder of the installation media:

   `\Components\ActiveRoles Synchronization Service`
2. To open the Windows command prompt, click the navigation bar of Windows Explorer, enter **`cmd`**, then press **Enter**.
3. To install Synchronization Service Management Shell only, enter the following command, then press **Enter**:

   `SyncService.msi INSTALLSYNCSHELL=1`

   The installer then silently installs Synchronization Service Management Shell.
4. To check if Management Shell has finished installation, search the application either in the Windows Start Menu, or in the **Apps & Features** list of the operating system. After the setup finished the installation, Management Shell will appear in these lists.

   To uninstall, navigate to **Add or remove programs**, click Active Roles Synchronization Service Management Shell, then click **Uninstall**.

   > NOTE: Running the Active Roles Synchronization Service installation wizard with the `.exe` file of the installation media always installs both the Synchronization Service Console and the Synchronization Service Management Shell.
   >
   > One Identity recommends using the installation wizard to install both the Synchronization Service Console and the Synchronization ServiceManagement Shell for most use cases.

Configuring Synchronization Service

To configure Synchronization Service, you can use one of the following methods:

- Specify new SQL Server or Azure SQL Server databases for storing the Synchronization Service data.

  With this method, you can store the configuration settings and synchronization data either in a single new SQL Server database or in two separate databases.

- Share existing configuration settings between two or more instances of Synchronization Service.

Prerequisites

- If you are using an Azure SQL Server, set the **`db_owner`** database role to the user of the Azure SQL Server.
- If you are using an SQL Server, set the **`dbcreator`** server role to the user of the SQL Server.

  `dbcreator` is the minimum role that the user of the SQL Server or Azure SQL Server requires for the initial configuration of Synchronization Service.

  After creating the new database, you can revoke the **`dbcreator`** role because the **`db_owner`** role that is automatically assigned to the same user of the SQL Server is sufficient for the Synchronization Service database connection.

To configure Synchronization Service using a new database

1. Start the Synchronization Service Console.
2. Follow the steps in the wizard that starts automatically to configure Synchronization Service.
3. On the **Service Account and Mode** page, specify the following and click **Next**:
   - The account under which you want Synchronization Service to run.
   - The mode (local or remote) in which you want to use Synchronization Service. Use the remote mode to work with connectors installed remotely. For more information, see Using connectors installed remotely. If you select the remote mode, click **Finish** to close the wizard.
4. Select **Create a new configuration** and click **Next**.
5. On the **Database Connection** page, specify an SQL Server database.
   - **SQL Server**: Enter the name of the SQL Server computer that hosts the database you want to participate in data synchronization operations.
   - **Database**: Enter a name for the new SQL Server database.
6. (Optional) Select **Store sync data in a separate database**.
   - If you want to store the configuration settings and synchronization data in a single SQL Server database, clear the check box.
   - If you want to store the configuration settings and synchronization data in two separate databases, select the check box, then specify the database in which you want to store the synchronization data.

7. On the **Database Connection** page, select an SQL Server authentication method, and click **Next**.

   NOTE: For all Azure SQL Server variants, select **Use SQL Server authentication** because Windows authentication is not supported.

   - **Use Windows authentication**: Allows you to access the SQL Server in the security context of the account under which the Synchronization Service is running.
   - **Use SQL Server authentication**: Allows you to access the SQL Server in the security context of the SQL Server user account whose user name and password you specify.

8. On the **Configuration File** page, select the file for storing the created configuration profile, protect the file with a password, and click **Finish**.

To configure Synchronization Service using an existing database

1. Start the Synchronization Service Console.
2. Follow the steps in the wizard that starts automatically to configure Synchronization Service.
3. On the **Service Account and Mode** page, specify the following and click **Next**:
   - The account under which you want Synchronization Service to run.
   - The mode (local or remote) in which you want to use Synchronization Service. Use the remote mode to work with connectors installed remotely. For more information, see Using connectors installed remotely. If you select the remote mode, click **Finish** to close the wizard.
4. Select **Use an existing configuration** and click **Next**.

   NOTE: If the Synchronization Service is already configured, using an existing configuration file does not override the existing SQL Server or Azure SQL Server database settings. To change the settings of the database, you must reconfigure it or reinstall the Synchronization Service with the new configuration.

5. On the **Configuration File** page, select **I have the configuration file** to provide the configuration file you exported from an existing Synchronization Service instance, enter the password if necessary, and click **Next**. If you do not have the configuration file, after clicking **Next** you will need to enter the required settings.
6. If you provided the configuration file, specify the authentication method for accessing the database. Otherwise, enter the required database name and select the authentication method. Click **Finish**.

After you configure Synchronization Service, you can change its settings at any time using the **Configuration Wizard**. To start the wizard, start the Synchronization Service Console and click the gear icon in the upper right corner of the Synchronization Service Console.

Configuring Azure BackSync

In hybrid environments, on-premises Active Directory (AD) objects are synchronized to Azure AD, for example via Azure AD Connect. When you deploy Active Roles Synchronization Service in such a hybrid environment, this synchronization works only if existing user and group information (such as the Id) are also synchronized back from Azure AD to the on-premises AD. Active Roles Synchronization Service uses Azure back-synchronization (also known as Azure BackSync) for this purpose.

Prerequisites

The hybrid environment must meet the following requirements to configure Azure BackSync:

- Azure Active Directory (Azure AD) module version 2.0.0.131 or later must be installed and configured.
- The Directory Writers role must be enabled in Azure AD. To enable the role, use the following script:

```
$psCred=Get-Credential
Connect-AzureAD -Credential $psCred
$roleTemplate = Get-AzureADDirectoryRoleTemplate | ? { $_.DisplayName -eq
"Directory Writers" }

# Enable an instance of the DirectoryRole template

Enable-AzureADDirectoryRole -RoleTemplateId $roleTemplate.ObjectId
```

In addition, the user account you use to configure Azure BackSync must have the following roles:

- User Administrator
- Exchange Administrator
- Application Administrator

Automatic and Manual Azure BackSync

You can perform Azure back-synchronization with Active Roles Synchronization Service, either automatically or manually:

- You can configure automatic Azure back-synchronization via the **(Settings)** > **Configure Azure BackSync** option of Active Roles Synchronization Service. For more information, see Configuring automatic Azure BackSync.
- You can also configure manual Azure back synchronization, using existing Active Roles Synchronization Service feature components. For more information, see Configuring manual Azure BackSync.

Configuring manual Azure BackSync

You can configure manual Azure back synchronization (Azure BackSync) by using the existing features of Active Roles Synchronization Service components. When setting up manual Azure BackSync, you must configure sync workflow to identify Azure AD-specific users or groups, and to map them to the corresponding on-premises Active Directory (AD) users or groups. After a manual Azure BackSync operation is completed, Active Roles Synchronization Service will display the configured Azure attributes for the synchronized objects.

For more information on setting up automatic Azure back-synchronization, see Configuring automatic Azure BackSync.

Prerequisites

The hybrid environment must meet the following requirements to configure Azure BackSync manually:

- Azure Active Directory (Azure AD) module version 2.0.0.131 or later must be installed and configured.
- You must authenticate the Azure tenant of the Azure AD for which you configure back-synchronization. Also, you must consent Active Roles as an Azure application.

  For more information, see *Configuring Active Roles to manage Azure AD using the GUI* in the *Active Roles Administration Guide*.
- For the container where Active Roles Synchronization Service performs back-synchronization, you must enforce the built-in Azure AD policy that automatically sets the attribute `edsvaazureOffice365enabled` to **`true`**.
- Your Active Roles Synchronization Service user must have write permissions for the following attributes:
  - `edsvaAzureOffice365Enabled`
  - `edsaAzureContactObjectId`
  - `edsvaAzureObjectID`
  - `edsvaAzureAssociatedTenantId`
- Your Active Roles Synchronization Service user must also have local administrator privileges on the machine where Active Roles Synchronization Service is running.

To configure a manual Azure BackSync workflow

1. Create a connection to Azure AD using the Azure AD Connector. The configuration requires the following data:
   - The Azure domain name.
   - The Client ID in Azure AD.
   - The Client Key to establish the connection to Azure AD.
2. Create an Azure application (or use any relevant existing Azure application) under the Azure tenant of your Azure AD. The application must have application permissions to read and write directory data in Azure AD.

   TIP: You can assign the required permissions to the application by running a Windows PowerShell script. For more information, see Creating a Microsoft Azure Active Directory connection.
3. Open the application properties and copy the following:
   - Client ID
   - The valid Client Key of the application.
4. Use the Client ID and Client Key when creating a new Azure AD connection or modifying an existing one. For more information, see Creating a Microsoft Azure Active Directory connection.

   NOTE: Two applications are required for Azure BackSync operations:
   - The Web Application that you created in this step, or is already available for the Synchronization Service Azure AD Connector.
   - An Azure application that you created while configuring Azure AD in the Active Roles Administration Service.

     For details, see *Configuring Active Roles to manage Azure AD using the GUI* in the *Active Roles Administration Guide*).

   Both applications are required for Azure BackSync operations.
5. Create a connection to Active Roles using the **Active Roles Connector**. The configuration requires the local domain details and the version of Active Roles you use. Define the scope to select the container from which Active Roles Synchronization Service will select the objects for synchronization.
6. In the Active Roles Synchronization Service, create a new sync workflow with **Sync Workflows** > **Add sync workflow**. Use the Azure AD and Active Roles connections configured previously, and add a synchronization step to synchronize the Azure AD users or groups with the on-premises users or groups in Active Roles Synchronization Service.
7. In the on-premises Active Roles users or groups, set the `edsvaAzureAssociatedTenantIdattribute` attribute to the value of the Azure tenant ID.

   NOTE: If you did not configure `edsvaAzureAssociatedTenantIdattribute`, an error will be logged for each object in the Event Viewer.

8. Configure the **Forward Sync Rule** to synchronize the following:
    - The Azure Object ID property of the Azure AD user or group to the **edsvaAzureObjectID** property of the corresponding on-premises Active Roles user or group.
    - Set the **edsvaAzureOffice365Enabled** attribute in the on-premises Active Roles user or group to `true`.
    - Set the **edsvaAzureAssociatedTenantId** attribute to the value of the Azure tenant ID.
9. Create a **Mapping Rule**. A mapping rule has two functions:
    - It uniquely identifies the synchronized users or groups both in Azure AD in the on-premises AD.
    - It maps the specified properties from Azure AD to Active Roles Synchronization Service appropriately.

    For example, the property **userprincipalname** can be used to map users between the on-premises AD and Azure AD in a federated environment.

    CAUTION: Based on the environment, make sure to create the correct mapping rule to identify the user or group uniquely. Incorrect mapping rules may create duplicate objects, resulting in Azure BackSync not working as expected.

    NOTE: Consider the following when configuring manual Azure back synchronization:
    - You must perform the initial configuration and back synchronization of Azure AD user IDs only once.
    - Azure AD groups cannot be created in Federated or Synchronized environments. Instead, Azure AD groups are created in Active Roles Synchronization Service and are synchronized to Azure AD using native Microsoft tools, such as Azure AD Connect. To manage the Azure AD group through Active Roles Synchronization Service, you must perform periodic back synchronization to the on-premises AD.

Configuring automatic Azure BackSync

You can configure automatic Azure BackSync via the **(Settings)** > **Configure Azure BackSync** option of Active Roles Synchronization Service Console. After you finish configuration, the Synchronization Service Console will automatically create the Azure BackSync registration, its required connections, mappings and workflows.

For more information on setting up manual Azure BackSync, see Configuring automatic Azure BackSync.

To configure an automatic Azure BackSync workflow in Active Roles Synchronization Service

1. To open the **Configure BackSync operation in Azure with on-prem Active Directory objects** window of Synchronization Service Console, click **(Settings)** > **Configure Azure BackSync**.
2. Select one of the following options based on the number of Azure AD services in your Azure tenant:
   - **I have one Azure AD in my Azure tenant.**
   - **I have more than one Azure AD in my Azure tenant.**
3. Authenticate your access to Azure AD:
   - If you have selected **I have one Azure AD in my Azure tenant**, to authenticate your access to Azure AD, click **Log in to Azure**, and from the **Select Environment Type** drop-down, select the environment type of your Azure tenant.

     NOTE: Active Roles Synchronization Service supports Azure Cloud, Azure GCC and Azure GCC-H government tenants.
   - If you have selected **I have more than one Azure AD in my Azure tenant**, in **Tenant ID**, enter the GUID of the Azure AD for which you want to set up synchronization.

     TIP: For more information on how to find the GUID of an Azure AD service, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.

     After specifying the tenant ID, to authenticate your access to Azure AD, click **Log in to Azure**, and in the **Select Environment Type** drop-down, select the environment of your Azure tenant.

     NOTE: If you select **I have more than one Azure AD in my Azure tenant**, the **Log in to Azure** button will be enabled only if you specify a well-formed Azure AD GUID in the **Tenant ID** text box.
4. Under **Connect to**, specify the domain name of the computer where Active Roles Synchronization Service is running.
5. Select the validation method used to access Active Roles Administration Service. Depending on how Active Roles Synchronization Service has been deployed in your organization, you can either use **Synchronization Service account** or **Windows account**-based validation. If you have selected **Windows account** authentication, enter your Windows user name and password.
6. To test the configured Active Roles Synchronization Service connection, click **Test Active Roles Connection**.
7. To apply your changes, click **Configure BackSync**.

   NOTE: If the Azure BackSync settings have already been configured previously, Synchronization Service Console will display a warning message to confirm if you want to override the existing Azure BackSync settings with the new settings.

- To override the existing settings, click **Override BackSync Settings**.
- To keep the existing settings, click **Cancel**.

8. An **Application Consent** dialog will appear, prompting you for authentication. To consent Active Roles Synchronization Service, click **OK**.

   Synchronization Service Console will automatically register the Azure application, and it will also create the required connections, mappings, and workflow steps for back synchronization. For more information on the automatically created Azure BackSync settings, see Settings updated after Azure BackSync configuration operation.

9. To make the new Azure BackSync workflow appear under **Sync Workflows**, close and reopen Synchronization Service Console. The new Azure BackSync workflow will appear with the following default name: **AutoCreated_ AzureADBackSyncWorkFlow_<tenant-name>**.

Settings updated after Azure BackSync configuration operation

This section provides descriptions about the Azure App registration, connections, mappings, and workflow steps that are created automatically as a result of the Azure BackSync configuration operation.

Azure App registration

The Azure App is created automatically with the default name as `ActiveRoles AutocreatedAzureBackSyncApp_V2`.

NOTE: After the Azure App is registered in Azure, you must not delete or modify the application. The back synchronization operation will not work as expected in case you modify or delete the registered Azure App.

Sync workflows

On the Synchronization Service Console, click **Sync Workflows** to view the sync workflow named `AutoCreated_AzureADBackSyncWorkflow_<tenant name>` that is created as a result of the Azure BackSync configuration. The workflow displays the following synchronization update steps from Azure AD to Active Roles for users, groups, and contacts:

- **Step 1: AutoCreated_UpdateFromAzureToARSForBackSyncWorkFlowUser_ <tenant>** for users.
- **Step 2: AutoCreated_ UpdateFromAzureToARSForBackSyncWorkFlowGroup_<tenant>** for groups.
- **Step 3: AutoCreated_ UpdateFromO365ToARSForBackSyncWorkFlowContact_<tenant>** for contacts.

NOTE: Consider the following:

- Multiple tenants are supported in back synchronization. The workflows can be identified using the name of the tenant.
- The **Forward Sync Rules** to synchronize the following are automatically configured and displayed in the synchronization update steps for users and groups:
  - The Azure **ObjectID** property of a user or group is mapped to the Active Roles user or group **edsvaAzureObjectID** property.
  - The **edsvaAzureOffice365Enabled** attribute in the Active Roles user or group is set to `True`.
  - The **edsvaAzureAssociatedTenantId** attribute in the Active Roles user or group is set to the value of the Azure tenant ID.
- The **Forward Sync Rule** to synchronize the following are automatically configured and displayed in the synchronization update steps for contacts:
  - Azure **ExternalDirectoryObjectID** property of a contact is mapped to the Active Roles contact **edsaAzureContactObjectId** property.
  - The **edsvaAzureOffice365Enabled** attribute in the Active Roles user or group is set to `True`.
  - The **edsvaAzureAssociatedTenantId** attribute in the Active Roles user or group is set to the value of the Azure tenant ID.

Connections

On the Synchronization Service Console, click **Connections** to view the connections from Active Roles Synchronization Service, Azure AD, and Microsoft 365 to external data systems. The following connections are configured and displayed by default:

- **AutoCreated_ARSConnectorForBackSyncWorkFlow_<tenant>**
- **AutoCreated_AzureADConnectorForBackSyncWorkFlow_<tenant>**
- **AutoCreated_O365ConnectorForBackSyncWorkFlow_<tenant>**

NOTE: Multiple tenants are supported in back synchronization. The connection name can be identified using the name of the tenant.

Mapping

On the Synchronization Service Console, click **Mapping** to view the mapping rules which identify the users, groups, or contacts in Azure AD and on-premises AD uniquely and map the specified properties from Azure AD to Active Roles Synchronization Service appropriately.

On the **Mapping** tab, click a connection name to view or modify the mapping settings for the corresponding connection. The user, group, and contact mapping pair information is displayed by default as a result of the Azure BackSync configuration. For example, the property **userprincipalname** can be used to map users between on-premises AD and Azure AD in a federated environment.

NOTE: Consider the following when working with mapping rules:

- For more information to manage mapping pairs for the connections see Change scope for mapping rules.
- The mapping rules are created by default. Based on the environment, make sure that the default mapping rules identify the user or group uniquely. Otherwise, make sure to correct the mapping rule as required. Incorrect mapping rules may create duplicate objects and the back synchronization operation may not work as expected.
- Initial configuration and running of back synchronization operation for Azure AD users ID and group ID is a one-time activity. If required, you can reconfigure the Azure BackSync settings, which will override the previously configured back synchronization settings.

Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync

If the Azure tenant of your organization contains multiple Azure AD services, One Identity highly recommends to specify its GUID (also known as Tenant ID) when configuring Azure BackSync automatically.

For details on configuring Azure BackSync automatically, see Configuring automatic Azure BackSync.

The GUID of each Azure AD service is listed on the Microsoft Azure Portal.

To find the GUID (Tenant ID) of an Azure AD

1. Log in to the Microsoft Azure Portal.
2. Click **Show portal menu**.
3. Click **Azure Active Directory**.
4. In the **Overview** tab, under the **Basic information** heading, the value of the **Tenant ID** is the GUID (Tenant ID) of the Azure AD.

   TIP: If you have access to multiple Azure AD services, you can switch between them with **Manage tenants**.

Upgrade from Quick Connect and Synchronization Service

If you have sync workflows configured and run by Quick Connect (the predecessor of Synchronization Service), or earlier versions of Active Roles Synchronization Service, then you can transfer those sync workflows to the current version of Active Roles Synchronization Service.

You can transfer sync workflows from the following Quick Connect or Active Roles Synchronization Service versions:

- Quick Connect for Active Directory 6.1
- Quick Connect for AS400 1.4
- Quick Connect for Base Systems 2.4
- Quick Connect for Cloud Services 3.7
- Quick Connect for RACF 1.3
- Quick Connect Sync Engine 5.5 and 6.1
- Synchronization Service 7.5 and later

For more information, see *Transferring sync workflows from Quick Connect* in the *Active Roles Synchronization Service Administration Guide*.

Transferring sync workflows from Quick Connect

To transfer sync workflows from Quick Connect to Synchronization Service

1. Install Synchronization Service.

   You can install Synchronization Service on the computer running Quick Connect or on a different computer. For installation instructions, see Installing Synchronization Service.

2. Configure Synchronization Service to use a new database for storing configuration settings and synchronization data.

   To perform this step, use the **Configuration Wizard** that appears when you start the Synchronization Service Console the first time after you install Synchronization Service. For more information, see Configuring Synchronization Service.

3. Import configuration settings from Quick Connect or Synchronization Service.

   Before you proceed with this step, it is highly recommended to disable the scheduled workflows and mapping operations in Quick Connect or earlier versions of Synchronization Service. You can resume the scheduled workflows and mapping operations after you complete this step.

   To import configuration settings:

   a. On the computer where you have installed Synchronization Service, start the Synchronization Service Console.

   b. In the upper right corner of the Active Roles Synchronization Service window, click the gear icon, and then click **Import Configuration**.

c. In the wizard that appears, select the version of Quick Connect Sync Engine used by your Quick Connect version or Active Roles Synchronization Service from which you want to import the configuration settings.

   Optionally, you can select the **Import sync history** check box to import the sync history along with the configuration settings.

d. Follow the steps in the wizard to complete the import operation.

If the synchronization data you want to import is stored separately from the configuration settings, then, on the **Specify source SQL Server databases** step, select the **Import sync data from the specified database** check box, and specify the database.

4. Retype access passwords in the connections that were imported from Quick Connect.

   NOTE: Re-entering passwords in the imported connections is required because due to security reasons, the configuration import process does not retrieve encrypted passwords from Quick Connect. To modify the imported connections later, use the Synchronization Service Console. For more information, see External data systems supported with built-in connectors.

5. If your sync workflows involve synchronization of passwords, then you need to install the new version of Capture Agent on your domain controllers. For installation instructions, see Managing Capture Agent.

   The new version of Capture Agent replaces the old version. However, as the new version supports both Synchronization Service and Quick Connect, you do not lose the password synchronization functions of Quick Connect after you upgrade Capture Agent.

Communication ports used by Synchronization Service

Active Roles Synchronization Service uses the following default communication ports. To make sure that the specific traffic type works as intended, open the following ports on the machine running Active Roles Synchronization Service.

For more information on opening ports, see the instructions of the **Windows Defender Firewall with Advanced Security** console of your operating system, or the documentation of your network device.

Port required for Synchronization Service traffic

- Port **15173** (HTTPS), TCP, Inbound.

  NOTE: This port is also used by Capture Agent to communicate with Active Roles Synchronization Service. If you use Capture Agent, open this port on the domain controller (DC) where Capture Agent is installed.

Port required for DNS traffic

- Port **53**, TCP/UDP, Inbound / Outbound.

Port required for Kerberos traffic

- Port **88**, TCP/UDP, Inbound / Outbound.

Ports required for SMB / CIFS traffic

- Port **139**, TCP, Inbound / Outbound.
- Port **445**, TCP, Inbound / Outbound.

Ports required for LDAP traffic

- Port **389**, TCP / UDP, Outbound.
- Port **3268**, TCP, Outbound.

Ports required for SSL traffic

- **636**, TCP, Outbound.
- **3269**, TCP, Outbound.

> NOTE: This port is only required if Synchronization Service is configured to use SSL to connect to an Active Directory domain.

Ports required for Active Roles Capture Agent traffic

If Synchronization Service is configured to synchronize user passwords from an Active Directory domain to other connected data systems, open the following port on the DC where the Synchronization Service Capture Agent is installed.

- **7148** (HTTPS), TCP, Inbound.

Port required for RPC endpoint mapper traffic

- **135**, TCP, Inbound / Outbound.

  > NOTE: Port 135 is a dynamically allocated TCP port for RPC communication with Active Directory DCs. For more information about ports used for RPC communication, see the following Microsoft Support Knowledge Base articles at support.microsoft.com:
  >
  > - How to restrict Active Directory RPC traffic to a specific port (Original KB number: 224196).
  > - How to configure RPC dynamic port allocation to work with firewalls (Original KB number: 154596).

- How to configure RPC to use certain ports and how to help secure those ports by using IPsec (Original KB number: 908472).
- The default dynamic port range for TCP/IP has changed in Windows Vista and in Windows Server 2008 (Original KB number: 929851).

Deploying Synchronization Service for use with AWS Managed Microsoft AD

NOTE: This feature is officially supported starting from Active Roles Synchronization Service 8.1.3 SP1 (build 8.1.3.10). It is not supported on Active Roles Synchronization Service 8.1.3 (build 8.1.3.2) and earlier versions.

Active Roles Synchronization Service supports deployment and configuration in the Amazon cloud to manage AWS Managed Microsoft AD object synchronization.

This allows you to:

- Synchronize directory data from an on-premises AD environment to AWS Managed Microsoft AD.
- Synchronize passwords from an on-premises Active Directory to AWS Managed Microsoft AD (with certain limitations).

Supported AWS Managed Microsoft AD deployment configuration

To synchronize data to and from AWS Managed Microsoft AD, you must deploy Active Roles Synchronization Service in Amazon Web Services (AWS) in the following configuration:

- Active Roles Synchronization Service must be deployed on an Amazon Elastic Compute Cloud (EC2) instance or instances. For more information, see the *Amazon Elastic Compute Cloud documentation*.
- The SQL Server required by Active Roles Synchronization Service must run on a separate Amazon Relational Database Service for Microsoft SQL Server (RDS for SQL Server) instance. For more information, see the *Amazon RDS documentation*.
- The Active Directory environment must be hosted in AWS via AWS Directory Service. For more information, see the *AWS Directory Service documentation*.

NOTE: Support for AWS Managed Microsoft AD by Active Roles Synchronization Service was tested only in this configuration. Active Roles Synchronization Service does not officially support managing AWS Managed Microsoft AD environments in a hybrid deployment, that is, using an on-premises Active Roles Synchronization Service and/or SQL Server installation and hosting AD via AWS Directory Service.

Synchronization Service features and limitations when used with AWS Managed Microsoft AD

If configured to manage AWS Managed Microsoft AD in the Amazon cloud, Active Roles Synchronization Service offers the following features:

- Synchronization Service connections and sync workflows based on the following Active Roles Synchronization Service connectors:
  - Active Directory Connector
  - Active Roles Connector
  - Delimited Text File Connector
- Synchronizing passwords with Active Roles Synchronization Service from on-premises AD to AWS Managed Microsoft AD.

However, when using Synchronization Service in an EC2 instance in the Amazon cloud, also consider the following limitations.

Amazon Web Services limitations

For Active Roles Synchronization Service installations deployed in Amazon Elastic Compute Cloud (EC2) instances and SQL Servers hosted on Amazon Relational Database Service for SQL Server (RDS for SQL Server) instances, the known EC2 and RDS limitations apply.

- For more information about the known EC2 limitations, see Launch template restrictions, Hibernation limitations and (if applicable) Constraints on the size and configuration of an EBS volume in the *Amazon EC2 documentation*.
- For more information about the known Amazon RDS limitations, see Quotas and constraints in the *Amazon RDS documentation*.

Synchronization Service limitations

- When synchronizing directory data or passwords from on-premises Active Directory to AWS Managed Microsoft AD, Active Roles Synchronization Service has the following limitations:
  - Active Roles Synchronization Service was only tested to work with connections and sync workflows based on the following connectors:
    - Active Directory Connector
    - Active Roles Connector
    - Delimited Text File Connector

Sync workflows and connections based on other connectors are not officially supported.

- When synchronizing passwords from an on-premises Active Directory to AWS Managed Microsoft AD, synchronizing the `pwdHash` attribute and synchronizing then populating the `SIDHistory` attribute to AWS Managed Microsoft AD is not supported. This is because the Synchronization Service Capture Agent cannot be installed in an AWS Managed Microsoft AD environment.

- Synchronizing passwords from AWS Managed Microsoft AD to on-premises AD with Active Roles Synchronization Service is not supported. This is because the Synchronization Service Capture Agent cannot be installed in an AWS Managed Microsoft AD environment.

Main steps of configuring Active Roles Synchronization Service for AWS Managed Microsoft AD

If your organization and environment meet the Deployment requirements for AWS Managed Microsoft AD support, configuring Active Roles Synchronization Service for managing AWS Managed Microsoft AD via AWS Directory Service has the following main steps:

1. Creating your AWS Managed Microsoft AD environment.
2. Creating an Amazon Elastic Compute Cloud (EC2) instance for Active Roles Synchronization Service.
3. Joining the EC2 instance to AWS Managed Microsoft AD.
4. Creating an Amazon Relational Database Service for SQL Server (RDS for SQL Server) instance to host the Active Roles Synchronization Service database.
5. Verifying the connectivity between the EC2 and RDS instances.
6. Installing and configuring Active Roles Synchronization Service on the EC2 instance.

Deployment requirements for AWS Managed Microsoft AD support

Before starting the deployment and configuration of Active Roles Synchronization Service to manage AWS Managed Microsoft AD via AWS Directory Service, make sure that the following requirements are met.

NOTE: When setting up a virtual environment, carefully consider the configuration aspects such as CPU, memory availability, I/O subsystem, and network infrastructure to ensure the virtual layer has the necessary resources available. For more information

about environment virtualization, see One Identity's Product Support Policies.

Connectivity requirements

You must have:

- Stable network connectivity to Amazon Web Services (AWS).
- Port **1433** open and available for the Amazon Relational Database Service (RDS) service.
- Access to the AWS service with the **AWSAdministratorAccess** permission.

  NOTE: Make sure that you have **AWSAdministratorAccess** permission, as it is required for certain configuration steps. The **AWSPowerUserAccess** permission is not sufficient for completing the entire configuration procedure.

Infrastructure requirements

To deploy and configure Active Roles Synchronization Service for AWS Managed Microsoft AD, you must have access to the following AWS services and resources:

- AWS Managed Microsoft AD deployed via AWS Directory Service.
- One or more Amazon Elastic Compute Cloud (EC2) instance(s) hosting the Active Roles Synchronization Service services and components.

  The EC2 instance(s) must have, at minimum:

  - 2 vCPUs running at 2.0 GHz.
  - 4 GB of RAM.

  NOTE: AWS Managed Microsoft AD support was tested with a single **t2.large** EC2 instance.
- An Amazon Relational Database Service for SQL Server (RDS for SQL Server).

  NOTE: AWS Managed Microsoft AD support was tested with an RDS instance running the latest version of Microsoft SQL Server.

Make sure that all these components are discoverable or visible to each other.

Creating the AWS Managed Microsoft AD instance

To deploy and configure Active Roles Synchronization Service in Amazon Web Services (AWS) for managing AWS Managed Microsoft AD, first you must create an AWS Directory Service instance hosting your AWS Managed Microsoft AD instance in the AWS console. For more information on configuring the service in the AWS console, see Setting up AWS Directory Service in the *AWS Directory Service documentation*.

NOTE: Consider the following when creating the AWS Managed Microsoft AD instance:

- Make sure that the connectivity requirements listed in Deployment requirements for AWS Managed Microsoft AD support are met.
- During the procedure, take note of the following values, as they will be required in later procedures:
  - **Directory DNS name**: The fully qualified domain name (FQDN) of your AD service (for example, `activeroles.demo`).
  - **Directory NetBIOS name**: The NetBIOS name (or shortname) of your AD service (for example, `ARDEMO`).
  - **Admin password**: The password of the default admin account (named `admin`).
- After specifying all required settings, it takes approximately 30-40 minutes to create the AWS Managed Microsoft AD instance. If you run into any issues when creating the environment, see Troubleshooting AWS Managed Microsoft AD in the *AWS Managed Microsoft AD documentation*.

Creating the EC2 instance for Active Roles Synchronization Service

To deploy and configure Active Roles Synchronization Service in Amazon Web Services (AWS) for managing AWS Managed Microsoft AD, you must create an Amazon Elastic Compute Cloud (EC2) instance hosting your Active Roles Synchronization Service installation.

Complete the procedure in AWS as described in Set up to use Amazon EC2 in the *Amazon EC2 documentation*. If you run into any problems when configuring or connecting to the EC2 instance, see Troubleshoot EC2 Windows instances in the *Amazon EC2 documentation*.

NOTE: Consider the following when creating the EC2 instance:

- Make sure that the connectivity requirements listed in Deployment requirements for AWS Managed Microsoft AD support are met.
- For the operating system on the EC2 instance, select a **Microsoft Windows Server** AMI supported by Active Roles Synchronization Service. For the list of supported Windows Server operating systems, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.
- Select an EC2 instance type that has, at minimum:
  - 2 vCPUs running at 2.0 GHz.
  - 4 GB of RAM.
- One Identity recommends setting the storage to a minimum of 60 GiB of gp2 root volume.

TIP: For consistency, after you logged in to the EC2 instance, rename the virtual machine to the same name that you originally defined for the EC2 instance in the AWS console.

Joining the EC2 instance to AWS Managed Microsoft AD

After you created your AWS Managed Microsoft AD service and your EC2 instance(s), you must join the configured Amazon Elastic Compute Cloud (EC2) instance(s) to AWS Managed Microsoft AD.

Complete the procedure in Amazon Web Services (AWS) as described in Join an EC2 instance to your AWS Managed Microsoft AD directory in the *AWS Directory Service documentation*.

NOTE: Consider the following when joining the EC2 instance(s) to AWS Managed Microsoft AD:

- Make sure that the connectivity requirements listed in Deployment requirements for AWS Managed Microsoft AD support are met.
- You need to use the fully qualified domain name that your configured during Creating the AWS Managed Microsoft AD instance.

TIP: If the domain join process ends with an error, check the specified DNS addresses and Domain Admin credentials in the AWS console.

Creating the RDS instance for the Active Roles Synchronization Service SQL Server

If you manage AWS Managed Microsoft AD with Active Roles Synchronization Service in Amazon Web Services (AWS), you must store the Synchronization Service database in an Amazon Relational Database Service (RDS) instance.

Configure the RDS instance in AWS as described in Setting up for Amazon RDS in the *Amazon RDS documentation*.

NOTE: Consider the following when creating the EC2 instance:

- Make sure that the connectivity requirements listed in Deployment requirements for AWS Managed Microsoft AD support are met.
- Select the SQL Server edition that suits your needs the most. For most Active Roles Synchronization Service use cases, **SQL Server Standard Edition** is an optimal choice.
- Take note of the **Master username** and **Master password**, as these credentials will be required later.
- For **Storage type**, select **General Purpose SSD (gp2)**, and allocate a minimum storage of 60 GiB.
- Consider selecting **Enable storage autoscaling**. Selecting this setting is useful if the SQL Server is utilized with a heavy load most of the time, but it might incur

additional operational costs.

- For **Certificate authority**, select the **rds-ca-2019** certificate, as it is required for Microsoft OLE DB Driver 19 for SQL Server to function properly.

Verifying connectivity between the EC2 and RDS instances

After you created the RDS instance, you can test in the EC2 instance with the telnet client or Microsoft SQL Server Management Studio (SSMS) if the RDS connectivity was successfully configured.

To verify RDS connectivity in the EC2 instance

1. Log in to the EC2 instance created for Active Roles Synchronization Service.
2. To test connectivity to RDS, install the telnet client. To do so:
   a. Open Windows Server Manager.
   b. On the **Dashboard**, click **Add roles and features**.
   c. In **Installation Type**, select **Role-based or feature-based installation**, then click **Next**.
   d. In **Server Selection**, choose **Select a server from the server pool**, and make sure that the local server (the EC2 instance) is selected.
   e. In **Server Roles**, just click **Next**.
   f. In **Features**, select **Telnet Client**.
   g. In **Confirmation**, click **Install**, then **Close** the application.
3. To verify connectivity to the RDS instance, open the Windows Command Prompt, and run the following command:

   ```
   telnet <rds-server-endpoint> <port-number>
   ```

   To find the RDS server endpoint and port to specify, open the entry of the RDS instance in the AWS console, and check the values under **Connectivity & Security** > **Endpoint & port**.

   NOTE: If the command returns an empty prompt, that indicates connectivity between the EC2 instance and the RDS instance.
4. Download and install Microsoft SQL Server Management Studio (SSMS) on the EC2 instance.
5. To test the connection with SSMS, start the application, then in the **Connect to Server** dialog, specify the following attributes:
   - **Server type**: Select **Database Engine**.
   - **Server name**: The same RDS instance endpoint used in the telnet command.

- **Authentication**: Select **SQL Server Authentication**, then specify the admin user name and password created when configuring the RDS instance.

6. After you specified all connection properties, click **Connect**.

Installing and configuring Synchronization Service for AWS Managed Microsoft AD

When used to synchronize AWS Managed Microsoft AD resources and passwords from an on-premises AD environment to AWS Managed Microsoft AD, you must install and configure Synchronization Service on an Amazon Elastic Compute Cloud (EC2) instance.

Prerequisites

Before starting the procedure, make sure that the EC2 and RDS instances are connected, as described in Verifying connectivity between the EC2 and RDS instances.

To install and configure Synchronization Service for use with AWS Managed Microsoft AD

1. Download the Active Roles installation media to the EC2 instance.
2. Run the setup and install Active Roles Synchronization Service with all required prerequisites as described in Installing Synchronization Service.

   NOTE: Make sure that you install Microsoft OLE DB Driver 19 for SQL Server and all its prerequisites from the `Redistributables` folder of the installation media.

   Also, to make sure that the connection to the SQL Server is properly encrypted, download and install the latest AWS RDS Root Certificate by adding it to the **Trusted Root Certification Authorities** container of the `certmgr` (Manage User Certificates) utility. For more information, see Using SSL/TLS to encrypt a connection to a DB instance in the *Amazon RDS documentation*.
3. After installation is finished, start Active Roles Synchronization Service. The Configuration Wizard appears.
4. In **Service Account and Mode**, configure the following settings:
   - **Synchronization Service account**: Enter the user name and password of the domain admin account supplied by Amazon Web Services (AWS).
   - **Synchronization Service mode**: Select **Local**.

   When you are ready, click **Next**.
5. In **Instance Configuration**, select **Create a new configuration**, then click **Next**.
6. In **Database Connection**, configure the following settings:
   - **SQL Server**: Specify the endpoint URL of the RDS instance connected to your EC2 instance. You can check the endpoint of the RDS instance in the AWS

console by selecting the RDS instance, then navigating to **Connectivity & Security** > **Endpoint & port**.

- **Database**: Specify the name of the database that will be used by Synchronization Service (for example, `syncservice`).
- For authentication, select **Use SQL Server authentication**, then enter the user name and password of the primary user in your RDS instance (configured in Creating the RDS instance for the Active Roles Synchronization Service SQL Server).

7. In **Configuration File**, specify the name and save location of the Synchronization Service configuration file.
8. (Optional) For added security, specify a password for the configuration.
9. To apply your changes and start creating the configuration, click **Finish**.

2

Getting started

- Synchronization Service Console
- Synchronizing identity data
- Management Shell

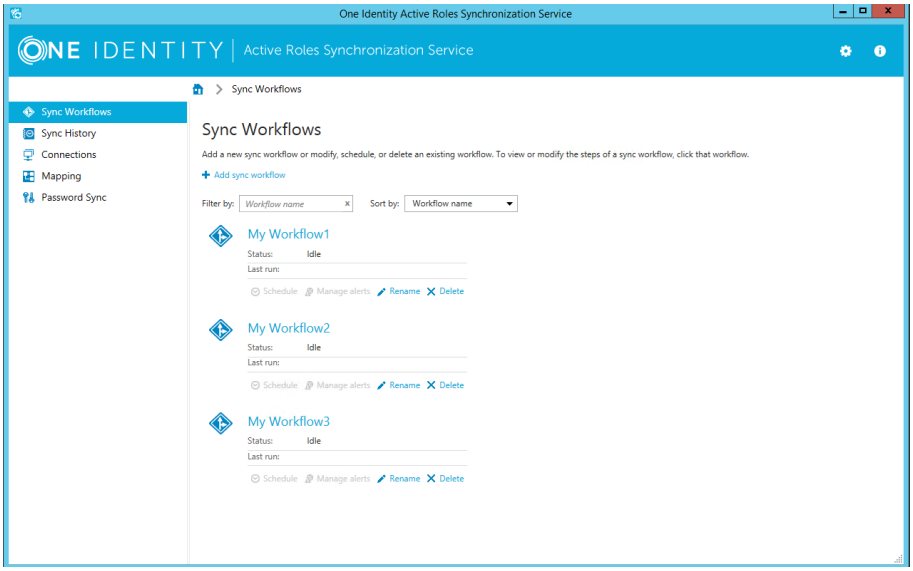
Synchronization Service Console

The Synchronization Service Console is a graphical user interface that provides access to the Synchronization Service functionality. You can use the Synchronization Service Console to connect Synchronization Service to external data systems, manage existing connections, and perform data synchronization operations between the connected data systems. The Synchronization Service Console is installed as part of Synchronization Service.

To start the Synchronization Service Console, depending on the version of your Windows operating system, click **Active Roles 8.2 Synchronization Service** on the **Apps** page or select **All Programs** > **One Identity Active Roles 8.2** > **Active Roles 8.2 Synchronization Service** from the **Start** menu.

The Synchronization Service Console looks similar to the following:

Figure 3: Synchronization Service Console



Gear icon

In the upper right corner of the Synchronization Service Console, you can click the gear icon.

The Gear icon provides the following commands:

- **Configure Sync Service**: Starts a wizard that helps you change the configuration settings of the current Synchronization Service instance.
- **Import Configuration**: Starts a wizard that helps you to import configuration settings from a configuration file created by another instance of Synchronization Service.
- **Export Configuration**: Starts a wizard that helps you to save the configuration profile of the current Synchronization Service instance to a file. You can use this file to apply the saved configuration to other instances of Active Roles Synchronization Service deployed in your environment.
- **Mail Profiles**: Allows you to add, edit, or delete mail profiles for sending notification emails about sync workflow runs. For more information on how to use the email notification, see Using sync workflow alerts.
- **Diagnostic Logging**: Allows you to specify settings for writing Synchronization Service diagnostic data to the Synchronization Service log file or Windows Event Log.
- **Communication Port**: Allows you to change the communication port number used by the Synchronization Service.
- **Configure Azure BackSync**: Allows you to configure back synchronization operation in Azure with on-premises Active Directory objects.

Sync Workflows tab

The **Sync Workflows** tab allows you to manage data sync workflows for connected data systems. A sync workflow can include a number of synchronization steps, each performing a specific data synchronization operation (creation, deprovision, or update). For more information on sync workflows and their steps, see Synchronizing identity data.

You can also use this tab to manage email notification settings for each existing sync workflow. For more information, see Using sync workflow alerts alerts.

On the **Sync Workflows** tab, you can use the following elements (some of these elements become available only after you create at least one sync workflow with one or more synchronization steps):

- **Add sync workflow**: Creates a new sync workflow.
- **Filter by**: Allows you to filter existing sync workflows by the letters or text you type in the text box. The filter applies to the sync workflow names.
- **Sort by**: Allows you to sort existing sync workflows by workflow name, last run time, or the number of synchronization steps.
- **<Workflow Name>**: Represents a sync workflow. You can click the workflow name to view and add, delete, or modify synchronization steps in that workflow.
- **Schedule**: Allows you to create a schedule for running the sync workflow.
- **Manage alerts**: Allows you to add, delete, or edit alerts for a sync workflow. An alert allows you to automatically send notification emails about the completion of the sync workflow run to specified recipients.
- **Rename**: Allows you to rename the sync workflow.
- **Delete**: Deletes the sync workflow.

Sync History tab

The **Sync History** tab allows you to view and selectively clean up the synchronization history. This is the history of sync workflow runs and object mapping operations. For more information, see Synchronization history.

On the **Sync History** tab, you can use the following elements:

- **Clean up now**: Allows you to selectively clean up sync history entries by specifying the age of the entries that you want to clean up.
- **Schedule cleanup**: Allows you to schedule a recurring cleanup operation for the sync history.
- **Sync Workflow History**: Allows you to view a list of completed sync workflow runs and the details of objects that participated in a particular sync workflow run.
- **Mapping History**: Allows you to view a list of completed map and unmap operations and the details of objects that participated in those operations.

- **Search**: Allows you to search the Synchronization Service synchronization history for completed creation, deprovision, update, and sync passwords operations. You can search by a number of criteria, such as the target connected data system and object type on which the operation was performed and the time period during which the operation completed.
- **Usage Statistics**: Allows you to view usage statistics for each connector i.e. a number of processed objects, sync runs, and so on.

Connections tab

The **Connections** tab allows you to manage connections between the Synchronization Service and the external data systems you want to use for data synchronization operations.

For more information on creating connections to external data systems supported out of the box, see External data systems supported with built-in connectors.

On the **Connections** tab, you can use the following elements (some of these elements become available only after you create at least one connection):

- **Add connection**: Allows you to create a new connection to an external data system.
- **Filter by**: Allows you to filter existing connections by the letters or text you type in the text box. The filter applies to the connection names.
- **Sort by**: Allows you to sort existing connections by connection name, name of the connector used, or the frequency of usage in sync workflow steps.
- **<Connection Name>**: Represents a connection to an external data system. You can click a connection name to view or modify the corresponding connection settings.
- **Connection settings**: Allows you to view or modify settings for the connection.
- **Synchronization scope**: Allows you to view or modify synchronization scope for the connection.
- **Delete connection**: Deletes the connection.

Mapping tab

The **Mapping** tab allows you to manage mapping pairs and mapping rules for existing connections. To view or modify mapping pairs or rules for a connection, click the name of that connection. For more information on mapping pairs and rules, see Mapping objects.

On the **Mapping** tab, you can use the following elements (some of these elements become available only after you create at least one connection to an external data system):

- **Filter by**: Allows you to filter existing connections by the letters or text you type in the text box. The filter only applies to the connection names.

- **Sort by**: Allows you to sort existing connections by connection name, name of the connector used, or the frequency of usage in the sync workflow steps.
- **<Connection Name>**: Displays the name of a connection. You can click a connection name to view or modify the mapping settings for the corresponding connection.

When you click a connection name on this tab, you can manage mapping pairs for the connection by using the following elements (some of these elements become available after you create at least one mapping pair for the connection):

- **Add mapping pair**: Allows you to specify the types of objects in two connected systems for which you want to create a mapping pair.
- **<ObjectType1> - <ObjectType2>**: Represents a mapping pair and displays the object types that belong to the same mapping pair. You can click a mapping pair to view and change the scope of conditions where the object types belonging to that mapping pair will be mapped. To define these conditions, you can create mapping rules.
- **Schedule**: Allows you to schedule a recurring map operation for the current pair of objects.
- **Map now**: Allows you to manually run the map operation on the current pair of objects.
- **Delete**: Deletes the mapping pair on which you click this link.

When you click a mapping pair, you can manage mapping rules for the mapping pair by using the following elements (some of these elements become available only after you create at least one mapping rule for the mapping pair):

- **Map now**: Allows you to manually run the map operation on the mapping pair by using the conditions specified in the existing mapping rules.
- **Unmap**: Allows you to unmap the objects that were earlier mapped according to the settings specified for the mapping pair.
- **Schedule mapping**: Allows you to schedule a recurring map operation for the mapping pair.
- **Add mapping rule**: Allows you to create a rule that will define a condition for mapping objects that belong to the mapping pair.
- **Delete rule**: Deletes the mapping rule on which you click this link.
- **Move up**: Moves the current mapping rule one position up in the list.
- **Move down**: Moves the current mapping rule one position down in the list.

Mapping rules are applied in the order they are listed.

Password Sync tab

The **Password Sync** tab allows you to manage password sync rules to automate password synchronization from a specified Active Directory domain to other connected data systems.

For more information, see Automated password synchronization.

On the **Password Sync** tab, you can use the following elements (some of these elements become available only after you create at least one password sync rule):

- **Add password sync rule**: Allows you to create a rule for synchronizing passwords from an Active Directory domain to another connected system.
- **Password sync settings**: Allows you to specify how many times you want to retry the password synchronization operation in the event of a failure. Also allows you to type a Windows PowerShell script to generate passwords for the target connected system. For more information, see Using PowerShell script to transform passwords.
- **Delete rule**: Deletes the password sync rule on which you click this link.

Configuring diagnostic logging

In the Synchronization Service Console, you can configure a number of settings to write the Synchronization Service diagnostic data to a separate log file or to the Windows Event Log.

To configure diagnostic logging

1. In the upper right corner of the Synchronization Service Console, select **Settings** > **Diagnostic Logging**.
2. In the dialog that opens, use the following options:
    - **Windows Event Log Level**: Drag the slider to select one of the following options to write Synchronization Service data to the Windows Event Log:
        - **Error, Warning, and Information**: Records errors, warnings, and information events generated by Synchronization Service to the Windows Event Log.
        - **Error and Warning**: Records error and warning events generated by Synchronization Service to the Windows Event Log.
        - **Error**: Records error events generated by Synchronization Service to the Windows Event Log.
        - **Off**: Disables writing Synchronization Service data to the Windows Event Log.
    - **Synchronization Service log level**: Drag the slider to select one of the following logging levels for the Synchronization Service log:
        - **All Possible Events**: Writes detailed diagnostic data to the Synchronization Service log file.
        - **Important Events**: Writes only essential events to the Synchronization Service log file.
        - **Off**: Disables writing data to the Synchronization Service log file.
3. When you are finished, click **OK** to apply your settings.

How to synchronize identity data

On a very high level, you need to complete the following steps to synchronize identity data between two external data systems:

1. Connect the Synchronization Service to the data systems between which you want to synchronize identity data.

   For more information, see External data systems supported with built-in connectors.
2. Configure synchronization scope for the connected data systems.

   For more information, see Modifying synchronization scope for a connection.
3. Create a sync workflow.

   For more information, see Creating a sync workflow.
4. Create one or more steps in the sync workflow, and, if necessary, define synchronization rules for these steps.

   For more information, see Synchronizing identity data.
5. Run the sync workflow you have created.

   For more information, see Running a sync workflow.

You can also use the Synchronization Service to automatically synchronize passwords from a specified Active Directory domain to other connected data systems. For more information, see Automated password synchronization.

Synchronization Service Management Shell

Synchronization Service Management Shell is implemented as a Windows PowerShell module, providing an extension to the Windows PowerShell environment. The commands provided by Synchronization Service Management Shell conform to the Windows PowerShell standards, and are fully compatible with the default command-line tools of Windows PowerShell.

You can open Synchronization Service Management Shell either from the list of installed applications, or directly from Windows PowerShell.

To launch Synchronization Service Management Shell

1. In the operating system, open the **Start menu**.
2. In the Start menu, search for **Active Roles Synchronization Service Management Shell 8.2**, then click it.

   Alternatively, use the **Search bar** of the system tray to find Synchronization Service Management Shell, then click it for launch.

To load Synchronization Service Management Shell in Windows PowerShell

1. Start Windows PowerShell.
2. To load the Synchronization Service Management Shell module, run the following command:

   ```
   Import-Module -Name "<full-path-to-synchronization-service-module-file>"
   ```

   For example, if you installed Synchronization Service to the default installation location, the full command is as follows:

   ```
   Import-Module -Name "C:\Program Files\One Identity\Active Roles\8.2\SyncService\SyncServiceShell\SyncServiceManagementShell.psd1"
   ```

NOTE: When loading Synchronization Service, your system may indicate that the certificate of some digitally-signed files published by One Identity are untrusted, and that you must enable trust for the certificate issuer to run Synchronization Service. If this happens, press either **R** (Run once) or **A** (Always run). One Identity recommends selecting **A** to prevent this message appearing again.

Cmdlet naming conventions

All cmdlets are presented in verb-noun pairs. The verb-noun pair is separated by a hyphen (-) without spaces, and the cmdlet nouns are always singular. The verb refers to the action that the cmdlet performs. The noun identifies the entity on which the action is performed. For example, in the `Get-QCObject` cmdlet name, the verb is `Get` and the noun is `QCObject`. All the Management Shell cmdlets have the nouns prefixed with `QC`, to distinguish the Management Shell cmdlets from those provided byPowerShell itself or by other PowerShell modules.

Getting help

This section provides instructions on how to get help information for the cmdlets added by Management Shell to the Windows PowerShell environment.

Table 1: To view help

To view this	Run this command
A list of all the Synchronization Service Management Shell cmdlets available to the shell.	`Get-QCCommand`
Information about the parameters and other components of a Synchronization Service Management Shell cmdlet.	Run one of the following: • `Get-QCCommand <CmdletName>` • `Get-Command <CmdletName>`

To view this	Run this command
	NOTE: You can use wildcard character expansion. For example, to view information about the cmdlets with the names ending in `Workflow`, run this command: `Get-Command *Workflow`.
Basic help information for a Synchronization Service Management Shell cmdlet.	`Get-Help <CmdletName>`
Detailed help information for a Synchronization Service Management Shell cmdlet, including the descriptions of available parameters and usage examples.	`Get-Help <CmdletName> -full`
Basic information about how to use the help system in Windows PowerShell, including Help for the Synchronization Service Management Shell.	`Get-Help`

3

Connections to external data systems

- External data systems supported out of the box
- Using connectors installed remotely
- Creating a connection
- Renaming a connection
- Deleting a connection
- Modifying synchronization scope for a connection
- Using connection handlers
- Specifying password synchronization settings for a connection

External data systems supported with built-in connectors

Active Roles Synchronization Service supports the following external data systems with built-in connectors:

- Working with Active Directory
- Working with an AD LDS (ADAM) instance
- Working with Skype for Business Server
- Working with Oracle Database
- Working with Oracle Database user accounts
- Working with Exchange Server
- Working with Active Roles
- Working with One Identity Manager
- Working with a delimited text file
- Working with Microsoft SQL Server
- Working with Micro Focus NetIQ Directory
- Working with Salesforce
- Working with ServiceNow
- Working with Oracle Unified Directory
- Working with an LDAP directory service
- Working with an OpenLDAP directory service
- Working with IBM DB2
- Working with IBM AS/400
- Working with IBM RACF
- Working with MySQL database
- Working with an OLE DB-compliant relational database
- Working with SharePoint
- Working with Microsoft 365

- Working with Microsoft Azure Active Directory
- Configuring data synchronization with the SCIM Connector
- Configuring data synchronization with the Generic SCIM Connector

NOTE:

Due to limitations in Graph API that prevent creating Azure contacts in Azure AD, Active Roles Synchronization Service cannot synchronize Azure contacts from any source data system to Azure AD.

Depending on the source data system, attempting to synchronize Azure contacts to Azure AD can result in the following error messages:

- If the source data system is Active Roles:

  ```
  An error occurred: "A parameter cannot be found that matches parameter name '<attribute>."
  ```

- If the source data system is not Active Roles:

  ```
  Error: "Unexpected error."
  ```

Working with Active Directory

This section describes how to create or modify a connection to Active Directory so that Synchronization Service could work with data in that data system.

To create a connection to Active Directory domain, you need to use Synchronization Service in conjunction with a special connector called Active Directory Connector. This connector is included in the Synchronization Service package.

The Active Directory Connector supports the following features:

Table 2: Active Directory Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	Yes
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

The Active Directory Connector supports linked attributes existing in the Active Directory schema. Linked attributes allow you to establish associations between two objects.

Linked attributes exist in pairs, as follows:

- **Forward link attribute**: This is a linked attribute that exists on a source object (for example, the `member` attribute on the Group object). Forward link attributes can be single-valued or multivalued.
- **Back link attribute**: This is a linked attribute that can be specified on a target object (for example, the `memberOf` attribute on the User object). Back link attributes are multivalued and they must have a corresponding forward link attribute. Back link attributes are not stored in Active Directory. Rather, they are calculated based on the corresponding forward link attribute each time a query is issued.

Creating an Active Directory connection

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
    - **Connection name**: Type a descriptive name for the connection.
    - **Use the specified connector**: Select **Active Directory Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
    - **Any available domain controller in the specified domain**: Allows you to connect to an available domain controller in the Active Directory domain you specify. In the **Domain** text box, type the fully qualified domain name of the domain to which you want to connect.
    - **Specified domain controller**: Allows you to connect to a specific domain controller in a particular Active Directory domain. In the **Domain controller** text box, type the fully qualified domain name of the domain controller to which you want to connect.
    - **Active Directory forest**: Allows you to connect to the Active Directory forest you specify in this option. When synchronizing data to or from a connected forest, Synchronization Service automatically selects the appropriate domain controllers in the forest to read and write data according to the synchronization scope configured for the connection.
        - **Secure Sockets Layer usage**: Use this list to select one of the following:
            - **None**: Allows you to connect without using Secure Sockets Layer (SSL).
            - **Use**: Allows you to connect through SSL.

        - **Preferred**: Allows you to attempt the connection through SSL first. If this connection attempt fails, the Synchronization Service tries to connect without using SSL.
    - **Access Active Directory using**: Use this option to select one of the following:
        - **Synchronization Service account**: Allows you to access the Active Directory domain in the security context of the account under which the Synchronization Service is running.
        - **Windows account**: Allows you to access Active Directory in the security context of the account whose user name and password you specify below this option.
  - To test the connection with the new parameters, click **Test connection**.
5. Click **Finish** to create a connection to Active Directory.

Modifying an Active Directory connection

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing Active Directory connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
    - **Any available domain controller in the specified domain**: Allows you to connect to an available domain controller in the Active Directory domain you specify. In the **Domain** text box, type the fully qualified domain name of the domain to which you want to connect.
    - **Specified domain controller**: Allows you to connect to a specific domain controller in a particular Active Directory domain. In the **Domain controller** text box, type the fully qualified domain name of the domain controller to which you want to connect.
    - **Active Directory forest**: Allows you to connect to the Active Directory forest you specify in this option. When synchronizing data to or from a connected forest, Synchronization Service automatically selects the appropriate domain controllers in the forest to read and write data according to the synchronization scope configured for the connection.
        - **Secure Sockets Layer usage**: Use this list to select one of the following:
            - **None**: Allows you to connect without using Secure Sockets Layer (SSL).

        - **Use**: Allows you to connect through SSL.
        - **Preferred**: Allows you to attempt the connection through SSL first. If this connection attempt fails, the Synchronization Service tries to connect without using SSL.
    - **Access Active Directory using**: Use this option to select one of the following:
        - **Synchronization Service account**: Allows you to access the Active Directory domain in the security context of the account under which the Synchronization Service is running.
        - **Windows account**: Allows you to access Active Directory in the security context of the account whose user name and password you specify below this option.
  - To test the connection with the new parameters, click **Test connection**.
4. Optionally, you can narrow the number of objects participating in the connection scope by setting up filter conditions. On the **Connection Settings** tab, click the **Advanced** item to expand it, and then use the following list columns:
    - **Object type**: Use this column to select the Active Directory object types for which you want to configure filter conditions: click **Add Object Type** to add an object type to the list. Once you have added an object type, use the **Filter condition** column to specify a condition the objects of that type must meet in order to participate in the connection scope.
    - **Filter condition**: Use this column to specify a filter condition for the corresponding Active Directory object type. To specify a filter condition, type an LDAP query. The Active Directory objects that meet the specified filter condition will participate in the connection scope. When no filter condition specified for an object type, all objects that belong to that type participate in the connection scope.
5. When you are finished, click **Save**.

Communication ports required to synchronize data between two Active Directory domains

When synchronizing data between two Active Directory domains, Synchronization Service uses the following ports to access domain controllers in the domains:

Table 3: Required communication ports

Port	Protocol	Type of traffic	Direction of traffic
53	TCP/UDP	DNS	Inbound
88	TCP/UDP	Kerberos	Outbound
389	TCP/UDP	LDAP	Outbound
636	TCP	LDAP over SSL (LDAPS)	Outbound

Synchronizing user passwords between two Active Directory domains

You can automatically synchronize user passwords from one Active Directory domain to the other by using Synchronization Service. The next procedure assumes that Synchronization Service is already connected to the source and target domains. For more information, see Creating an Active Directory connection.

To synchronize user passwords between two Active Directory domains

1. Install Capture Agent on all domain controllers in the source and target Active Directory domains.
2. Use the `pwdHash` attribute to perform an initial synchronization of user passwords between the source and target domains:
   a. Create a new or choose an existing creating or updating synchronization step for the source and target domains.

      If you use an updating synchronization step, ensure that user objects in the source domain are properly mapped to their counterparts in the target domain. For more information on mapping objects, see Mapping objects.

      In the creating or updating synchronization step, configure a rule to synchronize the `pwdHash` attribute value from the user objects in the source domain to their counterparts in the target domain.

   b. Run the creating or updating synchronization step to perform an initial synchronization of user passwords from the source to the target domain.

   The step to perform an initial synchronization allows you to synchronize user passwords only once. If you want to synchronize all subsequent password changes on a permanent basis, complete the step to create a recurring run schedule.
3. Create a recurring run schedule for the synchronization step you configured previously. For instructions, see Running a sync workflow on a recurring schedule.
   - To synchronize all subsequent password changes from the source to the target domain, do one of the following:
     - Configure a password sync rule to automate the password synchronization between the two Active Directory domains. For more information, see Automated password synchronization.

Synchronizing SID history of users or groups

You can use Synchronization Service to synchronize SID history between user or group objects in two Active Directory domains. For example, you can synchronize SID history when migrating users from one Active Directory domain to the other.

NOTE: Consider the following when synchronizing SID history:

- To read SID data in the source Active Directory domain, you can use the `sIDHistory` or `objectSid` attribute.
- To write SID data to the target Active Directory domain, always use the `sIDHistory` attribute.

To synchronize SID history of users or groups

1. Install Capture Agent on all domain controllers in the source and target Active Directory domains you want to participate in the SID history synchronization.

   For more information on how to install Capture Agent, see Managing Capture Agent.

2. Use the **Specified domain controller** option to connect Synchronization Service to the source and target domains.

   For more information on how to connect Synchronization Service to an Active Directory domain, see Creating an Active Directory connection.

3. Create a new or choose an existing creating or updating synchronization step for the source and target domains.

   If you use an updating synchronization step, ensure that user objects in the source domain are properly mapped to their counterparts in the target domain. For more information on mapping objects, see Mapping objects.

4. Configure the synchronization step to do the following:

   - Read SID data in the source Active Directory domain. For this purpose, you can use the sIDHistory attribute or the `objectSid` attribute, or both.
   - Write SID data to the target Active Directory domain by using the **sIDHistory** attribute.

   To read attribute values in the source domain and write them to the target domain, you can configure attribute modification rules in your sync workflow step. For more information, see Modifying attribute values by using rules.

5. Run the created step to synchronize SID history.

Working with an AD LDS (ADAM) instance

This section explains how to create or modify a connection to an AD LDS (ADAM) instance so that Synchronization Service could work with data in that data system.

To create a connection to an AD LDS (ADAM) instance, you need to use Synchronization Service in conjunction with a special connector called AD LDS (ADAM) Connector. This connector is included in the Synchronization Service package.

The AD LDS (ADAM) Connector supports the following features:

Table 4: AD LDS (ADAM) Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	Yes
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Creating an AD LDS (ADAM) instance connection

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **AD LDS (ADAM) Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Server**: Type the fully qualified domain name of the computer on which the AD LDS (ADAM) instance to which you want to connect is running.
   - **Port**: Type the LDAP communication port number used by the AD LDS (ADAM) instance.
     - **Access AD LDS (ADAM) instance using**: Use this option to select one of the following:
       - **Synchronization Service account**: Allows you to access the target AD LDS (ADAM) instance in the security context of the account under which the Synchronization Service is running.
       - **Windows account**: Allows you to access the target AD LDS (ADAM) instance in the security context of the account whose user name and password you specify below this option.
   - **Advanced**: Click to specify advanced settings for connecting to the AD LDS

(ADAM) instance.

- To test the connection with the new parameters, click **Test connection**.

5. Click **Finish** to create a connection to the AD LDS (ADAM) instance.

Modifying an existing AD LDS (ADAM) instance connection

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing AD LDS (ADAM) instance connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
   - **Server**: Type the fully qualified domain name of the computer on which the AD LDS (ADAM) instance to which you want to connect is running.
   - **Port**: Type the LDAP communication port number used by the AD LDS (ADAM) instance.
     - **Access AD LDS (ADAM) instance using**: Use this option to select one of the following:
       - **Synchronization Service account**: Allows you to access the target AD LDS (ADAM) instance in the security context of the account under which the Synchronization Service is running.
       - **Windows account**: Allows you to access the target AD LDS (ADAM) instance in the security context of the account whose user name and password you specify below this option.
   - **Advanced**: Click to specify advanced settings for connecting to the AD LDS (ADAM) instance.
   - To test the connection with the new parameters, click **Test connection**.
4. Optionally, you can narrow the number of AD LDS (ADAM) objects participating in the connection scope by setting up filter conditions. On the **Connection Settings** tab, click the **Advanced** item to expand it, and then use the following list columns:
   - **Object type**: Use this column to select the AD LDS (ADAM) object types for which you want to configure filter conditions: click **Add Object Type** to add an object type to the list. Once you have added an object type to the list, use the **Filter condition** column to specify a condition the objects of that type must meet in order to participate in the connection scope.
   - **Filter condition**: Use this column to specify a filter condition for the corresponding AD LDS (ADAM) object type. To specify a filter condition, type an LDAP query. The AD LDS (ADAM) objects that meet the specified filter condition

will participate in the connection scope. When no filter condition specified for an object type, all objects that belong to that type participate in the connection scope.

5. When you are finished, click **Save**.

Working with Skype for Business Server

This section describes how to create or modify a connection to Microsoft Skype for Business Server with the Active Roles Synchronization Service, to read and write data in Skype for Business Server. It also lists the type of data you can read and/or write using the configured connection.

To create a connection to Skype for Business Server, use the **Skype for Business Server Connector** of Active Roles Synchronization Service.

The Skype for Business Connector supports the following features:

Table 5: Skype for Business Server Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes For more information on what data you can read and write in Skype for Business Server, see Supported Skype for Business Server data.
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No

Creating a new Skype for Business Server connection

You can create a new Skype for Business Server connection in the Synchronization Service Console.

Prerequisites

Before creating a new Skype for Business Server connection, make sure that unsigned Windows PowerShell scripts are allowed to run on the computer on which Active Roles Synchronization Service is installed. This is required because Synchronization Service uses Windows PowerShell scripts to work with Microsoft Skype for Business Server.

NOTE: To view the current Windows PowerShell initialization policy, use the `Get-ExecutionPolicy` cmdlet supplied with Windows PowerShell. To change the Windows PowerShell initialization policy, you can use the `Set-ExecutionPolicy` cmdlet of Windows PowerShell.

To create a new Skype for Business Server connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   a. **Connection name**: Type a descriptive name for the connection.
   b. **Use the specified connector**: Select **Skype for Business Server Connector**.
3. Click **Next**.
4. Set the following settings:
   - **Skype for Business Server computer name**: Specify the fully qualified domain name (FQDN) of the Skype for Business Server computer to which you want to connect.
   - **User name**: Specify a domain user account that has sufficient rights to administer Skype for Business Server users. The account must be a member of all of the following groups that Skype for Business Server creates in Active Directory: **CsAdministrator**, **CsUserAdministrator**, and **CsServerAdministrator**.
   - **Password**: Type the password of the specified user account.

   To verify the specified connection settings, click **Test Connection**.
5. To apply your changes, click **Finish**.

Modifying an existing Skype for Business Server connection

You can modify an existing Skype for Business Server connection in the Synchronization Service Console.

To modify an existing Skype for Business connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing Skype for Business Server connection you want to modify.
3. Expand the **Specify Skype for Business Server name and access account** element to modify the following settings:
   - **Skype for Business Server computer name**: Specify the fully qualified domain name (FQDN) of the Skype for Business Server computer to which you want to connect.
   - **User name**: Specify a domain user account that has sufficient rights to administer Skype for Business Server users. The account must be a member of all of the following groups that Skype for Business Server creates in Active Directory: **CsAdministrator**, **CsUserAdministrator**, and **CsServerAdministrator**.
   - **Password**: Type the password of the specified user account.
4. When you are finished, click **Save**.

Supported Skype for Business Server data

The following table lists the Skype for Business Server object types and the data synchronization operations supported by the Skype for Business Server Connector.

Table 6: Supported objects and operations

Object	Read	Create	Delete	Update
User Allows you to read and write data related to users in Skype for Business Server.	Yes	Yes	Yes	Yes
ArchivingPolicy Allows you to read and write data related to custom archiving policies configured on a per-user basis in Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
ClientPolicy Allows you to read and write data related to custom client policies configured on a per-user basis in Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute

Object	Read	Create	Delete	Update
Client policies define which Skype for Business Server features are available to users.				provided for this object type.
ClientVersionPolicy Allows you to read and write data related to custom client version policies configured on a per-user basis in Skype for Business Server. These policies define what clients (such as Microsoft Office Communicator 2007 R2) and their versions can be used in conjunction with Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
ConferencingPolicy Allows you to read and write data related to custom conferencing policies configured on a per-user basis in Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
DialPlanPolicy Allows you to read and write data related to custom dial plan policies configured on a per-user basis in Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
ExternalAccessPolicy Allows you to read and write data related to custom external access policies configured on a per-user basis in Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
LocationPolicy Allows you to read and write data related to custom location policies configured on	Yes	No	No	Yes NOTE: You can only

Object	Read	Create	Delete	Update
a per-user basis in Skype for Business Server. These policies determine the configuration of the Enhanced 9-1-1 (E9-1-1) Location Information service.				update one attribute provided for this object type.
MobilityPolicy Allows you to read and write data related to custom mobility policies configured on a per-user basis in Skype for Business Server. These policies determine who can use mobility features (such as Call via Work, Voice over IP (VoIP), or video).	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
PersistentChatPolicy Allows you to read and write data related to custom persistent chat policies configured on a per-user basis in Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
PinPolicy Allows you to read and write data related to custom PIN policies configured on a per-user basis in Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
VoicePolicy Allows you to read and write data related to custom voice policies configured on a per-user basis in Skype for Business Server.	Yes	No	No	Yes NOTE: You can only update one attribute provided for this object type.
Skype for BusinessSettings Allows you to read data related to a number of Skype for Business Server	Yes	No	No	No

Object	Read	Create	Delete	Update
settings. `Skype for BusinessSettings` is not a native Skype for Business Server object type and only exists in the Skype for Business Server Connector schema.				

For each of the previous Skype for Business Server object types, Synchronization Service provides special attributes that allow you to read or write data in Skype for Business Server. You can access and use these attributes from the Synchronization Service Console, for example when selecting the source and target attributes you want to include in the synchronization operation.

The following table shows the attributes provided by Synchronization Service and explains what data you can read or write in Skype for Business Server by using a particular attribute for every object, except the `Skype for BusinessSettings` object.

Table 7: General object attributes

Attribute	Type	Description	Supported operations
`Description`	Single-valued, string	Gets the policy description.	Read
`Identity`	Single-valued, string	Gets the unique identifier of the policy.	Read
`Members`	Multivalued, reference	Gets or sets the user accounts to which the policy is applicable.	Read, write
`Name`	Single-valued, string	Gets the name of the policy.	Read
`ObjectClass`	Single-valued, string	Gets the type of the Skype for Business Server object.	Read

The following table lists the `Skype for BusinessSettings` object attributes and the type of data you can read or write in Skype for Business Server by using a particular attribute.

Table 8: Skype for BusinessSettings attributes

Attribute	Type	Description	Supported operations
`Domains`	Multivalued, string	Gets information about Session Initiation Protocol (SIP) domains existing in your organization.	Read
`Identity`	Single-	Gets the unique identifier of the Skype	Read

Attribute	Type	Description	Supported operations
	valued, string	for Business object.	
`ObjectClass`	Single-valued, string	Gets the type of the Skype for Business Server object.	Read
`Pools`	Multivalued, string	Gets information about Skype for Business Server pools. A pool is a collection of computers that all run the same set of Skype for Business Server services.	Read
`ServerVersion`	Single-valued, string	Gets the Skype for Business Server version.	Read

Attributes required to create a Skype for Business Server user

To create a Skype for Business Server user, you must populate the following required attributes provided by Synchronization Service:

- `RegistrarPool`
- `SipAddress`
- `DistinguishedName`, `DisplayName`, or `Identity`

For more information about the attributes listed above, see Supported Skype for Business Server data.

Getting or setting the Telephony option value in Skype for Business Server

To get or set the **Telephony** option value for a Skype for Business Server user object, use the following attributes provided by Synchronization Service:

- `AudioVideoDisabled`
- `EnterpriseVoiceEnabled`
- `RemoteCallControlTelephonyEnabled`

For more information about these and other attributes that Synchronization Service provides for a Skype for Business Server user object, see Supported Skype for Business Server data.

The following table lists the attribute value combinations that correspond to a particular value in the **Telephony** option.

Table 9: Attribute value combinations in the Telephony option

Telephony option value in Skype for Business Server	AudioVideoDisabled	EnterpriseVoiceEnabled	RemoteCallControlTelephonyEnabled
Audio/video disabled	TRUE	FALSE	FALSE
PC-to-PC only	FALSE	FALSE	FALSE
Enterprise voice	FALSE	TRUE	FALSE
Remote call control	FALSE	FALSE	TRUE
Remote call control only	TRUE	FALSE	TRUE

Working with Oracle Database

This section describes how to create or modify an Active Roles Synchronization Service connection to Oracle Databases, so that you can synchronize data stored in those systems. It also lists the type of data you can read and/or write in an Oracle Database with Synchronization Service.

To create a connection to an Oracle Database, use the **Oracle Database Connector** of the Synchronization Service.

The Oracle Database Connector supports the following features:

Table 10: Oracle Database Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes

Feature	Supported
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No

Creating an Oracle Database connection

You can create a new Oracle Database connection in theSynchronization Service Console.

To create a new Oracle Database connection

1. Make sure that the Synchronization Service computer has the following software installed:
   - Oracle Client: Ensure Oracle Client is configured to connect to the Oracle service that can be used to access Oracle Database that hosts the data you want to work with.
   - Oracle Net Services
   - Oracle Data Provider for .NET

     For supported versions of this software, see the *System Requirements* section in the *Active Roles Synchronization Service Release Notes*.
2. In the Synchronization Service Console, open the **Connections** tab.
3. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Oracle Database Connector**.
4. Click **Next**.
5. On the **Specify connection settings** page, use the following options:
   - **Oracle service name**: Specify the name of the Oracle service you want to use to access Oracle Database. You can click **Refresh** to get a list of available Oracle services.
   - **Access Oracle service with**: Type the user name and password of the account with which you want to access the Oracle service.
   - To test the connection with the new parameters, click **Test connection**.
6. Click **Next**.
7. On the **Specify how to select and modify data** page, use the following options:

- **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
- **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.

8. Click **Next**.
9. On the **Specify attributes to identify objects** page, use the following options:
   - **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
   - **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
   - **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
   - **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
   - **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.
10. To finish creating the Oracle Database connection, click **Finish**.

Modifying an existing Oracle Database connection

You can modify an existing Oracle Database connection in the Synchronization Service Console.

To modify an Oracle Database connection

1. Make sure that the Synchronization Service computer has the following software installed:
   - Oracle Client: Ensure Oracle Client is configured to connect to the Oracle service that can be used to access Oracle Database that hosts the data you want to work with.
   - Oracle Net Services
   - Oracle Data Provider for .NET

     For supported versions of this software, see the *System Requirements* section in the *Active Roles Synchronization Service Release Notes*.
2. In the Synchronization Service Console, open the **Connections** tab.

3. Click **Connection settings** below the existing Oracle Database connection you want to modify.
4. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specifying connection settings for Oracle Database
   - Configuring advanced settings for an Oracle Database or Oracle Database user account connection
   - Specifying attributes to identify objects for Oracle Database
5. Click **Save**.

Specifying connection settings for Oracle Database

The **Specify connection settings** option provides the following options that allow you to modify the connection settings:

- **Oracle service name**: Specify the name of the Oracle service you want to use to access Oracle Database. You can click **Refresh** to get a list of available Oracle services.
- **Access Oracle service with**: Type the user name and password of the account with which you want to access the Oracle service.
- To test the connection with the new parameters, click **Test connection**.

Configuring advanced settings for an Oracle Database or Oracle Database user account connection

The **Advanced** setting provides the following options that allow you to specify custom SQL queries which will automatically run each time Synchronization Service has created, updated, or deleted a user account in Oracle Database:

- **SQL queries to run after user provisioned**: Specifies the SQL queries to run each time Synchronization Service creates a user account in the Oracle Database.
- **SQL queries to run after user updated**: Specifies the SQL queries to run each time Synchronization Service updates a user account in the Oracle Database.
- **SQL queries to run after user deprovisioned**: Specifies the SQL queries to run each time Synchronization Service deletes a user account in the Oracle Database.

Below each of these options, you can use the following buttons:

- **Add**: Adds a new SQL query to the list.
- **Edit**: Allows you to edit the SQL query selected in the list.
- **Delete**: Deletes the SQL query selected in the list.

SQL queries run in the order they are listed. If necessary, you can rearrange the SQL queries in the lists: select an SQL query in the appropriate list, then click the up or down arrow button to move the query as necessary.

Specifying attributes to identify objects for Oracle Database

The **Specify attributes to identify objects** option provides the following options, allowing you to specify the attributes for uniquely identifying each object in the connected data system:

- **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
- **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
- **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
- **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
- **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

Sample SQL queries for working with an Oracle Database

The sample queries provided below are only applicable if Synchronization Service is connected to the target Oracle Database through the Oracle Database Connector.

Example: Adding a new entry

This SQL query illustrates how to add a new entry to the table named `SQLConnTest1` in Oracle Database to which you want to provision data from another connected system.

Table 11: Adding a new entry to the SQLConnTest1 table

Database table structure	Sample query
`CREATE TABLE "SQLConnTest1"("Id" number,"attr1" nchar(64), "attr2" nchar (64))`	`Insert into SQLConnTest1(attr1) values(:attr1) returning Id into :Id`

In this sample query, `Id` stands for the attribute that uniquely identifies each object in the Oracle Database.

Example: Creating a new user

This SQL query illustrates how to create a new user in the Oracle Database:

```
call dbms_utility.exec_ddl_statement('CREATE USER ' || :USERNAME || '
IDENTIFIED BY ' || :newPassword)
```

In this sample query:

- `USERNAME` refers to the name of the attribute that uniquely identifies the user in the Oracle Database.
- `newPassword` refers to the name of the attribute that will store the initial password you want to set for the new Oracle Database user.

Working with Oracle Database user accounts

This section describes how to create or modify a connection to Oracle Database user accounts with the Active Roles Synchronization Service. It also lists the type of data you can read and/or write in Oracle Database user accounts with the Synchronization Service.

To create a connection to Oracle Database user accounts and work with the user accounts in that data system, use the **Oracle Database User Account Connector** of the Synchronization Service.

The Oracle Database User Account Connector supports the following features:

Table 12: Oracle Database User Account Connector – Supported features

Feature	Supported
Bidirectional synchronization	Yes

Feature	Supported
Specifies whether you can both read and write data in the connected data system.	
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes NOTE: Password synchronization is only supported for user accounts that are authenticated entirely by Oracle Database. The Oracle Database User Accounts Connector does not support password synchronization for Oracle Database user accounts that use external or global authentication from the side of the connected Oracle system.

Creating an Oracle Database user accounts connection

You can create a new Oracle Database user accounts connection in the Synchronization Service Console.

To create a new Oracle Database user accounts connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Oracle Database User Accounts Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Oracle service name**: Specify the name of the Oracle service you want to use to access Oracle Database user account. You can click **Refresh** to get a list of available Oracle services.

- **Access Oracle service with**: Type the user name and password of the account with which you want to access the Oracle service.
- To test the connection with the new parameters, click **Test connection**.

5. Click **Next**.
6. On the **Specify how to select and modify data** page, use the following options:
   - **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
   - **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.
7. Click **Next**.
8. On the **Specify attributes to identify objects** page, use the following options:
   - **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
   - **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
   - **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
   - **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
   - **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.
9. To complete configuring the connection to the Oracle Database, click **Finish**.

After connecting Synchronization Service to the Oracle Database with the Oracle Database User Accounts Connector, you can specify custom SQL queries that will automatically run each time after Synchronization Service created, updated, or deleted a user account in Oracle Database User Accounts. For more information, see Modifying an existing Oracle Database connection.

Modifying an existing Oracle Database user account connection

You can modify an existing Oracle Database user accounts connection in the Synchronization Service Console.

To modify an Oracle Database user accounts connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specifying connection settings for an Oracle Database user account connection
   - Configuring advanced settings for an Oracle Database or Oracle Database user account connection
3. Click **Save**.

Specifying connection settings for an Oracle Database user account connection

The **Specify connection settings** option provides the following settings, allowing you to modify the connection:

- **Oracle service name**: Specify the name of the Oracle service you want to use to access Oracle Database user account. You can click **Refresh** to get a list of available Oracle services.
- **Access Oracle service with**: Type the user name and password of the account with which you want to access the Oracle service.
- To test the connection with the new parameters, click **Test connection**.

Configuring advanced settings for an Oracle Database or Oracle Database user account connection

The **Advanced** setting provides the following options that allow you to specify custom SQL queries which will automatically run each time Synchronization Service has created, updated, or deleted a user account in Oracle Database:

- **SQL queries to run after user provisioned**: Specifies the SQL queries to run each time Synchronization Service creates a user account in the Oracle Database.
- **SQL queries to run after user updated**: Specifies the SQL queries to run each time Synchronization Service updates a user account in the Oracle Database.
- **SQL queries to run after user deprovisioned**: Specifies the SQL queries to run each time Synchronization Service deletes a user account in the Oracle Database.

Below each of these options, you can use the following buttons:

- **Add**: Adds a new SQL query to the list.
- **Edit**: Allows you to edit the SQL query selected in the list.
- **Delete**: Deletes the SQL query selected in the list.

SQL queries run in the order they are listed. If necessary, you can rearrange the SQL queries in the lists: select an SQL query in the appropriate list, then click the up or down arrow button to move the query as necessary.

Sample SQL queries for working with Oracle Database user accounts

This section provides some SQL query examples that you can use a baseline for your own queries toward the connected Oracle Database system.

Example: Calling an Oracle stored procedure

This SQL query illustrates how to call a specific Oracle stored procedure:

```
CALL "<ProcedureName>"('&USERNAME')
```

In this query:

- `ProcedureName` specifies the name of the Oracle stored procedure you want to call.
- `USERNAME` refers to the name of the attribute that uniquely identifies a user in the target Oracle Database system.

Example: Creating a new user in the Oracle Database

This SQL query illustrates how to create a new user in the connected Oracle Database:

```
insert into DatabaseTable(ColumnName) values (upper('&USERNAME'))
```

In this sample query:

- `DatabaseTable` specifies the name of the table into which the entry will be added.
- `USERNAME` refers to the name of the attribute that uniquely identifies a user in the target Oracle Database system.

Working with Exchange Server

This section describes how to create or modify a connection to Microsoft Exchange Server so that Synchronization Service could read and write data in that data system. This section

also describes what data you can read and/or write in Exchange Server by using Synchronization Service.

To create a connection to Microsoft Exchange, you need to use Synchronization Service in conjunction with a special connector called Exchange Server Connector. This connector is included in the Synchronization Service package.

The Exchange Server Connector supports the following features:

Table 13: Exchange Server Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No

Creating a new connection to Exchange Server

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Exchange Server Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Select the Exchange Server version to which you want to connect**: Select the Exchange Server version to which you want to connect. If you select the **Automatically select latest version** option, the connector searches your environment for available Exchange Server 2019, 2016, or 2013, and connects to the latest of these versions found. Use the **Automatically select latest version** option only together with the **Any available Exchange Server in the forest** option.
   - **Connect to**: Choose how you want to connect to Exchange Server by selecting one of the following:

    - **Any available Exchange Server in the forest**: Allows you to connect to any available Exchange Server computer residing in the Active Directory forest you specify. In the **Domain in the forest** text box, type the fully qualified domain name (FQDN) of any domain that belongs to the forest that includes the Exchange Server you want to connect to. If you select this option, make sure the account you specify under **Access Exchange Server using** has sufficient permissions to read the Root Directory Service Entry (rootDFS) and configuration naming context of the forest.
    - **Specified Exchange Server**: Allows you to connect to the Exchange Server computer whose fully qualified domain name (FQDN) you type in the provided text box.
  - **Advanced**: Opens a dialog that allows you to specify advanced options for connecting to Exchange Server and reading and writing Exchange configuration data in Active Directory.
  - Options related to reading and writing Exchange configuration data in Active Directory:
    - **Use default domain controller**: Causes Synchronization Service to read and write Exchange configuration data in Active Directory by using the default domain controller defined on the Exchange Server used for the connection.
    - **Use specified domain controller**: Causes Synchronization Service to read and write Exchange configuration data in Active Directory by using the domain controller whose FQDN is specified in the text box below this option.
  - Options related to connecting to Exchange Server:
    - **Connect using HTTPS**: Select this check box to connect to Exchange Server by using HTTPS.
    - **Validate server certificate**: Select this check box to validate server certificate on the target Exchange Server.
    - **Authentication method**: Select an authentication method to access Exchange Server.
  - **Access Exchange Server using**: Select one of the following access options:
    - **Synchronization Service account**: Allows you to access Exchange Server in the security context of the account under which the Synchronization Service is running.
    - **Windows account**: Allows you to access Exchange Server in the security context of the account whose user name and password you type in the provided text box.
  - To test the connection with the new parameters, click **Test connection**.
5. Click **Finish**.

Modifying an existing connection to Exchange Server

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
    - **Connection name**: Type a descriptive name for the connection.
    - **Use the specified connector**: Select **Exchange Server Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
    - **Select the Exchange Server version to which you want to connect**: Select the Exchange Server version to which you want to connect. If you select the **Automatically select latest version** option, the connector searches your environment for available Exchange Server 2019, 2016, or 2013, and connects to the latest of these versions found. Use the **Automatically select latest version** option only together with the **Any available Exchange Server in the forest** option.
    - **Connect to**: Choose how you want to connect to Exchange Server by selecting one of the following:
        - **Any available Exchange Server in the forest**: Allows you to connect to any available Exchange Server computer residing in the Active Directory forest you specify. In the **Domain in the forest** text box, type the fully qualified domain name (FQDN) of any domain that belongs to the forest that includes the Exchange Server you want to connect to. If you select this option, make sure the account you specify under **Access Exchange Server using** has sufficient permissions to read the Root Directory Service Entry (rootDFS) and configuration naming context of the forest.
        - **Specified Exchange Server**: Allows you to connect to the Exchange Server computer whose fully qualified domain name (FQDN) you type in the provided text box.
    - **Advanced**: Opens a dialog that allows you to specify advanced options for connecting to Exchange Server and reading and writing Exchange configuration data in Active Directory.
    - Options related to reading and writing Exchange configuration data in Active Directory:
        - **Use default domain controller**: Causes Synchronization Service to read and write Exchange configuration data in Active Directory by using the default domain controller defined on the Exchange Server used for the connection.

    - **Use specified domain controller**: Causes Synchronization Service to read and write Exchange configuration data in Active Directory by using the domain controller whose FQDN is specified in the text box below this option.
  - Options related to connecting to Exchange Server:
    - **Connect using HTTPS**: Select this check box to connect to Exchange Server by using HTTPS.
    - **Validate server certificate**: Select this check box to validate server certificate on the target Exchange Server.
    - **Authentication method**: Select an authentication method to access Exchange Server.
  - **Access Exchange Server using**: Select one of the following access options:
    - **Synchronization Service account**: Allows you to access Exchange Server in the security context of the account under which the Synchronization Service is running.
    - **Windows account**: Allows you to access Exchange Server in the security context of the account whose user name and password you type in the provided text box.
  - To test the connection with the new parameters, click **Test connection**.
5. When you are finished, click **Save**.

Exchange Server data supported out of the box

The next table lists the Exchange Server object types supported by the Exchange Server Connector out of the box and the operations you can perform on these objects by using the connector.

Table 14: Supported objects and operations

Object	Read	Create	Delete	Update
`ActiveSyncMailboxPolicy`	Yes	No	No	No
Allows you to read the Mobile Device mailbox policy settings for a specified Mobile Device mailbox policy. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.				

Object	Read	Create	Delete	Update
`AddressBookPolicy` Allows you to read data related to address book policies. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No
`AddressList` Allows you to read data related to a specified address list. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No
`DistributionGroup` Allows you to read or write data related to a specified distribution group. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	Yes	Yes	Yes
`DynamicDistributionGroup` Allows you to read or write data related to a specified dynamic distribution group. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization	Yes	Yes	Yes	Yes

Object	Read	Create	Delete	Update
Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.				
`ExchangeServer` Allows you to read attribute values of a specified Exchange Server. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No
`GlobalAddressList` Allows you to read data related to a specified global address list (GAL). This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No
`Mailbox` Allows you to read or write data related to a specified mailbox. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	Yes	Yes	Yes
`MailboxDatabase` Allows you to read a specified mailbox database object. This object type works with the Exchange	Yes	No	No	No

Object	Read	Create	Delete	Update
Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.				
`MailContact`	Yes	Yes	Yes	Yes
Allows you to read or write data related to a specified mail-enabled contact. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*. NOTE: The Exchange Server Connector cannot create new users in Active Directory. You can create new AD users with the Active Directory Connector.				
`MailUser`	Yes	Yes	Yes	Yes
Allows you to read or write data related to a specified mail-enabled user. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*. NOTE: The Exchange Server Connector cannot create new users in Active Directory. You can create new AD users with the Active Directory Connector.				
`OfflineAddressBook`	Yes	No	No	No
Allows you to read data related to an offline address book (OAB). This object type works with the Exchange				

Object	Read	Create	Delete	Update
Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.				
`OrganizationConfig` Allows you to read configuration data of an Exchange organization. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No
`OwaMailboxPolicy` Allows you to read data related to Microsoft Office Outlook Web App mailbox policies in the Exchange organization. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No
`PublicFolder` Allows you to read data related to a public folder. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No

Object	Read	Create	Delete	Update
`RoleAssignmentPolicy` Allows you to read data related to a management role assignment policy. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No
`UmDialPlan` Allows you to read data related to a Unified Messaging (UM) dial plan. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No
`UmMailboxPolicy` Allows you to read data related to a Unified Messaging (UM) mailbox policy. This object type works with the Exchange Server versions supported by Active Roles Synchronization Service. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.	Yes	No	No	No

For each of the above-listed Exchange Server object types Synchronization Service provides a number of special attributes that allow you to read and/or write the data related to that object type in Exchange Server. You can access and use these attributes from the Synchronization Service Console (for example, when selecting the source and target attributes you want to participate in the synchronization operation).

The next sections describe the attributes provided by Synchronization Service and explain what data you can read and/or write in Exchange Server by using a particular attribute.

ActiveSyncMailboxPolicy object attributes

Table 15: ActiveSyncMailboxPolicy attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `ActiveSyncMailboxPolicy` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-ActiveSyncMailboxPolicy`**

AddressBookPolicy object attributes

Table 16: AddressBookPolicy attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `AddressBookPolicy` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-AddressBookPolicy`**

AddressList object attributes

Table 17: AddressList object attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `AddressList` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-AddressList`**

DistributionGroup object attributes

Table 18: DistributionGroup attributes

Attribute	Type	Description	Supported operations
`Members`	Multivalued, reference	Gets or sets the distribution group members. For recipients, this attribute accepts any of the following values: • `Alias` • `Canonical DN` • `Display Name` • `Distinguished Name (DN)` • `Domain\Account` • `GUID` • `Immutable ID` • `Legacy Exchange DN` • `SMTP Address` • `User Principal Name` For Active Directory users, this attribute accepts any of the following values: • `Distinquished Name (DN)` • `Domain\Account` • `GUID` • `User Principal Name (UPN)`	Read, Write
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `DistributionGroup` object have the same names and descriptions as parameters or return types of the following Exchange Management Shell cmdlets:

- **`Enable-DistributionGroup`**
- **`Get-DistributionGroup`**
- **`Set-DistributionGroup`**

DynamicDistributionGroup object attributes

Table 19: DynamicDistributionGroup attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `DynamicDistributionGroup` object have the same names and descriptions as parameters or return types of the following Exchange Management Shell cmdlets:

- **`Get-DynamicDistributionGroup`**
- **`New-DynamicDistributionGroup`**
- **`Set-DynamicDistributionGroup`**

ExchangeServer object attributes

Table 20: ExchangeServer attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `ExchangeServer` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-ExchangeServer`**

GlobalAddressList object attributes

Table 21: GlobalAddressList attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `GlobalAddressList` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- `Get-GlobalAddressList`

Mailbox object attributes

Table 22: Mailbox attributes

Attribute	Type	Description	Supported operations
`LinkedCredentialLogin`	Single-valued, string	Specifies the user name of the account with which you want to access the domain controller specified in the `LinkedDomainController` attribute.	Write
`LinkedCredentialPassword`	Single-valued, string	Specifies the password that matches the user name specified in the `LinkedCredentialLogin` attribute.	Write
`MoveMailboxTo`	Single-valued, string	Moves mailbox to the Exchange Server database whose name is specified in this attribute.	Write
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read
`RecipientTypeDetails`	Single-valued, string	Gets or sets a mailbox type. When you create a mailbox object, this attribute supports the following values: • `DiscoveryMailbox` • `EquipmentMailbox` • `RoomMailbox` • `SharedMailbox` • `UserMailbox` When you update a mailbox object, this attribute supports the following values: • `EquipmentMailbox` • `RoomMailbox` • `SharedMailbox`	Read, Write

Attribute	Type	Description	Supported operations
		• `UserMailbox` When you read data of a mailbox object, this attribute supports the following values: • `DiscoveryMailbox` • `EquipmentMailbox` • `LegacyMailbox` • `LinkedMailbox` • `RoomMailbox` • `SharedMailbox` • `UserMailbox`	

Other attributes provided for the `Mailbox` object have the same names and descriptions as parameters or return types of the following Exchange Management Shell cmdlets:

- **`Set-CalendarProcessing`**
- **`Get-CASMailbox`**
- **`Set-CASMailbox`**
- **`Disable-Mailbox`** (called by `Archive` and `RemoteArchive` attributes)
- **`Enable-Mailbox`** (called by `Archive` and `RemoteArchive` attributes)
- **`Get-Mailbox`**
- **`Set-Mailbox`**
- **`Get-MailboxAutoReplyConfiguration`**
- **`Set-MailboxAutoReplyConfiguration`**
- **`Get-MailboxStatistics`**
- **`Get-MoveRequest`**
- **`New-MoveRequest`**
- **`Remove-MoveRequest`**
- **`Set-MoveRequest`**
- **`Disable-UMMailbox`** (called by `UMEnabled` attribute)
- **`Enable-UMMailbox`** (called by `UMEnabled` attribute)
- **`Get-UMMailbox`**
- **`Set-UMMailbox`**
- **`Get-UMMailboxPIN`**
- **`Set-UMMailboxPIN`**

NOTE: Some attributes may perform actions by calling certain Exchange Management Shell cmdlets, as noted in the table.

MailContact object attributes

Table 23: MailContact attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `MailContact` object have the same names and descriptions as parameters or return types of the following Exchange Management Shell cmdlets:

- **`Enable-MailContact`**
- **`Get-MailContact`**
- **`Set-MailContact`**

NOTE: The Exchange Server Connector cannot create new users in Active Directory. You can create new AD users with the Active Directory Connector.

MailboxDatabase object attributes

Table 24: MailboxDatabase attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `MailboxDatabase` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-MailboxDatabase`**

MailUser object attributes

Table 25: MailUser attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `MailUser` object have the same names and descriptions as parameters or return types of the following Exchange Management Shell cmdlets:

- **`Enable-MailUser`**
- **`Get-MailUser`**
- **`Set-MailUser`**

NOTE: The Exchange Server Connector cannot create new users in Active Directory. You can create new AD users with the Active Directory Connector.

OfflineAddressBook object attributes

Table 26: OfflineAddressBook attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `OfflineAddressBook` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-OfflineAddressBook`**

OrganizationConfig object attributes

Table 27: OrganizationConfig attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `OrganizationConfig` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-OrganizationConfig`**

OwaMailboxPolicy object attributes

Table 28: OwaMailboxPolicy attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `OwaMailboxPolicy` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-OwaMailboxPolicy`**

PublicFolder object attributes

Table 29: PublicFolder attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `PublicFolder` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-PublicFolder`**

PublicFolderDatabase object attributes

Table 30: PublicFolderDatabase attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `PublicFolderDatabase` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-PublicFolderDatabase`**

RoleAssignmentPolicy object attributes

Table 31: RoleAssignmentPolicy attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `RoleAssignmentPolicy` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-RoleAssignmentPolicy`**

StorageGroup object attributes

Table 32: StorageGroup attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `StorageGroup` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-StorageGroup`**

UmDialPlan object attributes

Table 33: UmDialPlan attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `UmDialPlan` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-UMDialPlan`**

UmMailboxPolicy object attributes

Table 34: UmMailboxPolicy attributes

Attribute	Type	Description	Supported operations
`ObjectID`	Single-valued, string	Gets the unique identifier for a specified object in Exchange Server.	Read

Other attributes provided for the `UmMailboxPolicy` object have the same names and descriptions as parameters of the following Exchange Management Shell cmdlet:

- **`Get-UMMailboxPolicy`**

Scenario: Migrate mailboxes from one Exchange Server to another

To migrate a mailbox, you need to use the `MoveMailboxTo` attribute provided for the `Mailbox` object. Update the value of the `MoveMailboxTo` attribute, so that it includes the name or GUID of the Exchange Server database to which you want to move the mailbox. As a result, the mailbox is migrated to the Exchange Server computer that hosts the specified database.

NOTE: Before migrating mailboxes, consider the following:

- You can only migrate mailboxes between Exchange Servers that belong to the same Exchange organization.
- If the computers between which you want to migrate mailboxes run the same version of Exchange Server, make sure they have either no or the same Exchange Server Service Pack installed.

Configuring a connection to Exchange Server

Configure a connection to the Exchange Server installation you will use to move the mailbox object. For more information on the Microsoft Exchange Server versions supported by Active Roles Synchronization Service, see *System requirements* in the *Active Roles Synchronization Service Release Notes*.

NOTE: Both the source and target computers must have either the same Exchange Server Service Packs installed, or no Exchange Server Service Packs installed at all.

For more information on how to configure a connection to Exchange Server, see Creating a new connection to Exchange Server.

Creating a new sync workflow

For more information on how to create a new sync workflow, see Creating a sync workflow.

Configuring a step to update MoveMailboxTo attribute value

To configure a step to update MoveMailboxTo attribute value

1. In the sync workflow you created, create a new update step.
2. In the update step, select the target data system for the data synchronization operation. This must be the Exchange Server to which you created the connection.
3. Configure the update step so that it updates the value of the **MoveMailboxTo** attribute on the appropriate **Mailbox** objects. The new attribute value must include the name or GUID of the Exchange Server database to which you want to move the mailboxes.

For instructions on how to create and configure an update step, see Creating an update step.

Running the sync workflow

For more information on how to run a sync workflow, see Running a sync workflow.

Working with Active Roles

To create a connection to Active Roles, you need to use Synchronization Service in conjunction with a special connector called Active Roles Connector included in the Synchronization Service package.

The Active Roles Connector supports the following Synchronization Service features:

Table 35: Active Roles Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	Yes

Feature	Supported
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

The Active Roles Connector supports linked attributes in the Active Directory schema. Linked attributes allow you to associate one object with another object. Linked attributes exist in pairs:

- **Forward link attribute**: This is a linked attribute that exists on a source object (for example, the `member` attribute on the Group object). Forward link attributes can be single-valued or multivalued.
- **Back link attribute**: This is a linked attribute that can be specified on a target object (for example, the `memberOf` attribute on the User object). Back link attributes are multivalued and they must have a corresponding forward link attribute. Back link attributes are not stored in Active Directory. Rather, they are calculated based on the corresponding forward link attribute each time a query is issued.

Creating an Active Roles connection

You can create a connection to Active Roles right after you install Synchronization Service on your computer.

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
    - **Connection name**: Type a descriptive name for the connection.
    - **Use the specified connector**: Select **Active Roles Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
    - **Connect to**: Allows you to specify the Active Roles Administration Service to be used by the Synchronization Service. You can use one of the following options:
        - **Administration Service on the specified computer**: Type the name of the computer running the Administration Service you want Active Roles Synchronization Service to use.
        - **Any Administration Service of the same configuration**: Specify any Administration Service whose database holds the necessary configuration: type the DNS name of the computer running that Administration Service. If Active Roles replication is used to synchronize

configuration data, this must be any Administration Service whose database server acts as the Publisher for the configuration database.

- **Active Roles version**: Prompts you to specify the version of the Active Roles Administration Service to which you want to connect. You can choose to connect either to version 7.0 or later or to version 6.9 or earlier. In the latter case, you have to install the Active Roles ADSI Provider of the respective legacy Active Roles version on the computer running the Synchronization Service. For installation instructions, see the *Active Roles Installation Guide*.
- **Access Active Roles Administration Service using**: Allows you to specify an authentication option to access the Active Roles Administration Service. You can use one of the following options:
  - **Active Roles account**: Allows you to access the Administration Service in the security context of the user account under which the Active Roles is running.
  - **Windows account**: Allows you to access the Administration Service in the security context of the user account whose user name and password you specify below this option.
- To test the connection with the new parameters, click **Test connection**.

5. Click **Finish** to create a connection to Active Roles.

Modifying an Active Roles connection

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing Active Roles connection you want to modify.
3. On the **Specify connection settings** page, use the following options:
   - **Connect to**: Allows you to specify the Active Roles Administration Service to be used by the Synchronization Service. You can use one of the following options:
     - **Administration Service on the specified computer**: Type the name of the computer running the Administration Service you want Active Roles Synchronization Service to use.
     - **Any Administration Service of the same configuration**: Specify any Administration Service whose database holds the necessary configuration: type the DNS name of the computer running that Administration Service. If Active Roles replication is used to synchronize configuration data, this must be any Administration Service whose database server acts as the Publisher for the configuration database.

- **Active Roles version**: Prompts you to specify the version of the Active Roles Administration Service to which you want to connect. You can choose to connect either to version 7.0 or later or to version 6.9 or earlier. In the latter case, you have to install the Active Roles ADSI Provider of the respective legacy Active Roles version on the computer running the Synchronization Service. For installation instructions, see the *Active Roles Installation Guide*.
- **Access Active Roles Administration Service using**: Allows you to specify an authentication option to access the Active Roles Administration Service. You can use one of the following options:
  - **Active Roles account**: Allows you to access the Administration Service in the security context of the user account under which the Active Roles is running.
  - **Windows account**: Allows you to access the Administration Service in the security context of the user account whose user name and password you specify below this option.
- To test the connection with the new parameters, click **Test connection**.

4. Click **Save**.

Working with One Identity Manager

To create a connection to One Identity Manager, use the **One Identity Manager Connector** of Active Roles Synchronization Service.

The One Identity Manager Connector supports the following Synchronization Service features:

Table 36: One Identity Manager Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	Yes
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No

Creating a One Identity Manager connection

Synchronization Service supports One Identity Manager, allowing you to create a connection to Identity Manager right after installing Synchronization Service.

To create a new Identity Manager connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **One Identity Manager Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Application Server URL**: Specify the address of the One Identity Manager application server to which you want to connect.
   - **Authentication module**: Identifies the One Identity Manager authentication module that is to be used to verify the connection's user ID and password.
   - **User name**: Specify the user ID for this connection.
   - **Password**: Specify the password of the user ID for this connection.
   - To test the connection with the new parameters, click **Test connection**.
5. Click **Next**.

   The One Identity Manager modules, target systems, and containers appear.
6. Select the required One Identity Manager modules.

   NOTE: The One Identity Manager target systems and One Identity Manager containers are applicable only for the Target System Base module (that is, `UNS<x>B` tables).
7. To finish creating the connection to One Identity Manager, click **Finish** .

Modifying a One Identity Manager connection

You can modify an existing One Identity Manager Connector in the Active Roles Synchronization Service Console.

To modify an existing One Identity Manager Identity Manager connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing One Identity Manager connection you want to modify.
3. On the **Specify connection settings** page, use the following options:

- **Application Server URL**: Specify the address of the One Identity Manager application server to which you want to connect.
- **Authentication module**: Identifies the One Identity Manager authentication module that is to be used to verify the connection's user ID and password.
- **User name**: Specify the user ID for this connection.
- **Password**: Specify the password of the user ID for this connection.
- To test the connection with the new parameters, click **Test connection**.

4. Click **Next**.

   The One Identity Manager modules, target systems, and containers are displayed.

5. Select the required One Identity Manager modules.

   NOTE: The One Identity Manager target systems and One Identity Manager containers are applicable only for the Target System Base module (that is, `UNS<x>B` tables).

6. To finish creating the One Identity Manager connection, click **Finish**.

One Identity Manager Connector configuration file

One Identity Manager Connector saves its configuration settings in the configuration file (`.xml` file) located in the Active Roles Synchronization Service installation folder. You can edit the XML elements in the file to configure the various parameters of the One Identity Manager Connector. The table below describes the XML elements you can edit.

Table 37: XML elements

XML element	Description

`<ExcludeDeletedObjects>`	Specifies how Active Roles Synchronization Service will treat objects marked as deleted in Identity Manager. This element can take one of the following values:
	• **TRUE**: Specifies to ignore deleted objects during data synchronization operations.
	• **FALSE**: Specifies to process deleted objects during data synchronization operations.
	For example:
	`<ExcludeDeletedObjects>`
	`TRUE`
	`</ExcludeDeletedObjects>`

`<PasswordAttributes>`	Specifies the default Identity Manager attribute to be used for storing passwords for objects of a particular type. Specifying an attribute for storing passwords in the Active Roles Synchronization Service GUI overrides the value set in

XML element	Description

	this XML element.
	For example:
	`<PasswordAttributes>`
	`  <PasswordAttributeDefinitions>`
	`  <PasswordAttributeDefinition objectType-`
	`e="Person" attribute="CentralPassword" />`
	`  </PasswordAttributeDefinitions>`
	`</PasswordAttributes>`

`<ReadFullSync>`	Specifies a value of the `FullSync` variable for Read operations performed in Identity Manager.
`<CreateFullSync>`	Specifies a value of the `FullSync` variable for Create operations performed in Identity Manager.
`<ModifyFullSync>`	Specifies a value of the `FullSync` variable for Modify operations performed in Identity Manager.
`<DeleteFullSync>`	Specifies a value of the `FullSync` variable for Delete operations performed in Identity Manager.
`<ObjRefFullSync>`	Specifies a value of the `FullSync` variable for Modify Object Reference operations performed in Identity Manager.
`<SyncStatusFullSync>`	Specifies a value of the `FullSync` variable for Sync Status operations performed in Identity Manager.

For more information about the `FullSync` variable and the values it can take, see the One Identity Manager documentation.

Working with a delimited text file

This section describes how to create or modify a connection to a delimited text file so that Synchronization Service could work with data in that file.

To create a connection to a delimited text file, you need to use Synchronization Service in conjunction with a special connector called Delimited Text File Connector. This connector is included in the Synchronization Service package.

The Delimited Text File Connector supports the following features:

Table 38: Delimited Text File Connector – Supported features

Feature	Supported
Bidirectional synchronization	No

Feature	Supported
Specifies whether you can both read and write data in the connected data system.	
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	Yes
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No

Creating a delimited text file connection

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Delimited Text File Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Delimited text file**: Click **Browse** to locate and select the delimited text file to which you want to connect.
   - **Access delimited text file using**: Select an access option:
     - **Synchronization Service account**: Access the delimited text file in the security context of the account under which the Synchronization Service is running.
     - **Windows account**: Access the delimited text file in the security context of the account whose user name and password you specify below this option.
   - To test the connection with the new parameters, click **Test connection**.
5. Click **Next**.
6. On the **Specify delimited text file format** page, use the following options to provide information about the delimited text file format:
   - **Delimiter**: Select the delimiter used in the file you specified.
   - **Use first row for attribute names**: Select this check box if the first line of the specified file contains names of attributes. Otherwise, leave this check box

cleared.

- **Advanced**: Click this button to specify advanced options to access the delimited text file, such as encoding, row delimiter, value delimiter, and text qualifier.

7. Click **Next**.
8. On the **Specify attributes to identify objects** page, use the following options:
   - **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
   - **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
   - **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
   - **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
   - **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.
9. Click **Finish** to create a connection to the delimited text file.

Modifying an existing delimited text file connection

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing delimited text file connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specify connection settings for a delimited text file connection
   - Specify delimited text file format
   - Schema
   - Specify attributes to identify objects for a delimited text file connection
4. When you are finished, click **Save**.

Specify connection settings for a delimited text file connection

In this expandable item, you can use the following options:

- **Delimited text file**: Click **Browse** to locate and select the delimited text file to which you want to connect.
- **Access delimited text file using**: Select an access option:
  - **Synchronization Service account**: Access the delimited text file in the security context of the account under which the Synchronization Service is running.
  - **Windows account**: Access the delimited text file in the security context of the account whose user name and password you specify below this option.
- To test the connection with the new parameters, click **Test connection**.

Specify delimited text file format

This expandable item provides the following options:

- **Delimiter**: Select the delimiter used in the file you specified.
- **Use first row for attribute names**: Select this check box if the first line of the specified file contains names of attributes. Otherwise, leave this check box cleared.
- **Advanced**: Click this button to specify advanced options to access the delimited text file, such as encoding, row delimiter, value delimiter, and text qualifier.

Schema

You can use this expandable item to view and modify the delimited text file schema saved in the Synchronization Service configuration database.

When you create a connection to a delimited text file, Synchronization Service reads the schema in the file (that is, the fields or columns related to each record in the file), and then saves the schema in the Synchronization Service configuration database. Synchronization Service then uses the saved file schema to read and modify the data in the connected file. Should the schema in the connected file change, you will need to reflect these changes in the **Schema** option so that Synchronization Service could correctly handle (read and write) the data in the changed file.

This expandable item provides the following options:

- **Attributes**: Lists the names of Synchronization Service attributes that correspond to certain columns or fields in the connected file. Basically, these are the names of attributes you can select and use in the Synchronization Service Console for each object in the connected delimited text file.

- **Add**: Allows you to add a new entry (for example, column or field) to the file schema saved in the Synchronization Service configuration database. You can use this button in case a new column or field was added to the connected file and you want to reflect this change in the file schema saved in the Synchronization Service configuration database.
- **Edit**: Allows you to edit the name of the selected Synchronization Service attribute associated with a certain column or field in the connected file. For example, you can use this button in case a field or column name was changed in the connected file and you want to reflect this change in the file schema saved in the Synchronization Service configuration database. Also you can use this button to edit the display name of a Synchronization Service attribute associated with a certain column or field in the connected file.
- **Remove**: Allows you to remove the selected attribute from the file schema saved in the Synchronization Service configuration database. For example, you can use this button in case a field or column name was deleted from the connected file and you want to reflect this change in the file schema saved in the Synchronization Service configuration database.
- **Reload scheme**: Allows you to update the file schema saved in the Synchronization Service configuration database by reloading the schema from the file to the configuration database. As a result, the file schema saved in the Synchronization Service configuration database will be completely rewritten with new data from the file.
- **Up arrow**: Moves the selected attribute up.
- **Down arrow**: Moves the selected attribute down.

Specify attributes to identify objects for a delimited text file connection

This expandable item provides the following options that allow you to specify the attributes with which you wish to uniquely identify each object in the delimited text file:

- **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
- **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
- **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
- **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
- **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

Working with Microsoft SQL Server

This section describes how to create or modify a connection to Microsoft SQL Server so that Synchronization Service could work with data in that data system.

To create a connection to Microsoft SQL Server, use the **Microsoft SQL Server Connector** included by default in the Active Roles Synchronization Service.

The Microsoft SQL Server Connector supports the following features:

Table 39: Microsoft SQL Server Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Creating a Microsoft SQL Server connection

You can create a new Microsoft SQL Server connection in the Synchronization Service Console.

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Microsoft SQL Server Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **SQL Server**: Type or select the name of the SQL Server computer that hosts the database you want to participate in data synchronization operations.
   - **Access SQL Server using**: Select an access option:

    - **Use Windows authentication**: Allows you to access the SQL Server in the security context of the account under which the Synchronization Service is running.
    - **Use SQL Server authentication**: Allows you to access the SQL Server in the security context of the SQL Server user account whose user name and password you specify below this option.
  - To test the connection with the new parameters, click **Test connection**.
5. Click **Next**.
6. On the **Specify how to select and modify data** page, use the following options:
  - **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
  - **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.
  - **Configure Settings**: Provides settings for modifying data in the connected system during synchronization operations. For example, you can specify the database tables in which you want to insert, update, or delete data during synchronization operations.
7. Click **Next**.
8. On the **Specify attributes to identify objects** page, use the following options:
  - **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
  - **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
  - **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
  - **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
  - **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.
9. To finish creating the connection to the Microsoft SQL Server database, click **Finish**.

Modifying an existing Microsoft SQL Server connection

You can modify an existing Microsoft SQL Server connection in theSynchronization Service Console.

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing Microsoft SQL Server connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specifying connection settings for a Microsoft SQL Server connection
   - Specifying how to select and modify data for a Microsoft SQL Server connection
   - Advanced
   - Specifying attributes to identify objects for a Microsoft SQL Server connection
4. When you are finished, click **Save**.

Specifying connection settings for a Microsoft SQL Server connection

This expandable item provides the following options that allow you to modify the connection settings:

- **SQL Server**: Type or select the name of the SQL Server computer that hosts the database you want to participate in data synchronization operations.
- **Access SQL Server using**: Select an access option:
  - **Use Windows authentication**: Allows you to access the SQL Server in the security context of the account under which the Synchronization Service is running.
  - **Use SQL Server authentication**: Allows you to access the SQL Server in the security context of the SQL Server user account whose user name and password you specify below this option.
- To test the connection with the new parameters, click **Test connection**.
- **Connect to database**: Type the name of the SQL database to which you want to connect.

Specifying how to select and modify data for a Microsoft SQL Server connection

The **Specify how to select and modify data** setting allows you to configure how to select and modify the data you want to be included in the synchronization process:

- **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
- **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.
- **Configure Settings**: Specifies the settings for modifying data in the connected system during synchronization operations. For example, you can specify the database tables in which you want to insert, update, or delete data during synchronization operations.

Advanced

Allows you to configure the running timeout for all SQL queries you specified in the connection settings (for example, those specified in the **Specify How to Select and Modify Data** option). Use the **SQL query execution timeout** box to type the timeout value you want to use.

Specifying attributes to identify objects for a Microsoft SQL Server connection

The **Specify attributes to identify objects** setting provides the following options that allow you to specify the attributes with which you want to uniquely identify each object in the connected data system:

- **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
- **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
- **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
- **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
- **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

Sample queries to modify SQL Server data

This section provides some sample SQL queries illustrating how to modify SQL Server data during synchronization operations. In the sample queries, `Id` refers to an attribute (a column name in an SQL Server table) that uniquely identifies an object in your SQL database. These examples can be used only for configuring connections to Microsoft SQL Server 2005.

Example: Inserting an object into a table

This sample illustrates how to create a query that inserts an object with specified attributes into the table named `SQLConnTest1`.

Table 40: Inserting an object into a table

Database table structure	Sample query
`CREATE TABLE [SQLConnTest1]([Id] [bigint] IDENTITY (1,1),[attr1] [nchar](64),[attr2] [nchar](64)))`	`INSERT into SQLConnTest1(Id) values(@Id)`

Example: Creating an SQL Server account

This sample illustrates how to create a SQL Server account, and then retrieve the `UniqueID` attribute for that account.

To define the scope where to create the SQL Server account, insert the following query in the **Query Editor** dialog:

```
SELECT sid as Id,name as login from sys.server_principals
```

Insert the following SQL query into the **Configure SQL Statements** dialog:

```
EXEC sp_addlogin @login, @newPassword;
EXEC sp_adduser @login,@login,'db_owner';
SELECT sid as Id from sys.server_principals where name=@login;
```

IMPORTANT: None of the attribute names used in SQL queries can include whitespace characters. For example, you cannot use names such as **"user password"**.

Working with Micro Focus NetIQ Directory

This section describes how to create or modify a connection to Micro Focus NetIQ Directory (formerly known as Novell eDirectory) so that Synchronization Service could work with Micro Focus NetIQ Directory data in that data system.

To create a connection to Micro Focus NetIQ Directory, use the **Micro Focus NetIQ Directory Connector**, included by default in Active Roles Synchronization Service.

The Micro Focus NetIQ Directory Connector supports the following features:

Table 41: Micro Focus NetIQ Directory Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Creating a Micro Focus NetIQ Directory connection

You can create a new Micro Focus NetIQ Directory connection in the Synchronization Service Console.

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Micro Focus NetIQ Directory Connector**.
3. Click **Next**.

4. On the **Specify connection settings** page, use the following options:
   - **Server**: Type the fully qualified domain name of the Micro Focus NetIQ Directory server to which you want to connect.
   - **Port**: Type the number of the communication port used by the Micro Focus NetIQ Directory server.
   - **Access Micro Focus NetIQ Directory Service using**: Type the user name and password with which you want to access Micro Focus NetIQ Directory. Ensure the account has sufficient permissions to perform operations (read, write) on objects in Micro Focus NetIQ Directory.
   - **Advanced**: Click this button to specify a number of advanced options to access Micro Focus NetIQ Directory. For example, you can select an authentication method, configure TLS/SSL usage for the connection, and select whether or not you want to use paged search.

     From this **Authentication method** list, select one of the following methods:
     - **Anonymous**: Allows you to establish the connection without passing credentials.
     - **Basic**: Specifies to use basic authentication.
     - **Microsoft Negotiate**: Specifies to use Microsoft Negotiate authentication.
     - **NTLM**: Specifies to use Windows NT Challenge/Response authentication.
     - **Digest**: Specifies to use Digest Access authentication.
     - **Sicily**: Employs a negotiation mechanism (Sicily) to choose the Microsoft Network Authentication Service, Distributed Password Authentication, or NTLM method.
     - **Distributed Password Authentication**: Specifies to use DPA authentication.
     - **Microsoft Network Authentication Service**: Specifies to authenticate with Microsoft Network Authentication Service.
     - **External**: Specifies to use an external authentication method for the connection.
     - **Kerberos**: Specifies to use Kerberos authentication.

     You can also use the following check boxes:
     - **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
     - **Switch to TLS/SSL after establishing connection**: Establishes the connection without using the TLS (SSL) encryption. Then, after the connection has been established, enables the TLS (SSL) encryption.
     - **Verify TLS/SSL certificate**: Specifies whether or not to check the TLS (SSL) certificate on the server.

- **Use paged search**: Specifies whether or not to use paged search for the connection. When selecting this check box, you can set a page size limit in the text box below.

- To test the connection with the new parameters, click **Test connection**.

5. To complete creating the Micro Focus NetIQ Directory connection, click **Finish**.

Modifying an existing Micro Focus NetIQ Directory connection

You can modify the various settings for an existing connection to Micro Focus NetIQ Directory, such as the Micro Focus NetIQ Directory server to connect to, communication port, access credentials, and the attributes used for naming objects in Micro Focus NetIQ Directory.

Every object in Micro Focus NetIQ Directory has a naming attribute from which the object name is formed. When you create a connection to the directory, a default naming attribute is selected for each object type in that data system. You can use the **Specify Naming Attributes** item to view the naming attribute currently selected for each object type in Micro Focus NetIQ Directory and optionally specify a different naming attribute.

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing Micro Focus NetIQ Directory connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specifying connection settings for a Micro Focus NetIQ Directory connection
   - Specifying naming attributes for a Micro Focus NetIQ Directory connection
4. Click **Save**.

Specifying connection settings for a Micro Focus NetIQ Directory connection

The **Specify connection settings** option provides the following options that allow you to modify the connection settings:

- **Server**: Type the fully qualified domain name of the Micro Focus NetIQ Directory server to which you want to connect.
- **Port**: Type the number of the communication port used by the Micro Focus NetIQ Directory server.

- **Access Micro Focus NetIQ Directory Service using**: Type the user name and password with which you want to access Micro Focus NetIQ Directory. Ensure the account has sufficient permissions to perform operations (read, write) on objects in Micro Focus NetIQ Directory.
- **Advanced**: Click this button to specify a number of advanced options to access Micro Focus NetIQ Directory. For example, you can select an authentication method, configure TLS/SSL usage for the connection, and select whether or not you want to use paged search.

  From this **Authentication method** list, select one of the following methods:

  - **Anonymous**: Allows you to establish the connection without passing credentials.
  - **Basic**: Specifies to use basic authentication.
  - **Microsoft Negotiate**: Specifies to use Microsoft Negotiate authentication.
  - **NTLM**: Specifies to use Windows NT Challenge/Response authentication.
  - **Digest**: Specifies to use Digest Access authentication.
  - **Sicily**: Employs a negotiation mechanism (Sicily) to choose the Microsoft Network Authentication Service, Distributed Password Authentication, or NTLM method.
  - **Distributed Password Authentication**: Specifies to use DPA authentication.
  - **Microsoft Network Authentication Service**: Specifies to authenticate with Microsoft Network Authentication Service.
  - **External**: Specifies to use an external authentication method for the connection.
  - **Kerberos**: Specifies to use Kerberos authentication.

  You can also use the following check boxes:

  - **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
  - **Switch to TLS/SSL after establishing connection**: Establishes the connection without using the TLS (SSL) encryption. Then, after the connection has been established, enables the TLS (SSL) encryption.
  - **Verify TLS/SSL certificate**: Specifies whether or not to check the TLS (SSL) certificate on the server.
  - **Use paged search**: Specifies whether or not to use paged search for the connection. When selecting this check box, you can set a page size limit in the text box below.
- To test the connection with the new parameters, click **Test connection**.

Specifying naming attributes for a Micro Focus NetIQ Directory connection

Every object in Micro Focus NetIQ Directory has a naming attribute from which the object name is formed. When you create a connection to the directory, a default naming attribute is selected for each object type in that data system. You can use the **Specify Naming Attributes** item to view the naming attribute currently selected for each object type in Micro Focus NetIQ Directory and optionally specify a different naming attribute.

This expandable item provides following options:

- **Default naming attribute**: Displays the default naming attribute set for the currently selected object type.
- **Add**: Adds a new naming attribute for the selected object type.
- **Edit**: Allows you to edit the name of the naming attribute currently specified for the selected object type.
- **Remove**: Removes the currently selected entry from the list.

Working with Salesforce

This section describes how to create or modify a connection to Salesforce so that Synchronization Service could work with data in that data system.

To create a connection to Salesforce, use the **Salesforce Connector** of Active Roles Synchronization Service.

The Salesforce Connector supports the following features:

Table 42: Salesforce Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes
Secure Sockets Layer (SSL) data encryption Specifies whether the connector can use SSL to encrypt data transmitted	Yes

Feature	Supported
between Active Roles Synchronization Service and the connected data system.	

Creating a Salesforce connection

You can create a new Salesforce connection in the Synchronization Service Console.

To create a new Salesforce connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Salesforce Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Connect to Salesforce Sandbox**: Select this check box if you want to connect to your Salesforce testing environment. If you want to connect to production environment, make sure this check box is cleared. For more information about Salesforce Sandbox, see the Salesforce documentation.
   - **User name**: Type the user name of the account with which you want to access Salesforce. The account must have the System Administrator profile in the target Salesforce system.
   - **Password**: Type the password of the account with which you want to access Salesforce.
   - **Security token**: Enter the security token provided to you by Salesforce. For more information on what a security token is and how to obtain it, see the Salesforce documentation.
   - **Use a proxy server for your LAN**: Select this check box if your LAN uses a proxy server, and then enter the proxy server address in the Proxy server box.
   - **Use credentials for proxy**: Select this check box if your proxy server requires authentication. Use the appropriate text boxes to specify the user name and password with which you want to authenticate.
   - To test the connection with the new parameters, click **Test connection**.
5. To complete the configuration of the Salesforce connection, click **Finish**.

Modifying an existing Salesforce connection

You can modify an existing Salesforce connection in the Synchronization Service Console.

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing Salesforce connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
   - **Connect to Salesforce Sandbox**: Select this check box if you want to connect to your Salesforce testing environment. If you want to connect to production environment, make sure this check box is cleared. For more information about Salesforce Sandbox, see the Salesforce documentation.
   - **User name**: Type the user name of the account with which you want to access Salesforce. The account must have the System Administrator profile in the target Salesforce system.
   - **Password**: Type the password of the account with which you want to access Salesforce.
   - **Security token**: Enter the security token provided to you by Salesforce. For more information on what a security token is and how to obtain it, see the Salesforce documentation.
   - **Use a proxy server for your LAN**: Select this check box if your LAN uses a proxy server, and then enter the proxy server address in the Proxy server box.
   - **Use credentials for proxy**: Select this check box if your proxy server requires authentication. Use the appropriate text boxes to specify the user name and password with which you want to authenticate.
   - To test the connection with the new parameters, click **Test connection**.
4. Click **Save**.

Salesforce data supported for synchronization

The Salesforce Connector of Active Roles Synchronization Service supports all Salesforce object types, with all operations (Create, Read, Update, Delete) that you can perform on those objects with native Salesforce tools.

To read and/or write data related to a particular object in Salesforce, you can use the following resources:

- **Native Salesforce fields**: In the Synchronization Service Console user interface these fields are referred to as attributes. For more information on native Salesforce fields, see the "Reference | Standard Objects" section in the Salesforce Web Services API Developer's Guide available online at www.salesforce.com/us/developer/docs/api/.
- **Additional attributes provided by the Salesforce Connector**: The names of all such attributes start with the va prefix. For information about these attributes, see

Additional user object attributes for a Salesforce connection and Additional group object attributes for a Salesforce connection

Additional user object attributes for a Salesforce connection

You can specify the following additional user attributes in your Salesforce connection.

Table 43: Additional user attributes

Attribute	Description	Supported operations
`vaProfileName`	Allows you to specify a Salesforce profile. For example, you can use this attribute to assign a Salesforce profile to a user being provisioned to Salesforce. To specify a profile, enter the profile name as it appears in the Salesforce user interface. Examples of `vaProfileName` values: • `System Administrator` • `Force.com - Free User`	Read, Write
`vaRoleName`	Allows you to specify a Salesforce role. For example, you can use this attribute to assign a Salesforce role to a user being provisioned to Salesforce. To specify a role, enter the role name in the format used in the Salesforce user interface. For more information on roles, see the *Salesforce documentation*.	Read, Write
`vaManagerName`	Allows you to specify a manager for a particular user. To specify a manager, enter the manager name in the format used in the Salesforce user interface.	Read, Write
`vaContactName`	Allows you to specify an associated contact for a particular user. To specify an associated contact, enter the associated contact name in the format used in the Salesforce user	Read, Write

	interface.	
`vaMemberOf`	Allows you to define group membership for a particular user. NOTE: Consider the following: • This attribute is primarily intended for group membership synchronization. • This attribute contains references to the groups where the user is a member.	Read, Write
`vaMemberOfName`	Allows you to define group membership for a particular user (for example, when provisioning a user to Salesforce). Specify the names of the Salesforce groups where you want the user to be a member.	Read, Write
`vaLocale`	Allows you to specify a locale for a particular user (for example, when provisioning a user to Salesforce). To specify a locale, enter the locale name in the format used in the Salesforce user interface. Example of a `vaLocale` value: `English (United States)`	Read, Write
`vaTimeZone`	Allows you to specify a time zone for a user (for example, when provisioning a user to Salesforce). To specify a time zone, enter the time zone name in the format used in the Salesforce user interface. Example of a `vaTimezone` value: `(GMT+00:00) Greenwich Mean Time (GMT)`	Read, Write
`vaEmailEncoding`	Allows you to specify outbound email encoding to be used for a user (for example, when provisioning a user to Salesforce). Specify email encoding in the format used in the Salesforce user interface.	Read, Write

	Example of a `vaEmailEncoding` value: `Unicode (UTF-8)`	
`vaLanguage`	Allows you to specify a user interface language for a particular user. The Salesforce user interface and help will be displayed to the user in the language you specify in this attribute.	Read, Write
`vaDelegatedApproverUserName`	Allows you to specify the name of the user you want to appoint as a delegated approver.	Read, Write
`vaDelegatedApproverGroupName`	Allows you to specify the name of a group all members of which you want to appoint as delegated approvers.	Read, Write

Additional group object attributes for a Salesforce connection

You can specify the following additional group attributes in your Salesforce connection.

Table 44: Additional group attributes

Attribute	Description	Supported operations
`vaMemberOf`	Allows you to define group membership for the group in Salesforce. NOTE: Consider the following when using this attribute: • This attribute is primarily intended for group membership synchronization. • This attribute contains references to other groups where this group is a member.	Read, Write
`vaMemberOfName`	Allows you to define group membership for the group. Specify the names of Salesforce groups where you want the group to be a member.	Read, Write
`vaMember`	Allows you to define members of the group. This attribute contains references to the users and/or groups that are members of a particular group.	Read, Write
`vaMemberName`	Allows you to define members of a particular group. Specify the names of users and/or groups you want to be members of the group.	Read, Write

Scenario: Provisioning users from an Active Directory domain to Salesforce

This scenario illustrates how to configure a sync workflow to provision users from an Active Directory domain to Salesforce.

Configuring a connection to the source Active Directory domain

For instructions on how to create a new connection to an Active Directory domain, see Creating an Active Directory connection.

Configuring a connection to Salesforce

For instructions on how to create a new connection to Salesforce, see Creating a Salesforce connection.

Creating a new sync workflow

For instructions on how to create a new sync workflow for the configured Salesforce connection, see Scenario: Provisioning users from an Active Directory domain to Salesforce.

Configuring a workflow step

Once the required connections and the sync workflow are set, configure a new workflow step.

To configure a workflow step

1. In the Synchronization Service Console, navigate to the **Workflows** tab and open the sync workflow you created by clicking its name. Then, click **Add synchronization step**.
2. On the **Select an action** page, click **Provision**, then click **Next**.
3. On the **Specify source and criteria** page, do the following:
    a. Click **Specify** in the **Source connected system** option, then click **Select existing connected system**, and select the Active Directory connection you configured in the **Configuring a connection to source Active Directory domain** step.
    b. Click **Finish**.
    c. In **Source object type**, click **Select**, then select the **User object type** from the list. Click **OK**.
    d. Click **Next**.
4. On the **Specify target** page, do the following:

a. Click **Specify** in the **Target connected system** option, then click **Select existing connected system**, and select the Salesforce connection you configured in the **Configuring a connection to Salesforce** step.

b. Click **Finish**.

c. Click **Select** in the **Target object type** option, then select the **User object type** from the list. Click **OK**.

d. Click **Next**.

5. On the **Specify provisioning rules** page, in the **Initial Attribute Population Rules** option, add rules to populate the following required attributes:

   - **Username**: Use this attribute to specify a Salesforce user name for the user being provisioned. Make sure the user name you specify meets the format `<UserName>@<Domain>`, for example **`jdoe@domain.com`**.
   - **vaProfileName**: Use this attribute to assign a Salesforce profile to the user being provisioned. A profile defines specific permissions a user has in Salesforce. For more information on profiles, see the Salesforce documentation. Alternatively, you can specify a Salesforce profile by using the **ProfileId** attribute.
   - **Email**: Use this attribute to specify an existing valid email address for the user being provisioned.
   - **LastName**: Use this attribute to specify the last name of the user being provisioned.
   - **Alias**: Use this attribute to specify a unique Salesforce alias for the user being provisioned. A Salesforce alias can include up to 8 characters. For more information on the **Alias** attribute, see the *Salesforce documentation*.

Running your workflow

For instructions on how to run a sync workflow, see Running a sync workflow.

Working with ServiceNow

This section describes how to create or modify a connection to ServiceNow so that Synchronization Service could work with data in that data system.

To create a connection to ServiceNow, use the **ServiceNow Connector** of Active Roles Synchronization Service.

The ServiceNow Connector supports the following features:

Table 45: ServiceNow Connector – Supported features

Feature	Supported
Bidirectional synchronization	Yes

Feature	Supported
Specifies whether you can both read and write data in the connected data system.	
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes
Secure Sockets Layer (SSL) data encryption Specifies whether the connector can use SSL to encrypt data transmitted between Active Roles Synchronization Service and the connected data system.	Yes

Creating a ServiceNow connection

To create a new ServiceNow connection, you must:

1. Configure ServiceNow to accept synchronization requests from Active Roles Synchronization Service.
2. Create a new ServiceNow connection in the Synchronization Service with the **ServiceNow Connector**.
3. Synchronize the configured ServiceNow Connector schema with the connected ServiceNow instance.

Configuring ServiceNow

To configure ServiceNow

1. Open the website of your ServiceNow instance.
2. In the left pane of the ServiceNow website, under **System Properties**, click **Web Services**.
3. Make sure ServiceNow requires basic authorization for incoming RSS and SOAP requests.
4. In the right pane, make sure you clear the check box below **This property sets the elementFormDefault attribute**.
5. Click **Save**.

Creating a new connection to ServiceNow

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **ServiceNow Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **ServiceNow instance name**: Type the name of the ServiceNow instance to which you want to connect.
   - **Access ServiceNow instance using**. Type the user name and password of the account with which you want to access the specified ServiceNow instance.
   - **Use a proxy server for your LAN**: Select this check box if your LAN uses a proxy server. Then enter the proxy server address in the Proxy server box.
   - **Use credentials for proxy**: Select this check box if your proxy server requires authentication. Use the appropriate text boxes to specify the user name and password with which you want to authenticate.
   - To test the connection with the new parameters, click **Test connection**.
5. To complete the configuration of the ServiceNow connection, click **Finish**.
6. Synchronize the ServiceNow Connector schema with that of the connected ServiceNow instance.

   This step is required to pass information about object classes and attributes existing in the connected ServiceNow instance to the ServiceNow Connector, so that the connector could correctly read and write data in the connected ServiceNow instance.

   To synchronize the connector schema, do the following:

   a. Below the ServiceNow connection you have just created, click the **Connection settings** link.

   b. On the **Connection Settings** tab, click the **Update connector schema** item to expand it.

   c. Click **Update Schema**.

Modifying an existing ServiceNow connection

You can modify an existing ServiceNow connection in theSynchronization Service Console.

To modify the connection settings of a ServiceNow connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing ServiceNow connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
    - The **Specify connection settings** item:
        - **ServiceNow instance name**: Type the name of the ServiceNow instance to which you want to connect.
        - **Access ServiceNow instance using**. Type the user name and password of the account with which you want to access the specified ServiceNow instance.
        - **Use a proxy server for your LAN**: Select this check box if your LAN uses a proxy server. Then enter the proxy server address in the Proxy server box.
        - **Use credentials for proxy**: Select this check box if your proxy server requires authentication. Use the appropriate text boxes to specify the user name and password with which you want to authenticate.
        - To test the connection with the new parameters, click **Test connection**.
    - The **Update connector schema** item:
        - **Update Schema**: Synchronizes the ServiceNow Connector schema with changes in the connected ServiceNow instance. Use this button whenever schema changes occur in the connected ServiceNow instance (for example, object classes or attributes are added or deleted in the ServiceNow instance). The ServiceNow Connector can only read and write data correctly if the connector schema is completely in sync with the ServiceNow instance.
4. Click **Save**.

ServiceNow data supported for synchronization

The ServiceNow Connector supports all object classes and attributes existing in the connected ServiceNow instance, provided that the ServiceNow Connector schema and the ServiceNow instance schema are completely in sync.

To synchronize the ServiceNow Connector schema with the connected ServiceNow instance schema, use the **Update Connector Schema** button in the ServiceNow connection settings. For more information, see Modifying an existing ServiceNow connection.

Working with Oracle Unified Directory

This section describes how to create or modify a connection to Oracle Unified Directory (formerly known as Sun One Directory) so that Synchronization Service could work with data in that data system.

To create a connection to Oracle Unified Directory, use the **Oracle Unified Directory Connector** of the Active Roles Synchronization Service.

The Oracle Unified Directory Connector supports the following features:

Table 46: Oracle Unified Directory Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Creating an Oracle Unified Directory connection

You can create a new Oracle Unified Directory connection in the Synchronization Service Console.

To create a new Oracle Unified Directory connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Oracle Unified Directory Server Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Server**: Type the fully qualified domain name of the computer running Oracle Unified Directory Server that manages the directory to which you want to

connect.

- **Port**: Type the number of the communication port used by Oracle Unified Directory Server.
- **Access Oracle Unified Directory Server using**: Type the user name and password of the account with which you want to access Oracle Unified Directory Server. Ensure the account has sufficient permissions to perform operations (read, write) on objects in the directory managed by Oracle Unified Directory Server.
- **Advanced**: Click this button to specify a number of advanced options to access the directory managed by Oracle Unified Directory Server. For example, you can select an authentication method, configure TLS/SSL usage for the connection, and select whether or not you want to use paged search.

  From this **Authentication method** list, select one of the following methods:

  - **Anonymous**: Allows you to establish the connection without passing credentials.
  - **Basic**: Specifies to use basic authentication.
  - **Microsoft Negotiate**: Specifies to use Microsoft Negotiate authentication.
  - **NTLM**: Specifies to use Windows NT Challenge/Response authentication.
  - **Digest**: Specifies to use Digest Access authentication.
  - **Sicily**: Employs a negotiation mechanism (Sicily) to choose the Microsoft Network Authentication Service, Distributed Password Authentication, or NTLM method.
  - **Distributed Password Authentication**: Specifies to use DPA authentication.
  - **Microsoft Network Authentication Service**: Specifies to authenticate with Microsoft Network Authentication Service.
  - **External**: Specifies to use an external authentication method for the connection.
  - **Kerberos**: Specifies to use Kerberos authentication.

  You can also use the following check boxes:

  - **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
  - **Switch to TLS/SSL after establishing connection**: Establishes the connection without using the TLS (SSL) encryption. Then, after the connection has been established, enables the TLS (SSL) encryption.
  - **Verify TLS/SSL certificate**: Specifies whether or not to check the TLS (SSL) certificate on the server.

- **Use paged search**: Specifies whether or not to use paged search for the connection. When selecting this check box, you can set a page size limit in the text box below.

- To test the connection with the new parameters, click **Test connection**.

5. To finish configuring the Oracle Unified Directory Server connection, click **Finish** .

Modifying an existing Oracle Unified Directory Server connection

You can modify the various settings for an existing connection to a directory managed by Oracle Unified Directory Server, such as the server computer to which the connection is established, communication port, access credentials, and the attributes used for naming objects in the directory.

Every object in a directory managed by Oracle Unified Directory Server has a naming attribute from which the object name is formed. When you create a connection to the directory, a default naming attribute is selected for each object type in that data system. You can view the default naming attribute currently selected for each object type in the directory and optionally specify a different naming attribute.

To modify the connection settings of an Oracle Unified Directory Server connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing Oracle Unified Directory connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specifying connection settings for an Oracle Unified Directory connection
   - Specifying naming attributes for an Oracle Unified Directory connection
4. Click **Save**.

Specifying connection settings for an Oracle Unified Directory connection

The **Specify connection settings** option provides the following settings that allow you to modify the connection:

- **Server**: Type the fully qualified domain name of the computer running Oracle Unified Directory Server that manages the directory to which you want to connect.
- **Port**: Type the number of the communication port used by Oracle Unified Directory Server.

- **Access Oracle Unified Directory Server using**: Type the user name and password of the account with which you want to access Oracle Unified Directory Server. Ensure the account has sufficient permissions to perform operations (read, write) on objects in the directory managed by Oracle Unified Directory Server.
- **Advanced**: Click this button to specify a number of advanced options to access the directory managed by Oracle Unified Directory Server. For example, you can select an authentication method, configure TLS/SSL usage for the connection, and select whether or not you want to use paged search.

  From this **Authentication method** list, select one of the following methods:

  - **Anonymous**: Allows you to establish the connection without passing credentials.
  - **Basic**: Specifies to use basic authentication.
  - **Microsoft Negotiate**: Specifies to use Microsoft Negotiate authentication.
  - **NTLM**: Specifies to use Windows NT Challenge/Response authentication.
  - **Digest**: Specifies to use Digest Access authentication.
  - **Sicily**: Employs a negotiation mechanism (Sicily) to choose the Microsoft Network Authentication Service, Distributed Password Authentication, or NTLM method.
  - **Distributed Password Authentication**: Specifies to use DPA authentication.
  - **Microsoft Network Authentication Service**: Specifies to authenticate with Microsoft Network Authentication Service.
  - **External**: Specifies to use an external authentication method for the connection.
  - **Kerberos**: Specifies to use Kerberos authentication.

  You can also use the following check boxes:

  - **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
  - **Switch to TLS/SSL after establishing connection**: Establishes the connection without using the TLS (SSL) encryption. Then, after the connection has been established, enables the TLS (SSL) encryption.
  - **Verify TLS/SSL certificate**: Specifies whether or not to check the TLS (SSL) certificate on the server.
  - **Use paged search**: Specifies whether or not to use paged search for the connection. When selecting this check box, you can set a page size limit in the text box below.
- To test the connection with the new parameters, click **Test connection**.

Specifying naming attributes for an Oracle Unified Directory connection

Every object in a directory managed by Oracle Unified Directory Server has a naming attribute from which the object name is formed. When you create a connection to the directory, a default naming attribute is selected for each object type in that data system. You can use the **Specify Naming Attributes** item to view the naming attribute currently selected for each object type in the directory and optionally specify a different naming attribute. The setting has the following options:

- **Default naming attribute**: Displays the default naming attribute set for the currently selected object type.
- **Add**: Adds a new naming attribute for the selected object type.
- **Edit**: Allows you to edit the name of the naming attribute currently specified for the selected object type.
- **Remove**: Removes the currently selected entry from the list.

Working with an LDAP directory service

This section describes how to create or modify a connection to an LDAP directory service so that Synchronization Service could work with data in that data system.

To create a connection to an LDAP directory service, you need to use Synchronization Service in conjunction with a special connector called Generic LDAP Connector. This connector is included in the Synchronization Service package.

The Generic LDAP Connector supports the following features:

Table 47: Generic LDAP Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Creating an LDAP directory service connection

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
    - **Connection name**: Type a descriptive name for the connection.
    - **Use the specified connector**: Select **Generic LDAP Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
    - **Server**: Type the fully qualified domain name of the computer running an LDAP directory service to which you want to connect.
    - **Port**: Type the number of the communication port used by the LDAP server to which you want to connect.
    - **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
    - **Use connectionless LDAP**: Enables the use of the connectionless LDAP (CLDAP) protocol for the connection.
    - **User name**: Type the user name of the account with which you want to bind.
    - **Password**: Type the password of the account with which you want to bind.
    - **Bind with Synchronization Service account**: Allows you to bind with the account under which the Synchronization Service is running.
    - **Bind with credentials**: Allows you to bind by specifying the credentials of a particular user account.
    - **Use simple bind**: Allows you to bind either without specifying user account credentials or with a user password only. In the latter case, the password you type is transmitted as clear text.
    - **Use custom bind**: Allows you to configure a number of advanced settings for binding. Click **Configure**, and then use the next options.
    - From this **Authentication method** list, select one of the following methods:
        - **Anonymous**: Allows you to establish the connection without passing credentials.
        - **Basic**: Specifies to use basic authentication.
        - **Microsoft Negotiate**: Specifies to use Microsoft Negotiate authentication.
        - **NTLM**: Specifies to use Windows NT Challenge/Response authentication.
        - **Digest**: Specifies to use Digest Access authentication.

- **Sicily**: Employs a negotiation mechanism (Sicily) to choose the Microsoft Network Authentication Service, Distributed Password Authentication, or NTLM method.
- **Distributed Password Authentication**: Specifies to use DPA authentication.
- **Microsoft Network Authentication Service**: Specifies to authenticate with Microsoft Network Authentication Service.
- **External**: Specifies to use an external authentication method for the connection.
- **Kerberos**: Specifies to use Kerberos authentication.

You can also use the following check boxes:

- **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
- **Switch to TLS/SSL after establishing connection**: Establishes the connection without using the TLS (SSL) encryption. Then, after the connection has been established, enables the TLS (SSL) encryption.
- **Verify TLS/SSL certificate**: Specifies whether or not to check the TLS (SSL) certificate on the server.
- **Use paged search**: Specifies whether or not to use paged search for the connection. When selecting this check box, you can set a page size limit in the text box below.

- To test the connection with the new parameters, click **Test connection**.

5. Click **Next**.
6. On the **Specify attributes to identify objects** page, specify the attributes with which you want to uniquely identify each object in the LDAP directory service.

   You can use the following options:

   - **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
   - **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
   - **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
   - **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
   - **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.
7. Click **Finish** to create a connection to the LDAP directory service.

Modifying an existing Generic LDAP directory service connection

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing generic LDAP connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specify connection settings for a Generic LDAP directory service connection
   - Specify directory partitions for a Generic LDAP directory service connection
   - Specify naming attributes for a Generic LDAP directory service connection
   - Specify attributes to identify objects for a Generic LDAP directory service connection
4. Click **Save**.

Specify connection settings for a Generic LDAP directory service connection

This expandable item provides the following options that allow you to modify the connection settings:

- **Server**: Type the fully qualified domain name of the computer running an LDAP directory service to which you want to connect.
- **Port**: Type the number of the communication port used by the LDAP server to which you want to connect.
- **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
- **Use connectionless LDAP**: Enables the use of the connectionless LDAP (CLDAP) protocol for the connection.
- **User name**: Type the user name of the account with which you want to bind.
- **Password**: Type the password of the account with which you want to bind.
- **Bind with Synchronization Service account**: Allows you to bind with the account under which the Synchronization Service is running.
- **Bind with credentials**: Allows you to bind by specifying the credentials of a particular user account.
- **Use simple bind**: Allows you to bind either without specifying user account credentials or with a user password only. In the latter case, the password you type is

transmitted as clear text.

- **Use custom bind**: Allows you to configure a number of advanced settings for binding. Click **Configure**, and then use the next options.
- From this **Authentication method** list, select one of the following methods:
  - **Anonymous**: Allows you to establish the connection without passing credentials.
  - **Basic**: Specifies to use basic authentication.
  - **Microsoft Negotiate**: Specifies to use Microsoft Negotiate authentication.
  - **NTLM**: Specifies to use Windows NT Challenge/Response authentication.
  - **Digest**: Specifies to use Digest Access authentication.
  - **Sicily**: Employs a negotiation mechanism (Sicily) to choose the Microsoft Network Authentication Service, Distributed Password Authentication, or NTLM method.
  - **Distributed Password Authentication**: Specifies to use DPA authentication.
  - **Microsoft Network Authentication Service**: Specifies to authenticate with Microsoft Network Authentication Service.
  - **External**: Specifies to use an external authentication method for the connection.
  - **Kerberos**: Specifies to use Kerberos authentication.

  You can also use the following check boxes:
  - **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
  - **Switch to TLS/SSL after establishing connection**: Establishes the connection without using the TLS (SSL) encryption. Then, after the connection has been established, enables the TLS (SSL) encryption.
  - **Verify TLS/SSL certificate**: Specifies whether or not to check the TLS (SSL) certificate on the server.
  - **Use paged search**: Specifies whether or not to use paged search for the connection. When selecting this check box, you can set a page size limit in the text box below.
- To test the connection with the new parameters, click **Test connection**.

Specify directory partitions for a Generic LDAP directory service connection

Allows you to specify the directory partitions you want to participate in the synchronization operations by selecting the check boxes next to such directory partitions. You can also use the following additional options:

- **Select all**: Selects the check boxes next to all directory partitions in the list.
- **Add**: Adds a new directory partition to the list.
- **Remove**: Removes currently selected directory partition from the list.
- To test the connection with the new parameters, click **Test connection**.

Specify naming attributes for a Generic LDAP directory service connection

Every object in an LDAP directory service has a naming attribute from which the object name is formed. When you create a connection to an LDAP directory service, a default naming attribute is selected for each object type in the data system. You can use the **Specify Naming Attributes** item to view the naming attribute currently selected for each object type in the data system and optionally specify a different naming attribute.

This expandable item provides following options:

- **Default naming attribute**: Displays the default naming attribute set for the currently selected object type.
- **Add**: Adds a new naming attribute for the selected object type.
- **Edit**: Allows you to edit the name of the naming attribute currently specified for the selected object type.
- **Remove**: Removes the currently selected entry from the list.

Specify attributes to identify objects for a Generic LDAP directory service connection

This expandable item provides the following options that allow you to specify the attributes with which you wish to uniquely identify each object in the connected LDAP directory service:

- **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
- **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
- **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
- **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
- **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

Specify password sync parameters for LDAP directory service

To synchronize passwords in an LDAP directory service connected to Synchronization Service through the Generic LDAP Connector, you must specify the following parameters:

- The target object type for which you want to synchronize passwords.
- The object attribute for storing passwords in the LDAP directory service.

To specify the target object type and attribute for storing passwords

1. Click the **Connection settings** link below the LDAP directory service connection for which you want to specify the target object type and attribute for storing passwords.
2. Open the **Password** tab.
3. Make sure the **Synchronize and manage passwords** check box is selected.
4. Use the **Synchronize passwords for objects of this type** option to specify the object type in LDAP directory service for which you want to synchronize passwords.
5. Use the **Store password in this attribute** option to specify the attribute in which you want to store passwords.
6. Click **Save**.

Working with an OpenLDAP directory service

This section describes how to create or modify a connection to an OpenLDAP directory service so that Synchronization Service could work with data in that data system.

To create a connection to an OpenLDAP directory service, use the **OpenLDAP Connector** of the Active Roles Synchronization Service.

The OpenLDAP Connector supports the following features:

Table 48: OpenLDAP Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No

Feature	Supported
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Creating an OpenLDAP directory service connection

You can create a new OpenLDAP directory service connection in the Synchronization Service Console.

To create a new OpenLDAP connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
    - **Connection name**: Type a descriptive name for the connection.
    - **Use the specified connector**: Select **OpenLDAP Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
    - **Server**: Type the fully qualified domain name of the computer running an OpenLDAP directory service to which you want to connect.
    - **Port**: Type the number of the communication port used by the OpenLDAP server to which you want to connect.
    - **Access LDAP directory service using**: Type the user name and password of the account with which you want to access the OpenLDAP directory service. Ensure the account has sufficient permissions to perform the operations you want (Read, Write) on objects in the OpenLDAP directory service.
    - **Advanced**: Click this button to specify a number of advanced options to access the OpenLDAP directory service. For example, you can select an authentication method to access the directory service, configure TLS/SSL usage for the connection, and select whether or not you want to use paged search.
    - From this **Authentication method** list, select one of the following methods:
        - **Anonymous**: Allows you to establish the connection without passing credentials.
        - **Basic**: Specifies to use basic authentication.
        - **Microsoft Negotiate**: Specifies to use Microsoft Negotiate authentication.
        - **NTLM**: Specifies to use Windows NT Challenge/Response authentication.
        - **Digest**: Specifies to use Digest Access authentication.

- **Sicily**: Employs a negotiation mechanism (Sicily) to choose the Microsoft Network Authentication Service, Distributed Password Authentication, or NTLM method.
- **Distributed Password Authentication**: Specifies to use DPA authentication.
- **Microsoft Network Authentication Service**: Specifies to authenticate with Microsoft Network Authentication Service.
- **External**: Specifies to use an external authentication method for the connection.
- **Kerberos**: Specifies to use Kerberos authentication.

You can also use the following check boxes:

- **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
- **Switch to TLS/SSL after establishing connection**: Establishes the connection without using the TLS (SSL) encryption. Then, after the connection has been established, enables the TLS (SSL) encryption.
- **Verify TLS/SSL certificate**: Specifies whether or not to check the TLS (SSL) certificate on the server.
- **Use paged search**: Specifies whether or not to use paged search for the connection. When selecting this check box, you can set a page size limit in the text box below.

- To test the connection with the new parameters, click **Test connection**.

5. To complete the configuration of the OpenLDAP directory service connection, click **Finish**.

After establishing a connection, you can define attributes to name objects in the data system. For more information, see Modifying an existing Generic LDAP directory service connection

Modifying an existing OpenLDAP directory service connection

You can modify the various settings for an existing OpenLDAP directory service connection, such as the directory service server, communication port, access credentials, and the attributes used for naming objects in the OpenLDAP directory service.

Every object in an OpenLDAP directory service has a naming attribute from which the object name is formed. When you create a connection to an OpenLDAP directory service, a default naming attribute is selected for each object type in the data system. You can view the default naming attribute currently selected for each object type in the data system and optionally specify a different naming attribute.

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing OpenLDAP connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specifying connection settings for an OpenLDAP directory service connection
   - Specifying naming attributes for an OpenLDAP directory service connection
4. Click **Save**.

Specifying connection settings for an OpenLDAP directory service connection

The **Specify connection settings** option provides the following settings that allow you to modify the connection settings:

- **Server**: Type the fully qualified domain name of the computer running an OpenLDAP directory service to which you want to connect.
- **Port**: Type the number of the communication port used by the OpenLDAP server to which you want to connect.
- **Access LDAP directory service using**: Type the user name and password of the account with which you want to access the OpenLDAP directory service. Ensure the account has sufficient permissions to perform the operations you want (Read, Write) on objects in the OpenLDAP directory service.
- **Advanced**: Click this button to specify a number of advanced options to access the OpenLDAP directory service. For example, you can select an authentication method to access the directory service, configure TLS/SSL usage for the connection, and select whether or not you want to use paged search.
- From this **Authentication method** list, select one of the following methods:
  - **Anonymous**: Allows you to establish the connection without passing credentials.
  - **Basic**: Specifies to use basic authentication.
  - **Microsoft Negotiate**: Specifies to use Microsoft Negotiate authentication.
  - **NTLM**: Specifies to use Windows NT Challenge/Response authentication.
  - **Digest**: Specifies to use Digest Access authentication.
  - **Sicily**: Employs a negotiation mechanism (Sicily) to choose the Microsoft Network Authentication Service, Distributed Password Authentication, or NTLM method.
  - **Distributed Password Authentication**: Specifies to use DPA authentication.

- **Microsoft Network Authentication Service**: Specifies to authenticate with Microsoft Network Authentication Service.
- **External**: Specifies to use an external authentication method for the connection.
- **Kerberos**: Specifies to use Kerberos authentication.

You can also use the following check boxes:

- **Use TLS/SSL**: Allows you to use the TLS (SSL) encryption to establish and maintain the connection.
- **Switch to TLS/SSL after establishing connection**: Establishes the connection without using the TLS (SSL) encryption. Then, after the connection has been established, enables the TLS (SSL) encryption.
- **Verify TLS/SSL certificate**: Specifies whether or not to check the TLS (SSL) certificate on the server.
- **Use paged search**: Specifies whether or not to use paged search for the connection. When selecting this check box, you can set a page size limit in the text box below.

- To test the connection with the new parameters, click **Test connection**.

Specifying naming attributes for an OpenLDAP directory service connection

The **Specify naming attributes** option allows you to specify a naming attribute for each object type in the connected OpenLDAP directory service data system. The option provides the following settings:

- **Default naming attribute**: Displays the default naming attribute set for the currently selected object type.
- **Add**: Adds a new naming attribute for the selected object type.
- **Edit**: Allows you to edit the name of the naming attribute currently specified for the selected object type.
- **Remove**: Removes the currently selected entry from the list.

Working with IBM DB2

This section describes how to create or modify a connection to IBM DB2 so that Synchronization Service could work with data in that data system.

To create a connection to IBM DB2, you need to use Synchronization Service in conjunction with a special connector called IBM DB2 Connector. This connector is included in the Synchronization Service package.

The IBM DB2 Connector supports the following features:

Table 49: IBM DB2 Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Creating an IBM DB2 connection

To create a new connection

1. On the system where Synchronization Service is installed, install IBM Data Server Client supplied with the IBM DB2 version with which you plan to work.
2. In the Synchronization Service Console, open the **Connections** tab.
3. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **IBM DB2 Connector**.
4. Click **Next**.
5. On the **Specify connection settings** page, use the following options:
   - **IBM DB2 server**: Type or select the fully qualified domain name of the IBM DB2 computer that hosts the database you want to participate in data synchronization operations. You can click **Refresh** to get a list of available IBM DB2 servers.
   - **Access IBM DB2 server using**: Type the user name and password with which you want to access the IBM DB2 server.
   - **Connect to database**: Type the name of the database to which you want to connect on the IBM DB2 server.
   - **Advanced**: Optionally, you can click this button to specify additional parameters you want to add to the connection string that will be used to access the IBM DB2 server. In the dialog box that opens, click **Add Parameter** to specify the name and value of the parameter you want to add

to the connection string.

- To test the connection with the new parameters, click **Test connection**.

6. Click **Next**.
7. On the **Specify how to select and modify data** page, use the following options:
   - **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
   - **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.
   - **Configure Settings**: Click this button to specify settings for modifying data in the connected system during synchronization operations. For example, you can specify the database tables in which you want to insert, update, or delete data during synchronization operations.
8. On the **Specify attributes to identify objects** page, use the following options:
   - **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
   - **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
   - **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
   - **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
   - **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.
9. Click **Finish** to create a connection to the IBM DB2 system.

Modifying an existing IBM DB2 connection

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing IBM DB2 connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

You can expand the following items:

- Specify connection settings for an IBM DB2 connection
- Specify how to select and modify data for an IBM DB2 connection
- Advanced
- Specify attributes to identify objects for an IBM DB2 connection

4. Click **Save**.

Specify connection settings for an IBM DB2 connection

This expandable item provides the following options that allow you to modify the connection settings:

- **IBM DB2 server**: Type or select the fully qualified domain name of the IBM DB2 computer that hosts the database you want to participate in data synchronization operations. You can click **Refresh** to get a list of available IBM DB2 servers.
- **Access IBM DB2 server using**: Type the user name and password with which you want to access the IBM DB2 server.
- **Connect to database**: Type the name of the database to which you want to connect on the IBM DB2 server.
- **Advanced**: Optionally, you can click this button to specify additional parameters you want to add to the connection string that will be used to access the IBM DB2 server. In the dialog box that opens, click **Add Parameter** to specify the name and value of the parameter you want to add to the connection string.
- To test the connection with the new parameters, click **Test connection**.

Specify how to select and modify data for an IBM DB2 connection

This expandable item provides the following options that allow you to specify the data you want to participate in the synchronization:

- **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
- **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.
- **Configure Settings**: Click this button to specify settings for modifying data in the connected system during synchronization operations. For example, you can specify the database tables in which you want to insert, update, or delete data during synchronization operations.

Advanced

Allows you to configure the running timeout for all SQL queries you specified in the connection settings (for example, those specified in the **Specify How to Select and Modify Data** option). Use the **SQL query execution timeout** box to type the timeout value you want to use.

Specify attributes to identify objects for an IBM DB2 connection

This expandable item provides the following options that allow you to specify the attributes with which you want to uniquely identify each object in the connected data system:

- **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
- **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
- **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
- **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
- **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

Working with IBM AS/400

This section describes how to create or modify a connection to IBM AS/400 Directory so that Synchronization Service could work with IBM AS/400 Directory data in that data system.

To create a connection to IBM AS/400 Directory, you need to use Synchronization Service in conjunction with a special connector called IBM AS/400 Directory Connector. This connector is included in the Synchronization Service package.

The IBM AS/400 Directory Connector supports the following features:

Table 50: IBM AS/400 Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data	Yes

Feature	Supported
system.	
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Prerequisites

- The IBM AS/400 server must have LDAP directory services installed and configured.
- An LDAP service account must be created on your IBM AS/400 server which has the appropriate permissions to administer users and groups on this platform.

Creating an IBM AS/400 connection

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
    - **Connection name**: Type a descriptive name for the connection.
    - **Use the specified connector**: Select **IBM AS/400 Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
    - **Server**: Enter or select the fully qualified DNS name of the IBM AS/400 server running the LDAP service.
    - **Port**: Enter the IBM AS/400 LDAP communication port number in use by the service.
    - **User name**: Specify the fully distinguished name (DN) of the account under which the application will access the IBM AS/400 LDAP directory service.
    - **Password**: Specify the password of the user account under which the application will access the IBM AS/400 LDAP directory service. One Identity recommends that you select the **SSL** check box if synchronizing sensitive data between connectors.

      NOTE: To successfully establish the connection, set the SSL certificate before enabling this option.
    - To test the connection with the new parameters, click **Test connection**.

5. Click **Next**.
6. Click **Finish** to create a connection to the IBM AS/400 system.

Modifying an existing IBM AS/400 connection

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection Settings** below the existing IBM AS/400 connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
   - **Server**: Enter or select the fully qualified DNS name of the IBM AS/400 server running the LDAP service.
   - **Port**: Enter the IBM AS/400 LDAP communication port number in use by the service.
   - **User name**: Specify the fully distinguished name (DN) of the account under which the application will access the IBM AS/400 LDAP directory service.
   - **Password**: Specify the password of the user account under which the application will access the IBM AS/400 LDAP directory service. One Identity recommends that you select the **SSL** check box if synchronizing sensitive data between connectors.

     NOTE: To successfully establish the connection, set the SSL certificate before enabling this option.
   - To test the connection with the new parameters, click **Test connection**.
4. Click **Save**.

Additional considerations for an IBM AS/400 connection

This topic briefs about the additional points to consider when configuring the IBM AS/400 connector.

Using groups with IBM AS/400

The IBM AS/400 operating system does not have any concept of groups as discrete entities. Instead, an administrator creates a user profile which is used as a group profile. Other user profiles are then linked to this using the `GrpPrf` or `SupGrpPrf` parameters of the `ChgUsrPrf` command. The `GrpPrf` value maps to the `os400-grpprf` attribute in the IBM AS/400 schema, while the `SupGrpPrf` value maps to the `os400-supgrpprf` attribute. The IBM AS/400 Quick

Connect mappings must be defined for users and groups to enable full user and group synchronization.

Optional IBM AS/400 account unlock during password reset function

You can optionally unlock a user's IBM AS/400 account at the same time as performing a password reset. This functionality is switched off by default and can be enabled by editing the connector's configuration file as follows:

1. Edit the `<Program Files folder>\One Identity\Active Roles\7.4\SyncService\AS400Connector_ConnectorConfig.xml` file.
2. Add the following lines just before the `</ConnectorInfo>` which appears on the last line of the file:

   ```
   <SelfConfig>
   <EnableAccount>true</EnableAccount>
   </SelfConfig>
   ```

   NOTE: Only the value `true` will enable the new functionality.

The LDAP password request sent to IBM AS/400 will then also include a request to modify the account status (`os400-status=*ENABLED)`).

The configuration file is read every time an LDAP connection is made to the IBM AS/400, so the new value will be picked up for the next set of synchronizations.

NOTE: If you edited `ConnectorConfig.xml` to implement the optional unlock of a user's IBM AS/400 account at the same time as performing a password reset in an earlier version of the connector for IBM AS/400, then you will need to repeat that edit after installing a later version.

Working with IBM RACF

To create a connection to IBM RACF connector, you need to use Synchronization Service in conjunction with a special connector called IBM RACF Connector. This connector is included in the Synchronization Service package.

The IBM RACF Connector supports the following features:

Table 51: IBM RACF Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode	No

Feature	Supported
Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Prerequisites

- The IBM mainframe must have LDAP directory services installed and configured.
- The IBM RACF connector can be installed on Microsoft Windows Server 2016 or later.

NOTE: There is an 8 character limit for user and group names on IBM RACF. The character limit is also applicable to the passwords on IBM RACF.

Creating an IBM RACF connection

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **IBM RACF Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Server**: Type the fully qualified DNS name of the IBM RACF server running the LDAP service. Type the fully qualified DNS name of the IBM RACF server running the LDAP service.
   - **Port**: Type the fully qualified DNS name of the IBM RACF server running the LDAP service.
   - **User name**: Specify the fully distinguished name (DN) of the account that the application will use to access the IBM RACF LDAP directory service
   - **Password**: Specify the password of the user account that the application will use to access the IBM RACF LDAP directory service.
   - To test the connection with the new parameters, click **Test connection**.
5. Click **Next**.
6. Click **Finish**to create a connection to IBM RACF connector.

Modifying an IBM RACF connection

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection Settings** below the existing IBM RACF connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
   - **Server**: Type the fully qualified DNS name of the IBM RACF server running the LDAP service. Type the fully qualified DNS name of the IBM RACF server running the LDAP service.
   - **Port**: Type the fully qualified DNS name of the IBM RACF server running the LDAP service.
   - **User name**: Specify the fully distinguished name (DN) of the account that the application will use to access the IBM RACF LDAP directory service
   - **Password**: Specify the password of the user account that the application will use to access the IBM RACF LDAP directory service.
   - To test the connection with the new parameters, click **Test connection**.
4. Click **Save**.

Example of mapping for dataset information

The IBM RACF connector can be used to synchronize IBM RACF dataset information. The LDAPX exit must be installed and configured for this functionality to be supported.

The examples in this topic shows how IBM RACF dataset information can be synchronized. IBM RACF dataset names contain asterisk (*) characters and as such cannot be synchronized to Active Directory which does not allow asterisk characters in names. As such, the example shows a synchronization to a Microsoft SQL database. It is assumed that Microsoft SQL Server and Microsoft SQL Server Manager have been installed and configured.

Creating SQL database and table

Using Microsoft SQL Server Manager, create a database called **IBM RACF_Datasets**. Within that database, create a table called **Datasets** with the following columns:

Column Name	Data Type
Audit	`nchar(100)`

Create_Group	`nchar(10)`
Owner	`nchar(10)`
UACC	`nchar(10)`
UID (database key)	`nchar(100)`

Create a connection to this database and table with the ARSS Microsoft SQL Server Connector.

Povisioning datasets

To synchronize the SQL table to IBM RACF follow the steps provided here.

To synchronize the SQL table to IBM RACF

1. Navigate to the **Workflows** tab.
2. Click **Add sync workflow**.
3. Enter **IBM RACF Datasets** and click **OK**.
4. Click **IBM RACF Datasets workflow**.
5. Click **Add synchronization step**.
6. Click **Creation** and then **Next**.
7. From the **Source connected system** section and click **Specify**.
8. Select your Microsoft SQL Server Connector and click **Finish**.

   The SQL source object type is currently set to `sql-Object`. Do not change this value.
9. Click **Next**.
10. In the **Target connected system** field, click **Specify** and then locate your IBM RACF connector and click **Finish**.
11. The object type in the **Target object system** field is populated automatically by Synchronization Service to **racfUser**. Change this to **racfDataset**.
12. Click **Next**.
13. In the **Specify provisioning** rules section, click **Forward Sync Rule**.
14. In the **Source attribute** field, click **Attribute** locate **UID** and click **OK**.
15. In the **Target attribute** field, click **Attribute**, locate **racfDataset** and click **OK**.
16. Repeat these steps so that the following five items are mapped:

SQL Attribute	IBM RACF Attribute

`Owner`	`racfOwner`
`UACC`	`racfUacc`
`Create_Group`	`racfCreateGroup`
`Audit`	`racfAudit`
`UID`	`racfDataset`

17. Click **OK**.
18. Click **Finish** to complete the synchronization.

Updating datasets

To synchronize the SQL table to IBM RACF

1. Navigate to the **Sync Workflows** tab, select **IBM RACF Datasets** and click **OK**.
2. Click **Add synchronization step**.
3. Click **Update** and then click **Next**.
4. From the **Source connected system** section and click **Specify**.
5. Select your Microsoft SQL Server Connector and click **Finish**.

   The SQL source object type is currently set to `sql-Object`. Do not change this value.
6. Click **Next**.
7. In the **Target connected system** field, click **Specify** and then locate your IBM RACF connector and click **Finish**.
8. The object type in the **Target object system** field is populated automatically by Synchronization Service to **racfUser**. Change this to **racfDataset**.
9. Click **Next**.
10. In the **Specify provisioning** rules section, click **Forward Sync Rule**.
11. In the **Source attribute** field, click **Attribute** locate **UID** and click **OK**.
12. In the **Target attribute** field, click **Attribute**, locate **racfDataset** and click **OK**.
13. Repeat these steps so that the following five items are mapped:

SQL Attribute	IBM RACF Attribute
`Owner`	`racfOwner`
`UACC`	`racfUacc`
`Create_Group`	`racfCreateGroup`

Audit	racfAudit
UID	racfDataset

14. Click **OK**.
15. Click **Finish** to complete the synchronization.

Deprovisioning datasets

To deprovision datasets

1. Navigate to the **Workflows** tab and select **IBM RACF Datasets**.
2. Click **Add synchronization step**.
3. Click **Deprovision** and then click **Next**.
4. From the **Source connected system** section and click **Specify**.
5. Select your Microsoft SQL Server Connector and click **Finish**.
6. Select **Source object is deleted or is out of synchronization scope** option in the **Deprovision target objects if** section.
7. Optionally, configure the **Source object meets the following criteria**.
8. Click **Next**.
9. In the **Target connected system** field, click **Specify** and then locate your IBM RACF connector and click **Finish**.
10. The object type in the **Target object system** field is populated automatically by Synchronization Service to **racfDataset**.
11. Click **Next**.
12. Select **Delete target object**.
13. Click **Finish** to complete the synchronization.

Working with TSO command

The IBM RACF connector can be used to run any command in the Time Sharing Option (TSO) environment on the target IBM mainframe. The LDAPX exit must be installed and configured for this functionality to be supported.

The TSO command is run using an Active Roles Synchronization Service synchronization step to create an object of type **`ldapxtsocmd`** on the target IBM RACF system and supplying the name of the TSO command or script to be run in the attribute `racfprogrammername`. When the step is run, the IBM RACF connector intercepts the create command and instead sends an LDAP search command with the required parameters via the LDAP protocol.

The LDAPX exit intercepts this request, extracts the TSO command information and runs the command. The LDAP response is constructed, containing the results obtained from

running the command. The IBM RACF connector receives this LDAP response, extracts the results and saves them in a text file that can be examined later.

No object is created during the synchronization step so it can be run indefinitely, each time executing the TSO command stored in the `racfprogrammername` attribute from the same or any other synchronization step.

The following example shows a method of issuing a TSO command using synchronisation from Active Directory (AD).

1. Using **Active Directory Users and Computers**, create a container in AD that can be filtered on by the ARSS. For example, create an organisational unit container called **TSO Commands**.
2. Create a dummy computer object within this container with name **TSOCMD** and description field set to the string **STATUS**. The TSO command **STATUS** will return the current system status.
3. Create a workflow called **Run TSO Command**.
4. Within this workflow, create a synchronization step item as follows:
   a. Synchronization step type: **Create**
   b. **Source object**: Active Directory, specified container as created above, name starts with **TSOCMD**.
   c. Target connector: **IBM RACF**
   d. Object type: **ldapxtsocmd**
   e. Mapping: from AD **Description** attribute to IBM RACF **racfprogrammername** attribute
5. Save the step.
6. Run the synchronization step. There should be one item to be created with the following properties:
   - objecttype: **ldapxtsocmd**
   - racfprogrammername: **STATUS**
7. Perform the synchronization step.
8. The LDAP command will be sent and interpreted by the LDAPX exit to run the TSO command.
9. Once complete, the synchronization step will show as being successful.
10. The output from running the command can be found in the following text file:

    `<ARSS installation folder>\SyncService\TSOCommandOutput\YYDDMM.txt`, where, `YYMMDD` represents the date when the command was run.
11. The text file will contain the output returned from IBM RACF having run the **STATUS** command.
12. Multiple commands run on the same day will have their output appended to the same daily text file.

Working with MySQL database

This section describes how to create or modify a connection to a MySQL database so that Synchronization Service could work with data in that data system.

To create a MySQL database connection, use the **MySQL Connector** of the Active Roles Synchronization Service.

The MySQL Connector supports the following features:

Table 52: MySQL Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	Yes

Creating a MySQL database connection

You can create a new MySQL database connection in theSynchronization Service Console.

Prerequisites

Before configuring the connector, make sure that the Connector/NET fully-managed ADO.NET driver is installed on the machine running the Synchronization Service.

To create a new connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
    - **Connection name**: Type a descriptive name for the connection.
    - **Use the specified connector**: Select **MySQL Connector**.
3. Click **Next**.

4. On the **Specify connection settings** page, use the following options:
    - **MySQL server**: Type the fully qualified domain name of the MySQL server that hosts the MySQL database that you want to participate in data synchronization operations.
    - **Access MySQL server using**: Type the user name and password of the account with which you want to access MySQL server. Ensure the account has sufficient permissions to perform operations (Read, Write) on objects in the database to which you want to connect.
    - **Connect to database**: Type the name of the database to which you want to connect on the MySQL server.
    - **Advanced**: Click this button to specify additional parameters you want to add to the connection string that will be used to access the MySQL server. In the dialog box that opens, click the Add Parameter button to specify the name and value of the parameter you want to add to the connection string.
    - To test the connection with the new parameters, click **Test connection**.
5. Click **Next**.
6. On the **Specify how to select and modify data** page, use the following options:
    - **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
    - **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.
    - **Configure Settings**: Click this button to specify settings for modifying data in the connected system during synchronization operations. For example, you can specify the database tables in which you want to insert, update, or delete data during synchronization operations.
7. Click **Next**.
8. On the **Specify attributes to identify objects** page, use the following options:
    - **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
    - **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
    - **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
    - **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.

- **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

9. To complete the configuration of the MySQL database connection, click **Finish**.

Modifying an existing MySQL database connection

You can modify the settings of an existing **MySQL Connector** with theSynchronization Service Console.

To modify connection settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing MySQL connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specifying connection settings for a MySQL database connection
   - Specifying how to select and modify data for a MySQL database connection
   - Advanced
   - Specifying attributes to identify objects for a MySQL database connection
4. Click **Save**.

Specifying connection settings for a MySQL database connection

The **Specify connection settings** option provides the following options that allow you to modify the connection settings:

- **MySQL server**: Type the fully qualified domain name of the MySQL server that hosts the MySQL database that you want to participate in data synchronization operations.
- **Access MySQL server using**: Type the user name and password of the account with which you want to access MySQL server. Ensure the account has sufficient permissions to perform operations (Read, Write) on objects in the database to which you want to connect.
- **Connect to database**: Type the name of the database to which you want to connect on the MySQL server.
- **Advanced**: Click this button to specify additional parameters you want to add to the connection string that will be used to access the MySQL server. In the dialog box that opens, click the Add Parameter button to specify the name and value of the

parameter you want to add to the connection string.

- To test the connection with the new parameters, click **Test connection**.

Specifying how to select and modify data for a MySQL database connection

The **Specify how to select and modify data** setting provides the following options that allow you to specify the data you want to participate in the synchronization:

- **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
- **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.
- **Configure Settings**: Click this button to specify settings for modifying data in the connected system during synchronization operations. For example, you can specify the database tables in which you want to insert, update, or delete data during synchronization operations.

Advanced

Allows you to configure the running timeout for all SQL queries you specified in the connection settings (for example, those specified in the **Specify How to Select and Modify Data** option). Use the **SQL query execution timeout** box to type the timeout value you want to use.

Specifying attributes to identify objects for a MySQL database connection

The **Specify attributes to identify objects** setting provides the following options that allow you to specify the attributes with which you want to uniquely identify each object in the connected data system:

- **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
- **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
- **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.

- **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
- **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

Working with an OLE DB-compliant relational database

This section describes how to create or modify a connection to an OLE DB-compliant relational database so that Synchronization Service could work with data in that database.

To create a connection to an OLE DB-compliant relational database, use the **OLE DB Connector** of the Active Roles Synchronization Service.

> NOTE: To create a connection to an OLE DB-compliant relational database, the **OLE DB Connector** requires any version of Microsoft OLE DB Driver for SQL Server that is supported by Microsoft to be installed on the machine running Active Roles Synchronization Service.
>
> The Active Roles Synchronization Service installer is shipped with and automatically installs Microsoft OLE DB Driver 19.x for SQL Server.

Table 53: OLE DB Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	No NOTE: By using OLE DB Connector, you can only read data in the connected data system.
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No

Creating an OLE DB-compliant relational database connection

You can create a new OLE DB-compliant database connection in the Synchronization Service Console.

To create a new OLE DB-compliant relational database connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**. Select **OLE DB Connector**.
3. Click **Next**.
4. Use the **Connection string** text box to type the connection parameters to access the OLE DB-compliant relational database. Alternatively, you can click **Configure** to specify the connection parameters by using a dialog provided by Windows.
5. Click **Next**.
6. On the **Specify how to select and modify data** page, use the following options:
   - **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
   - **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.
7. Click **Next**.
8. On the **Specify attributes to identify objects** page, use the following options:
   - **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
   - **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
   - **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
   - **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
   - **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

9. To finish configuring the connection to the OLE DB-compliant relational database, click **Finish**.

Modifying an existing OLE DB-compliant data source connection

You can modify an existing OLE DB-compliant database connection in the Active Roles Synchronization Service Console.

To modify the connection settings of an OLE DB Connector

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing OLE DB-compliant relational database connection you want to modify.
3. On the **Connection Settings** tab, click an appropriate item to expand it and use the options it provides.

   You can expand the following items:

   - Specifying connection settings for an OLE DB-compliant relational database connection
   - Specifying how to select data for an OLE DB-compliant relational database connection
   - Advanced
   - Specifying attributes to identify object for an OLE DB-compliant relational database connection
4. For more information on these settings, see the applicable subsections.
5. When you are finished, click **Save**.

Specifying connection settings for an OLE DB-compliant relational database connection

Use the **Connection string** text box to type the connection parameters to access the OLE DB-compliant relational database. Alternatively, you can click **Configure** to specify the connection parameters by using a dialog provided by Windows.

Specifying how to select data for an OLE DB-compliant relational database connection

The **Specify how to select data** setting provides the following options that allow you to specify the data you want to participate in the synchronization:

- **Use data from this table**: Allows you to select a database table that includes the data you want to participate in the synchronization operations. You can click **Preview** to preview the database table you have selected.
- **Use an SQL query to specify data**: Allows you to compose an SQL query that provides a more flexible way for specifying the data for synchronization. For example, you can use this option to specify multiple database tables.

Advanced

Allows you to configure the running timeout for all SQL queries you specified in the connection settings (for example, those specified in the **Specify How to Select and Modify Data** option). Use the **SQL query execution timeout** box to type the timeout value you want to use.

Specifying attributes to identify object for an OLE DB-compliant relational database connection

The **Specify attributes to identify object** provides the following options that allow you to specify the attributes with which you want to uniquely identify each object in the connected data system:

- **Available attributes**: Lists the attributes that are available in the external data system. Use this list to select the attributes whose values you want to use to generate a unique identifier for each object in the external data system. You can filter attributes by typing in the text box at the top of this list. To select multiple attributes, hold down **CTRL** and click to select attributes in the list.
- **UniqueID attributes**: Lists the attributes whose values are currently used to generate a unique identifier for each object in the external data system.
- **Add**: Moves the selected attributes from the **Available attributes** list to the **UniqueID attributes** list.
- **Remove**: Moves the selected attributes from the **UniqueID attributes** list to the **Available attributes** list.
- **Constructed UniqueID**: Displays a combination of the attributes whose values will make up a unique identifier for each object in the external data system.

Working with SharePoint

This section describes how to create or modify a connection to Microsoft SharePoint so that Synchronization Service could work with data in that data system.

To create a connection to Microsoft SharePoint, use the **Sharepoint Connector** of Active Roles Synchronization Service.

The SharePoint Connector supports the following features:

Table 54: SharePoint Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No

Creating a SharePoint connection

You can create a new SharePoint connection in the Synchronization Service Console.

To create a new SharePoint connection

1. Ensure that you have installed the SharePoint Connector on the SharePoint server you want to work with.
2. In the Synchronization Service Console, open the **Connections** tab.
3. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**. Select **SharePoint Connector**.
4. Click **Next**.
5. To check that the connector can access SharePoint, on the **Specify connection settings** page, click the **Test Connection**.
6. If the test succeeds, click **Finish** to create a connection.

SharePoint data supported for data synchronization

The following table lists the data objects and data operations supported by the SharePoint Connector.

Synchronization Service provides special attributes for each supported SharePoint object type, allowing you to read or write data in SharePoint. You can access and use these attributes from the Synchronization Service Console, for example, when selecting the source and target attributes to include in the synchronization operation.

Table 55: Supported objects and operations

Object	Read	Create	Delete	Update
AlternateURL Allows you to read data related to an incoming URL and the zone with which it is associated.	Yes	No	No	No
ClaimProvider Allows you to read data related to a claim provider.	Yes	No	No	No
Farm Allows you to work with a SharePoint farm.	Yes	No	No	No
Group Allows you to work with a group on a SharePoint website.	Yes	Yes	Yes	Yes
Language Allows you to work with a language used in SharePoint.	Yes	No	No	No
Policy Allows you to work with a policy assigned to a user or group.	Yes	Yes	Yes	Yes
PolicyRole Allows you to work with the rights possessed by a policy role.	Yes	Yes	Yes	Yes
Prefix Allows you to work with a relative URL that determines segments of the URL under which sites may be created.	Yes	No	No	No
RoleAssignment Allows you to work with role assignments for a user or group.	Yes	Yes	Yes	Yes
RoleDefinition Allows you to work with a role definition, including name, description, management properties, and a set of rights.	Yes	Yes	Yes	Yes
Site Allows you to work with site collections in an Internet Information Services (IIS) web	Yes	Yes	Yes	Yes

Object	Read	Create	Delete	Update
application.				
User Allows you to work with a user in SharePoint.	Yes	Yes	Yes	Yes
Web Allows you to work with a SharePoint website.	Yes	Yes	Yes	Yes
WebApplication Allows you to work with an IIS load-balanced web application installed on a server farm.	Yes	No	No	Yes
WebTemplate Allows you to work with a site definition configuration or a web template used to create SharePoint sites.	Yes	No	No	No

The following sections describe the attributes provided by Synchronization Service and describe what data you can read or write in SharePoint by using a particular attribute.

AlternateURL object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `AlternateURL` object with the following synchronization operations.

Table 56: AlternateURL object attributes

Attribute	Type	Description	Supported operations
`Id`	Single-valued, string	Gets the object ID.	Read
`IncomingUrl`	Single-valued, string	Gets the incoming URL that is associated with the zone from which the request originated.	Read
`Parent`	Single-valued, string, reference (`WebApplication` object)	Gets the parent of the object.	Read
`Uri`	Single-valued, string	Gets the incoming URL associated with the zone from which the request originated, in the form of an URI.	Read
`UrlZone`	Single-valued, string	Gets the zone that is associated with the alternate request URL.	Read

ClaimProvider object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `ClaimProvider` object with the following synchronization operations.

Table 57: ClaimProvider object attributes

Attribute	Type	Description	Supported operations
`AssemblyName`	Single-valued, string	Gets the name of the assembly that implements the claims provider.	Read
`Description`	Single-valued, string	Gets the description of the claims provider.	Read
`DisplayName`	Single-valued, string	Gets the display name of the claims provider.	Read
`Id`	Single-valued, string	Gets the object ID.	Read
`IsEnabled`	Single-valued, Boolean	Gets whether the claims provider is enabled.	Read
`IsUsedByDefault`	Single-valued, Boolean	Gets whether the claims provider applies by default to all web applications and zones.	Read
`IsValid`	Single-valued, Boolean	Gets whether the claims provider is valid.	Read
`IsVisible`	Single-valued, Boolean	Gets whether the claims provider is visible.	Read
`Parent`	Single-valued, string, reference (`Farm` object)	Gets the parent of the object.	Read
`TypeName`	Single-valued, string	Gets the type of the object.	Read

Farm object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `Farm` object with the following synchronization operations.

Table 58: Farm object attributes

Attribute	Type	Description	Supported operations
BuildVersion	Single-valued, string	Gets the build version of the SharePoint server farm.	Read
CanBackupRestoreAsConfiguration	Single-valued, Boolean	Gets whether the farm can participate in a configuration-only backup or restore.	Read
CanRenameOnRestore	Single-valued, Boolean	Gets whether the farm can be renamed during its restore.	Read
CanSelectForBackup	Single-valued, Boolean	Gets whether the farm can be selected for backup.	Read
CanSelectForRestore	Single-valued, Boolean	Gets whether the farm can be selected for restore in the Central Administration user interface.	Read
DaysBeforePasswordExpirationToSendEmail	Single-valued, integer	Gets the number of days before password expiration when a notification email is sent.	Read
DefaultServiceAccount	Single-valued, string	Gets the default service account.	Read
EncodedFarmId	Single-valued,	Gets the farm identifier.	Read

Attribute	Type	Description	Supported operations
	integer		
`Id`	Single-valued, string	Gets the object ID.	Read
`Name`	Single-valued, string	Gets the farm name.	Read
`Parent`	Single-valued, string	Gets the parent of the object.	Read
`PasswordChangeEmailAddress`	Single-valued, string	Gets the email address that receives password change notification messages.	Read
`PasswordChangeGuardTime`	Single-valued, integer	Gets the time interval (in seconds) that is used to wait for other computers' response during password change operations.	Read
`PasswordChangeMaximumTries`	Single-valued, integer	Gets the maximum allowed number of password change attempts before the operation fails.	Read
`PersistedFileChunkSize`	Single-valued, integer	Gets the chunk size used to transfer files to	Read

Attribute	Type	Description	Supported operations
		or from the configuration database during a read or write operation.	
`Products`	Multivalued, string	Gets the identifiers of products installed in the farm.	Read
`ServerDebugFlags`	Multivalued, integer	Gets server debug flags.	Read
`Servers`	Multivalued, string	Gets the physical servers that are included in the farm.	Read
`TimerService`	Single-valued, string	Gets the timer service that is used by the farm.	Read
`TraceSessionGuid`	Single-valued, string	Gets the GUID that is used for trace session registration.	Read
`UseMinWidthForHtmlPicker`	Single-valued, Boolean	Gets the HTML select control.	Read
`UserLicensingEnabled`	Single-valued, Boolean	Gets whether user licensing is enabled.	Read
`XsltTransformTimeOut`	Single-valued, integer	Gets the timeout period (in seconds) for a customized XSLT transformation operation.	Read

Group object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `Group` object with the following synchronization operations.

Table 59: Group object attributes

Attribute	Type	Description	Supported operations
`AllowMembersEditMembership`	Single-valued, Boolean	Gets or sets whether group membership can be modified by the group members.	Read, write (update only)
`AllowRequestToJoinLeave`	Single-valued, Boolean	Gets or sets whether users can request to join or leave the group.	Read, write (update only)
`AutoAcceptRequestToJoinLeave`	Single-valued, Boolean	Gets or sets whether users are automatically added or removed from the group upon their request.	Read, write (update only)
`CanCurrentUserEditMembership`	Single-valued, Boolean	Gets whether the current user can modify membership of the group.	Read
`CanCurrentUserManageGroup`	Single-valued, Boolean	Gets whether the current user can manage the group.	Read
`CanCurrentUserViewMembership`	Single-valued, Boolean	Gets whether the current user can view a list of group members.	Read
`ContainsCurrentUser`	Single-valued, Boolean	Gets whether the group contains the current user.	Read
`Description`	Single-valued, string	Gets or sets the group description.	Read, write (update only)
`DistributionGroupAlias`	Single-valued, string	Gets the distribution group alias for the group.	Read

Attribute	Type	Description	Supported operations
`DistributionGroupEmail`	Single-valued, string	Gets the distribution group email.	Read
`DistributionGroupErrorMessage`	Single-valued, string	Gets the last error message encountered during an asynchronous distribution group operation.	Read
`ExplicitlyContainsCurrentUser`	Single-valued, Boolean	Gets whether the group explicitly contains the current user as a direct member.	Read
`Id`	Single-valued, string	Gets the object ID.	Read
`LoginName`	Single-valued, string	Gets the login name of the group.	Read
`Name`	Single-valued, string	Gets or sets the name of the group.	Read, write
`OnlyAllowMembersViewMembership`	Single-valued, Boolean	Gets or sets whether only group members can view the list of members for the group.	Read, write (update only)
`Owner`	Single-valued, string, reference (`User` or `Group` object)	Gets or sets the group owner. A group owner can be a user or another group.	Read, write (create only)
`Parent`	Single-valued, string, reference (`Site` object)	Gets the parent of the object.	Read
`RequestToJoinLeaveEmailSetting`	Single-valued, string	Gets or sets the email address that receives requests to join or leave the group.	Read, write (update only)
`Users`	Multivalued,	Gets or sets the	Read, write

Attribute	Type	Description	Supported operations
	string, reference (User object)	users that are members of the group.	(update only)
Xml	Single-valued, string	Gets the group properties in the XML string format.	Read

Language object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the Language object with the following synchronization operations.

Table 60: Language object attributes

Attribute	Type	Description	Supported operations
DisplayName	Single-valued, string	Gets the language name displayed on the user interface.	Read
Id	Single-valued, string	Gets the object ID.	Read
LanguageTag	Single-valued, string	Gets the language tag.	Read
Parent	Single-valued, string	Gets the parent of the object.	Read

Policy object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the Policy object with the following synchronization operations.

Table 61: Policy object attributes

Attribute	Type	Description	Supported operations
Alias	Single-valued, string	Gets the alias of the object.	Read
DisplayName	Single-valued, string	Gets or sets the display name of the policy.	Read, write (update only)
Id	Single-valued,	Gets the object ID.	Read

Attribute	Type	Description	Supported operations
	string		
`IsSystemUser`	Single-valued, Boolean	Gets or sets whether the user identified by the policy is represented as a system account in the user interface.	Read, write (update only)
`Parent`	Single-valued, string, reference (`WebApplication` object)	Gets the parent of the object.	Read
`PolicyRoleBindings`	Single-valued, string, reference (`PolicyRole` object)	Gets or sets policy roles for the policy.	Read, write (update only)
`UrlZone`	Single-valued, string	Gets or sets the originating zone of an incoming request.	Read, write (create only)
`UserName`	Single-valued, string	Gets the user name of the user or group associated with the policy.	Read, write (create only)

PolicyRole object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `PolicyRole` object with the following synchronization operations.

Table 62: PolicyRole object attributes

Attribute	Type	Description	Supported operations
`DenyRightsMask`	Multivalued, string	Gets or sets the rights which the policy role denies.	Read, write (update only)
`Description`	Single-valued, string	Gets or sets the policy role description.	Read, write (update only)
`GrantRightsMask`	Multivalued, string	Gets or sets the rights which the policy role grants.	Read, write (update only)
`Id`	Single-valued, string	Gets the policy role GUID.	Read
`IsSiteAdmin`	Single-valued, Boolean	Gets or sets whether the policy role grants site collection administrator status.	Read, write (update only)

Attribute	Type	Description	Supported operations
`IsSiteAuditor`	Single-valued, Boolean	Gets or sets whether the policy role grants site collection auditor status.	Read, write (update only)
`Name`	Single-valued, string	Gets or sets the policy role name.	Read, write (update only)
`Parent`	Single-valued, string, reference (`WebApplication` object)	Gets the parent of the object.	Read
`Type`	Single-valued, string	Gets the type of the policy role.	Read
`Xml`	Single-valued, string	Gets the policy role in the XML string format.	Read

Prefix object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `Prefix` object with the following synchronization operations.

Table 63: Prefix object attributes

Attribute	Type	Description	Supported operations
`Id`	Single-valued, string	Gets the object ID.	Read
`Name`	Single-valued, string	Gets the server-relative URL of the prefix without the leading forward slash.	Read
`Parent`	Single-valued, string, reference (`WebApplication` object)	Gets the parent of the object.	Read
`PrefixType`	Single-valued, string	Gets the type of the prefix.	Read

RoleAssignment object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `RoleAssignment` object with the following synchronization operations.

Table 64: RoleAssignment object attributes

Attribute	Type	Description	Supported operations
`Alias`	Single-valued, string	Gets the object alias.	Read
`Id`	Single-valued, string	Gets the object ID.	Read
`Member`	Single-valued, string, reference (`Role` or `Group` object)	Gets the user or group for the role assignment. This attribute is required to create a new `RoleAssignment` object in SharePoint.	Read
`Parent`	Single-valued, string, reference (`Web` object)	Gets the parent for the role assignment.	Read
`RoleDefinitionBindings`	Single-valued, string, reference (`RoleDefinition` object)	Gets the role definition bindings for the role assignment.	Read, write (update only)

RoleDefinition object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `RoleDefinition` object with the following synchronization operations.

Table 65: RoleDefinition object attributes

Attribute	Type	Description	Supported operations
`BasePermissions`	Multivalued, string	Gets or sets the base permissions for a role definition.	Read, write (update only)
`Description`	Single-valued, string	Gets or sets the role definition description.	Read, write (update only)
`Hidden`	Single-valued, Boolean	Gets whether the role definition is displayed in the user interface.	Read
`Id`	Single-valued, string	Gets the object identifier.	Read
`Members`	Multivalued, string, reference	Gets or sets role assignments for the role definition.	Read, write (update only)

Attribute	Type	Description	Supported operations
`Name`	Single-valued, string	Gets or sets the role definition name.	Read, write
`Order`	Single-valued, string	Gets or sets the order in which to display the permission levels in the user interface.	Read, write (update only)
`Parent`	Single-valued, string, reference	Gets the object parent.	Read
`Type`	Single-valued, string	Gets the role definition type.	Read
`Xml`	Single-valued, string	Gets the role definition permission in the XML format.	Read

Site object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `Site` object with the following synchronization operations.

Table 66: Site object attributes

Attribute	Type	Description	Supported operations
`AdministrationSiteType`	Single-valued, string	Gets or sets the administration site types supported by SharePoint.	Read, write (update only)
`AllowDesigner`	Single-valued, Boolean	Gets or sets the **Site Collection Allow Designer** property.	Read, write (update only)
`AllowExternalEmbedding`	Single-valued, string	Gets or sets the external domain embedding for the site collection.	Read, write (update only)
`AllowMasterPageEditing`	Single-valued, Boolean	Gets whether master page editing is allowed.	Read

Attribute	Type	Description	Supported operations
AllowRevertFromTemplate	Single-valued, Boolean	Gets or sets whether reverting from a template is allowed.	Read, write (update only)
AllowRssFeeds	Single-valued, Boolean	Gets whether the site collection allows RSS feeds.	Read
AllowSelfServiceUpgrade	Single-valued, Boolean	Gets or sets whether upgrade is allowed.	Read, write (update only)
AllowSelfServiceUpgradeEvaluation	Single-valued, Boolean	Gets or sets whether upgrade evaluation site collection can be created.	Read, write (update only)
AllowUnsafeUpdates	Single-valued, Boolean	Gets or sets whether updates to the database are allowed without security validation.	Read, write (update only)
ApplicationRightsMask	Multivalued, string	Gets the rights mask for the parent web application of the site collection.	Read
Archived	Single-valued, Boolean	Gets or sets whether the site is in archived mode.	Read, write (update only)
AuditLogTrimmingCallout	Single-valued, string	Gets or sets the class name of the object that performs audit log trimming.	Read, write (update only)
AuditLogTrimmingRetention	Single-valued, integer	Gets or sets the period (in days) during which the audit log data is retained.	Read, write (update only)
AverageResourceUsage	Single-valued, string	Gets the average resource usage of the site collection for the specified number of days.	Read

Attribute	Type	Description	Supported operations
`BrowserDocumentsEnabled`	Single-valued, Boolean	Gets whether the documents can be viewed in a web browser.	Read
`CanUpgrade`	Single-valued, Boolean	Gets whether the object is upgradeable.	Read
`CatchAccessDeniedException`	Single-valued, Boolean	Gets or sets whether SharePoint handles **Access denied** exceptions.	Read, write (update only)
`CertificationDate`	Single-valued, DateTime	Gets the confirmation date and time for the automatic deletion of the site collection.	Read
`CompatibilityLevel`	Single-valued, integer	Gets the major version number of the site collection. This version number is used to perform compatibility checks.	Read
`ContentDatabase`	Single-valued, string	Gets the content database associated with the site collection.	Read
`CurrentChangeToken`	Single-valued, string	Gets the change token that is used to write the next change to the site collection.	Read
`CurrentResourceUsage`	Single-valued, string	Gets the resource usage for the site collection.	Read
`DeadWebNotificationCount`	Single-valued, integer	Gets the number of notifications that were sent about the websites that are not in use within the site collection.	Read
`DenyPermissionsMask`	Multivalued, string	Gets or sets the deny permission mask for all site users, including the site administrator.	Read, write (update only)
`EvalSiteId`	Single-	Gets the identifier of the	Read

Attribute	Type	Description	Supported operations
	valued, string (GUID)	upgrade evaluation site collection, if it was created for the site collection.	
ExpirationDate	Single-valued, DateTime	Gets or sets the date after which an upgrade evaluation site collection gets automatically deleted.	Read, write (update only)
FileNotFoundUrl	Single-valued, string	Gets the URL to the file not found page. The HTTP requests where the resource cannot be found are redirected to this URL.	Read, write (update only)
HasAppPrincipalContext	Single-valued, Boolean	Gets whether the object is running within an application principal context.	Read
HideSystemStatusBar	Single-valued, Boolean	Gets whether the system status bar of the site is hidden.	Read
HostHeaderIsSiteName	Single-valued, Boolean	Gets whether the host header is used to uniquely identify the site collection.	Read
HostName	Single-valued, string	Gets the name of the server that hosts the site collection.	Read
Id	Single-valued, string	Gets the object ID.	Read
IISAllowsAnonymous	Single-valued, Boolean	Gets a value that indicates whether Internet Information Services (IIS) allows anonymous access.	Read
Impersonating	Single-valued, Boolean	Gets the impersonation status of the object.	Read
InheritAllowSelfServiceUpgradeEvaluationSetting	Single-valued,	Gets or sets whether to inherit the	Read, write

Attribute	Type	Description	Supported operations
	Boolean	`AllowSelfServiceUpgradeEvaluationSetting` value from the parent.	(update only)
`InheritAllowSelfServiceUpgradeSetting`	Single-valued, Boolean	Gets or sets whether to inherit the AllowSelfServiceUpgradeSetting value from the parent.	Read, write (update only)
`InvitedUserMaximumLevel`	Single-valued, integer	Description is not available.	Read, write (update only)
`IsEvalSite`	Single-valued, Boolean	Gets or sets whether the object is an upgrade evaluation site collection.	Read, write (update only)
`IsReadLocked`	Single-valued, Boolean	Gets or sets whether the site collection is unavailable for Read access.	Read, write (update only)
`Language`	Single-valued, integer, reference	Description is not available.	Read, write
`LastContentModifiedDate`	Single-valued, DateTime	Gets the date and time (in UTC) when the site content was last modified.	Read
`LastSecurityModifiedDate`	Single-valued, DateTime	Gets the date and time (in UTC) when the site collection security settings were last modified.	Read
`LockIssue`	Single-valued, string	Gets or sets the comment that was written when the site collection was locked.	Read, write (update only)
`MaintenanceMode`	Single-valued, Boolean	Gets whether the site is in maintenance mode.	Read

Attribute	Type	Description	Supported operations
`NeedsUpgrade`	Single-valued, Boolean	Gets or sets whether the site requires upgrading.	Read, write (update only)
`OutgoingEmailAddress`	Single-valued, string	Gets or sets the outgoing email address for the site.	Read, write (update only)
`Owner`	Single-valued, string, reference (`User` object)	Gets or sets the site collection owner. NOTE: This attribute is required to create a new site collection in SharePoint.	Read, write (create only)
`OwnerEmail`	Single-valued, string	Gets or sets the site collection owner email address.	Read, write
`Parent`	Single-valued, string, reference (`WebApplication` object)	Gets the parent of the object.	Read
`Port`	Single-valued, integer	Gets the port number used by the virtual server that hosts the site collection.	Read
`PortalName`	Single-valued, string	Gets or sets the portal name.	Read, write (update only)
`PortalUrl`	Single-valued, string	Gets or sets the portal URL.	Read, write (update only)
`PrimaryUri`	Single-valued,	Gets the portal URI.	Read

Attribute	Type	Description	Supported operations
	string		
`QuotaID`	Single-valued, integer	Gets of sets the quota ID.	Read, write (update only)
`ReadLocked`	Single-valued, Boolean	Gets or sets whether the site is unavailable for Read access.	Read, write (update only)
`ReadOnly`	Single-valued, Boolean	Gets or sets whether the site collection is read-only and unavailable for Write access.	Read, write (update only)
`ResourceQuotaExceeded`	Single-valued, Boolean	Gets whether the resource quota limit for the site collection has been exceeded since the last daily quota reset operation.	Read
`ResourceQuotaExceededNotificationSent`	Single-valued, Boolean	Gets whether a resource quota exceeded notification was sent since the last daily quota reset operation for the site collection.	Read
`ResourceQuotaWarningNotificationSent`	Single-valued, Boolean	Gets whether a resource quota exceeded warning was sent since the last daily quota reset operation for the site collection.	Read
`SchemaVersion`	Single-valued, string	Gets the site collection version number for upgrade compatibility checks.	Read
`SecondaryContact`	Single-valued, string, reference (`User` object)	Description is not available.	Read, write (update only)

Attribute	Type	Description	Supported operations
ServerRelativeUrl	Single-valued, string	Gets or sets the server-relative URL of the root website.	Read, write (update only)
ShareByEmailEnabled	Single-valued, Boolean	Gets or sets whether the users are allowed to grant access permissions to guests, so that they could access the site collection resources.	Read, write (update only)
ShareByLinkEnabled	Single-valued, Boolean	Gets or sets whether the users are allowed to share the site collection documents by providing hyperlinks to those documents.	Read, write (update only)
ShowURLStructure	Single-valued, Boolean	Gets or sets whether to show the site collection URL structure.	Read, write (update only)
SourceSiteId	Single-valued, string (GUID)	Gets the source site ID for an upgrade evaluation site collection.	Read
StorageMaximumLevel	Single-valued, LargeInteger	Gets or sets the maximum disk space limit used by the site.	Read, write (update only)
StorageWarningLevel	Single-valued, LargeInteger	Gets or sets the storage warning level, sent to administrators before reaching the maximum limit of the available site storage space.	Read, write (update only)
SyndicationEnabled	Single-valued, Boolean	Gets or sets whether RSS syndication is enabled for the site collection.	Read, write (update only)
SystemAccount	Single-valued,	Gets the system account of the site collection.	Read

Attribute	Type	Description	Supported operations
	string, reference (`User` object)		
`TrimAuditLog`	Single-valued, Boolean	Gets or sets whether to delete old data from the audit log.	Read, write (update only)
`UpgradeReminderDate`	Single-valued, DateTime	Specifies the date after which site administrators receive a reminder to upgrade the site.	Read
`Upgrading`	Single-valued, Boolean	Gets whether a site upgrade is currently in progress.	Read
`Url`	Single-valued, string	Gets or sets the full URL of the root website of the site collection. The URL contains the host name and port number. NOTE: This attribute is required to create a new site collection in SharePoint.	Read, write (create only)
`UserCodeEnabled`	Single-valued, Boolean	Gets whether the user code service is enabled for the site collection.	Read
`UserCodeMaximumLevel`	Single-valued, string	Gets or sets the maximum allowed resource usage for the site.	Read, write (update only)
`UserCodeWarningLevel`	Single-valued, string	Gets or sets the warning limit of the resource usage. When this limit is exceeded, a warning email will be sent to site administrators.	Read, write (update only)
`UserDefinedWorkflowsEnabled`	Single-valued, Boolean	Gets or sets whether user-defined workflows are enabled for the site	Read, write (update

Attribute	Type	Description	Supported operations
		collection.	only)
`UserIsSiteAdminInSystem`	Single-valued, Boolean	Gets whether the current user is a site collection administrator.	Read
`UserToken`	Single-valued, binary	Gets the user token associated with the site collection	Read
`WarningNotificationSent`	Single-valued, Boolean	Gets whether a warning notification has been sent.	Read
`WebTemplate`	Single-valued, string	Description is not available.	Read, write
`WriteLocked`	Single-valued, Boolean	Gets whether the site collection is unavailable for Write access.	Read
`Zone`	Single-valued, string	Gets the URL zone that was used when creating the site object.	Read

User object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `User` object with the following synchronization operations.

Table 67: User object attributes

Attribute	Type	Description	Supported operations
`Alias`	Single-valued, string	Gets the alias of the object.	Read
`AllowBrowseUserInfo`	Single-valued, Boolean	Gets or sets whether the user can view information about other users of the website.	Read, write (update only)

Attribute	Type	Description	Supported operations
`Email`	Single-valued, string	Gets or sets the email address of the user.	Read, write (update only)
`Groups`	Multivalued, string, reference (`Group` object)	Gets the groups in which the object is a member.	Read
`Id`	Single-valued, string	Gets the object ID.	Read
`IsApplicationPrincipal`	Single-valued, Boolean	Gets whether the user is an application principal.	Read
`IsDomainGroup`	Single-valued, Boolean	Gets whether the user is a domain group.	Read
`IsHiddenInUI`	Single-valued, Boolean	Gets whether the user is hidden in the user interface.	Read
`IsShareByEmailGuestUser`	Single-valued, Boolean	Gets or sets whether the user is shared by email guest user.	Read, write (update only)
`IsSiteAdmin`	Single-valued, Boolean	Gets or sets whether the user is a site collection administrator.	Read, write (update only)
`IsSiteAuditor`	Single-valued, Boolean	Gets whether the user is a site collection auditor.	Read
`IsUserSettingsSyncedWithProvider`	Single-valued, Boolean	Gets or sets whether user settings have been synchronized with External Settings Provider.	Read, write (update only)
`LoginName`	Single-valued, string	Gets or sets login name of the user.	Read, write (create only)

Attribute	Type	Description	Supported operations
`Name`	Single-valued, string	Gets or sets the display name of the user.	Read, write (update only)
`Notes`	Single-valued, string	Gets or sets notes for the user.	Read, write (update only)
`Parent`	Single-valued, string, reference (`Site` object)	Gets the parent of the object.	Read
`RawSid`	Single-valued, binary	Gets the system ID of the user.	Read
`RequireRequestToken`	Single-valued, Boolean	Gets or sets whether the user requires a request token.	Read, write (update only)
`Sid`	Single-valued, string	Gets the security identifier (SID) of the user's network account.	Read
`SystemUserKey`	Single-valued, string	Gets the user key specific to the configuration.	Read
`UserId`	Single-valued, string	Gets the identifier of the user and the issuer of that identifier.	Read
`UserToken`	Single-valued, binary	Gets the token that identifies the authentication process for the user.	Read
`Xml`	Single-valued, string	Gets information about the user in the XML format.	Read

Web object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `Web` object with the following synchronization operations.

Table 68: Web object attributes

Attribute	Type	Description	Supported operations
`AllowAnonymousAccess`	Single-valued, Boolean	Gets whether anonymous access is allowed for the website.	Read
`AllowAutomaticASPXPageIndexing`	Single-valued, Boolean	Gets or sets whether to index the `.aspx` page of the website for search operations.	Read, write (update only)
`AllowDesignerForCurrentUser`	Single-valued, Boolean	Gets whether the current user is allowed to use the designer for the website.	Read
`AllowMasterPageEditingForCurrentUser`	Single-valued, Boolean	Gets whether the current user is allowed to edit master pages.	Read
`AllowRevertFromTemplateForCurrentUser`	Single-valued, Boolean	Gets whether the current user is allowed to revert from the website template.	Read
`AllowRssFeeds`	Single-valued, Boolean	Gets whether the website allows RSS feeds.	Read
`AllowUnsafeUpdates`	Single-valued, Boolean	Gets whether database updates are allowed without security validation.	Read, write (update only)
`AllWebTemplatesAllowed`	Single-valued, Boolean	Gets whether all available web templates (returned	Read

Attribute	Type	Description	Supported operations
		by the `GetAvailableWebTemplates` method) are allowed.	
`AlternateCssUrl`	Single-valued, string	Gets or sets the URL pointing at an alternate CSS file for the website.	Read, write (update only)
`AlternateHeader`	Single-valued, string	Gets or sets the URL pointing at an alternate `.aspx` page that is used for rendering the top navigation area on the website.	Read, write (update only)
`AnonymousPermMask64`	Multivalued, string	Gets or sets base permissions for anonymous users of the website.	Read, write (update only)
`AnonymousState`	Single-valued, string	Gets or sets the level of access for anonymous users of the website.	Read, write (update only)
`AppDatabaseName`	Single-valued, string	Gets the name of the application database associated with the website.	Read
`AppDatabaseServerReferenceId`	Single-valued, string (`GUID`)	Gets the ID of the server on which the database is located.	Read
`AppDatabaseTargetApplicationId`	Single-valued, string	Gets the ID of the target application.	Read
`AppInstanceId`	Single-valued, string	Gets the ID of the App instance the website represents.	Read

Attribute	Type	Description	Supported operations
	(GUID)		
ASPXPageIndexed	Single-valued, Boolean	Gets whether the automatic indexing of the website .aspx pages is enabled.	Read
AssociatedMemberGroup	Single-valued, string, reference (Group object)	Gets or sets the users who can be contributors of the website.	Read, write
AssociatedOwnerGroup	Single-valued, string, reference (Group object)	Gets or sets the associated owner groups of the website.	Read, write (update only)
AssociatedVisitorGroup	Single-valued, string, reference (Group object)	Gets or sets the associated visitor group of the website.	Read, write
Author	Single-valued, string, reference (User object)	Gets or sets the user who created the website.	Read, write
CacheAllSchema	Single-valued, Boolean	Gets or sets whether caching of all schemas of the website is enabled.	Read, write (update only)
ClientTag	Single-valued, string (integer)	Gets or sets the client cache control number for the website.	Read, write (create only)
Configuration	Single-	Gets the ID of the	Read

Attribute	Type	Description	Supported operations
	valued, string (`integer`)	site definition configuration that was used to create the website or the template from which the website was created.	
`Created`	Single-valued, string (`DateTime`)	Gets or sets the date and time when the website was created.	Read, write (update only)
`CurrencyLocaleID`	Single-valued, string (`integer`)	Gets or sets the identifier of the currency that is used on the website.	Read, write (update only)
`CurrentChangeToken`	Single-valued, string (`SPChangeToken`)	Gets the token that is used for logging the next change to the website.	Read
`CurrentUser`	Single-valued, string, reference (`User` object)	Gets the current user of the website.	Read
`CustomJavaScriptFileUrl`	Single-valued, string	Gets or sets the URL pointing at the custom JavaScript file used by the `CustomJsUrl` web control.	Read, write (update only)
`CustomMasterUrl`	Single-valued, string	Gets or sets the URL pointing to a custom master page for the website.	Read, write (update only)

Attribute	Type	Description	Supported operations
CustomUploadPage	Single-valued, string	Gets or sets the path to a custom application page for uploading files.	Read, write (update only)
Description	Single-valued, string	Gets or sets the description for the website.	Read, write (update only)
DocumentLibraryCalloutOfficeWebAppPreviewersDisabled	Single-valued, Boolean	Gets whether the WAC previewers are disabled for the Document Library Callouts.	Read
EffectiveBasePermissions	Multivalued, string	Gets the effective base permissions assigned to the current user.	Read
EffectivePresenceEnabled	Single-valued, Boolean	Gets whether effective presence information is enabled for the website.	Read
EnableMinimalDownload	Single-valued, Boolean	Gets or sets whether Minimal Download Strategy is enabled for the website.	Read, write (update only)
ExcludeFromOfflineClient	Single-valued, Boolean	Gets or sets whether to download data from the website to the client during offline synchronization.	Read, write (update only)
ExecuteUrl	Single-valued, string	Gets the URL that is called after instantiating the site definition for business solutions.	Read

Attribute	Type	Description	Supported operations
`Exists`	Single-valued, Boolean	Gets a value that indicates whether the website exists.	Read
`FileDialogPostProcessorId`	Single-valued, string (`GUID`)	Gets or sets the ID for the user interface element used for web views in the file dialogs and forms of document libraries.	Read, write (update only)
`FirstUniqueAncestorWeb`	Single-valued, string, reference (`Web` object)	Gets the first unique website that has unique permissions.	Read
`FirstUniqueRoleDefinitionWeb`	Single-valued, string, reference (`Web` object)	Gets the website that defines role definitions for the current website.	Read
`HasUniqueRoleAssignments`	Single-valued, Boolean	Gets or sets whether the object has unique role assignments or inherits its assignments from a parent.	Read, write (create only)
`HasUniqueRoleDefinitions`	Single-valued, Boolean	Gets whether the object has unique role assignments, including those inherited from a parent object.	Read
`HideSiteContentsLink`	Single-valued, Boolean	Gets or sets whether a link to site contents is available in the site	Read, write (update only)

Attribute	Type	Description	Supported operations
		actions menu (the gear icon).	
`Id`	Single-valued, string	Gets the object ID.	Read
`IncludeSupportingFolders`	Single-valued, Boolean	Gets or sets whether supporting folders for files or folders in the website are included in enumeration operations for these files or folders.	Read, write (update only)
`IndexedPropertyKeys`	Multivalued, string	Gets the property keys for properties that need to be exposed through the Site Data Web Service.	Read
`IsADAccountCreationMode`	Single-valued, Boolean	Gets whether user accounts are created automatically in Active Directory when users are invited to the website.	Read
`IsADEmailEnabled`	Single-valued, Boolean	Gets whether email for AD DS is enabled on the website.	Read
`IsAppWeb`	Single-valued, Boolean	Gets whether the website is a container for an application.	Read
`IsMultilingual`	Single-valued, Boolean	Gets or sets whether the	Read, write (update

Attribute	Type	Description	Supported operations
		website has a multilingual user interface enabled.	only)
`IsRootWeb`	Single-valued, Boolean	Gets whether the website is the top-level site in the site collection.	Read
`Language`	Single-valued, reference (`Language` object)	Gets or sets the locale identifier of the default language for the website.	Read, write (create only)
`LastItemModifiedDate`	Single-valued, string (`DateTime`)	Gets or sets the date and time when the last modification was made to an item on the website.	Read, write (update only)
`Locale`	Single-valued, string (`CultureInfo`)	Gets the locale that is used to show time, currency, and numeric data on the website.	Read
`MasterPageReferenceEnabled`	Single-valued, Boolean	Gets whether master page referencing is enabled for the website.	Read
`MasterUrl`	Single-valued, string	Gets or sets the URL pointing at the master page for the website.	Read, write (update only)
`Name`	Single-valued, string	Gets or sets the name of the website.	Read, write (update only)
`NoCrawl`	Single-valued,	Gets or sets whether searching	Read, write

Attribute	Type	Description	Supported operations
	Boolean	is disabled for the website.	(update only)
`NonHostHeaderUrl`	Single-valued, string	Gets the full URL of the website.	Read
`OverwriteTranslationsOnChange`	Single-valued, Boolean	Gets or sets whether text changes made by user in the default language automatically overwrite existing translations in all other languages.	Read, write (update only)
`Parent`	Single-valued, string, reference (`Site` object)	Gets the parent of the object.	Read
`ParserEnabled`	Single-valued, Boolean	Gets or sets whether parsing is enabled for document libraries of the website.	Read, write (update only)
`PortalMember`	Single-valued, Boolean	Gets whether the website is associated with a portal site.	Read
`PortalName`	Single-valued, string	Gets the name of the portal site associated with the website.	Read
`PortalSubscriptionUrl`	Single-valued, string	Gets the URL that is used for alerts within the portal.	Read
`PortalUrl`	Single-valued,	Gets the URL that points to the portal	Read

Attribute	Type	Description	Supported operations
	string	site associated with the website.	
`PresenceEnabled`	Single-valued, Boolean	Gets or sets whether inline presence information is enabled for the website.	Read, write (update only)
`Provisioned`	Single-valued, Boolean	Gets or sets whether the website has been provisioned.	Read, write (update only)
`QuickLaunchEnabled`	Single-valued, Boolean	Gets or sets whether the Quick Launch area is enabled and available on the website.	Read, write (update only)
`RecycleBinEnabled`	Single-valued, Boolean	Gets whether the Recycle Bin is enabled for the website.	Read
`RequestAccessEmail`	Single-valued, string	Gets or sets the email address to which access requests are sent.	Read, write (update only)
`RequestAccessEnabled`	Single-valued, Boolean	Gets whether it is required to send a request in order to get access to the website.	Read
`RequireDynamicCanary`	Single-valued, Boolean	Gets whether the canary is required for all requests to the `UrlZone`.	Read
`SaveSiteAsTemplateEnabled`	Single-valued, Boolean	Gets or sets whether the website can be	Read, write (update

Attribute	Type	Description	Supported operations
		saved as a template.	only)
`ServerRelativeUrl`	Single-valued, string	Gets or sets the website URL in a server-relative format.	Read, write (update only)
`ShowUrlStructureForCurrentUser`	Single-valued, Boolean	Gets whether the current user is allowed to view the file structure of the website.	Read
`Site`	Single-valued, string, reference (`Site` object)	Gets the parent site collection for the website.	Read
`SiteClientTag`	Single-valued, string	Gets the client cache control number for the site collection.	Read
`SiteLogoDescription`	Single-valued, string	Gets or sets the description of the website logo.	Read, write (update only)
`SiteLogoUrl`	Single-valued, string	Gets or sets the absolute URL pointing at the website logo.	Read, write (update only)
`SupportedUICultures`	Multivalued, string (`CultureInfo`)	Gets information about the cultures supported by the website.	Read
`SyndicationEnabled`	Single-valued, Boolean	Gets or sets whether RSS is enabled for the website.	Read, write (update only)

Attribute	Type	Description	Supported operations
`ThemedCssFolderUrl`	Single-valued, string	Gets or sets the URL pointing to the folder that holds the CSS file that is used to display the website.	Read, write (update only)
`Title`	Single-valued, string	Gets or sets the website title.	Read, write (update only)
`TreeViewEnabled`	Single-valued, Boolean	Gets or sets whether Tree View is enabled in the website user interface.	Read, write (update only)
`UICulture`	Single-valued, string (`CultureInfo`)	Gets the default language for the website.	Read
`UIVersion`	Single-valued, string (`integer`)	Gets or sets the current version number of the user interface.	Read, write (update only)
`Url`	Single-valued, string	Gets or sets the absolute URL of the website.	Read, write (create only)
`UserIsSiteAdmin`	Single-valued, Boolean	Gets whether the user has administrator rights on the website.	Read
`UserIsWebAdmin`	Single-valued, Boolean	Gets whether the user is a member of the Administrator group for the website.	Read
`WebTemplate`	Single-	Gets the name of	Read

Attribute	Type	Description	Supported operations
	valued, string	the site definition or template that was used to create the website.	
`WebTemplateId`	Single-valued, string (`integer`)	Gets or sets the ID of the template or definition that was used to create the website.	Read, write (create only)

WebApplication object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `WebApplication` object with the following synchronization operations.

Table 69: WebApplication object attributes

Attribute	Type	Description	Supported operations
`AlertsEnabled`	Single-valued, Boolean	Gets or sets whether alerts are allowed in the web application.	Read, write (update only)
`AlertsLimited`	Single-valued, Boolean	Gets or sets whether a limit is imposed on the number of lists and list items for which alerts can be created.	Read, write (update only)
`AlertsMaximum`	Single-valued, integer	Gets or sets the maximum number of lists and list items for which a single user can create alerts.	Read, write (update only)
`AlertsMaximumQuerySet`	Single-valued,	Gets or sets the maximum number	Read, write

Attribute	Type	Description	Supported operations
	integer	of records in a query set that is associated with an alert object.	(update only)
`AllowAccessToWebPartCatalog`	Single-valued, Boolean	Gets or sets whether sites in the Web application can use Web Parts located in the global catalog.	Read, write (update only)
`AllowAnalyticsCookieForAnonymousUsers`	Single-valued, Boolean	Gets or sets whether analytics cookies are allowed for anonymous users.	Read, write (update only)
`AllowContributorsToEditScriptableParts`	Single-valued, Boolean	Gets or sets whether the contributors are allowed to edit scriptable Web Parts.	Read, write (update only)
`AllowDesigner`	Single-valued, Boolean	Gets or sets whether websites within the web application can be edited with SharePoint Designer.	Read, write (update only)
`AllowedInlineDownloadedMimeTypes`	Multivalued, string	Gets the MIME content types that are not force-downloaded to the computer of the user. Files not listed in this attribute value are considered to be script files and can interact with the	Read

Attribute	Type	Description	Supported operations
		web application on the user's behalf.	
AllowHighCharacterListFolderNames	Single-valued, Boolean	Gets or sets whether non-alphanumeric characters are allowed in the list folder names that are generated automatically.	Read, write (update only)
AllowMasterPageEditing	Single-valued, Boolean	Gets or sets whether the users are allowed to edit master pages.	Read, write (update only)
AllowOMCodeOverrideThrottleSettings	Single-valued, Boolean	Gets or sets whether custom object model code is allowed to override the throttle settings.	Read, write (update only)
AllowPartToPartCommunication	Single-valued, Boolean	Gets or sets whether the Web application allows communication between different Web Parts.	Read, write (update only)
AllowRevertFromTemplate	Single-valued, Boolean	Gets or sets whether customized sites can be rolled back to their base templates.	Read, write (update only)
AllowSelfServiceUpgradeEvaluation	Single-valued, Boolean	Gets or sets whether upgrade evaluation site collections can be created.	Read, write (update only)
AllowSilverlightPrompt	Single-valued, Boolean	Gets or sets whether UI	Read, write (update

Attribute	Type	Description	Supported operations
		elements that require Microsoft Silverlight prompt the user to download and install Silverlight.	only)
`AlwaysProcessDocuments`	Single-valued, Boolean	Gets or sets whether documents to be returned are always processed by document parsers.	Read, write (update only)
`ApplicationPrincipalMaxRights`	Multivalued, string	Gets or sets the maximum rights that any application principal user has in the web application.	Read, write (update only)
`AutomaticallyDeleteUnusedSiteCollections`	Single-valued, Boolean	Gets or sets whether to automatically delete unused site collections.	Read, write (update only)
`BlockedFileExtensions`	Multivalued, string	Gets the list of file name extensions that are forbidden for download from the sites within the web application.	Read
`BrowserCEIPEnabled`	Single-valued, Boolean	Gets or sets whether the Customer Experience Improvement Program is enabled in the web browser.	Read, write (update only)
`CanRenameOnRestore`	Single-valued, Boolean	Gets whether the web application can be renamed during its restore.	Read

Attribute	Type	Description	Supported operations
`CanSelectForBackup`	Single-valued, Boolean	Gets or sets whether the web application can be backed up.	Read, write (update only)
`CanSelectForRestore`	Single-valued, Boolean	Gets or sets whether the web application can be restored.	Read, write (update only)
`CascadeDeleteMaximumItemLimit`	Single-valued, integer	Gets or sets the maximum number of items that can be checked in a Cascade or Restrict delete operation.	Read, write (update only)
`CascadeDeleteTimeoutMultiplier`	Single-valued, integer	Gets or sets the cost per item deleted in a referential integrity delete operation.	Read, write (update only)
`CellStorageWebServiceEnabled`	Single-valued, Boolean	Gets or sets whether the Web service named `WebSvcCellStorage` is enabled.	Read, write (update only)
`ChangeLogExpirationEnabled`	Single-valued, Boolean	Gets or sets whether change logs get deleted after the retention period set in the `ChangeLogRetentionPeriod` property expires.	Read, write (update only)
`ChangeLogRetentionPeriod`	Single-valued, string (`TimeSpan`)	Gets or sets the period (in days) during which the change logs are retained.	Read, write (update only)
`CrossDomainPhotosEnabled`	Single-valued, Boolean	Gets or sets whether cross-domain photos are	Read, write (update

Attribute	Type	Description	Supported operations
		enabled.	only)
`CustomAppErrorLimit`	Single-valued, integer	Gets or sets the maximum number of calls that the Web application can make each 24 hours to log custom errors.	Read, write (update only)
`DailyStartUnthrottledPrivilegedOperationsHour`	Single-valued, integer	Gets or sets the hour (in the local time zone) when the unthrottled daily time window starts.	Read, write (update only)
`DailyStartUnthrottledPrivilegedOperationsMinute`	Single-valued, integer	Gets or sets the minute (in the local time zone) when the unthrottled daily time window starts.	Read, write (update only)
`DailyUnthrottledPrivilegedOperationsDuration`	Single-valued, integer	Gets or sets the period (in hours) during which the unthrottled daily time window remains open.	Read, write (update only)
`DaysToShowNewIndicator`	Single-valued, integer	Gets or sets the period (in days) during which the **New** icon is displayed next to new list items.	Read, write (update only)
`DefaultQuotaTemplate`	Single-valued, string	Gets or sets the default quota template applicable to all site collections.	Read, write (update only)
`DefaultServerComment`	Single-valued, string	Gets the default comment for the Internet	Read

Attribute	Type	Description	Supported operations
		Information Services (IIS) website. This default comment is used in situations where a comment is not specified by the web application.	
DefaultTimeZone	Single-valued, integer	Gets or sets the default time zone for the web application.	Read, write (update only)
DisableCoauthoring	Single-valued, Boolean	Gets or sets whether co-authoring using Microsoft Office is disabled.	Read, write (update only)
DisplayName	Single-valued, string	Gets the display name of the web application.	Read
DocumentLibraryCalloutOfficeWebAppPreviewersDisabled	Single-valued, Boolean	Gets or sets whether the Document Library Callout's WAC previewers are disabled.	Read, write (update only)
EmailToNoPermissionWorkflowParticipantsEnabled	Single-valued, Boolean	Gets or sets whether users that have no site permissions receive a notification email when they are assigned workflow tasks.	Read, write (update only)
EnabledClaimProviders	Multivalued, string	Reserved for internal use.	Read
EventHandlersEnabled	Single-	Gets or sets	Read,

Attribute	Type	Description	Supported operations
	valued, Boolean	whether event handlers are enabled for the Web application.	write (update only)
`EventLogRetentionPeriod`	Single-valued, string (`TimeSpan`)	Gets or sets the period (in days) during which the event logs are retained.	Read, write (update only)
`ExternalUrlZone`	Single-valued, string	Gets or sets the URL zone for cross-firewall access.	Read, write (update only)
`ExternalWorkflowParticipantsEnabled`	Single-valued, Boolean	Gets or sets whether external users can participate in a workflow if they have a document copy.	Read, write (update only)
`FileNotFoundPage`	Single-valued, string	Gets or sets the name of the HTML file that contains the error information to be displayed in a situation where a file is not found.	Read, write (update only)
`ForceseekEnabled`	Single-valued, Boolean	Gets or sets whether the FORCESEEK hint is enabled.	Read, write (update only)
`Id`	Single-valued, string	Gets or sets the object ID.	Read, write
`IncomingEmailServerAddress`	Single-valued, string	Gets or sets the name of the email server that is used to receive incoming email messages.	Read, write (update only)

Attribute	Type	Description	Supported operations
`InheritDataRetrievalSettings`	Single-valued, Boolean	Gets or sets whether the web application inherits data retrieval settings from the central administration application.	Read, write (update only)
`IsAdministrationWebApplication`	Single-valued, Boolean	Gets or sets whether the web application is the central administration application.	Read, write (update only)
`MasterPageReferenceEnabled`	Single-valued, Boolean	Gets or sets whether site administrators can enable dynamic master page referencing for the web application pages.	Read, write (update only)
`MaximumFileSize`	Single-valued, integer	Gets or sets the maximum file size limit for files to be uploaded.	Read, write (update only)
`MaxItemsPerThrottledOperation`	Single-valued, integer	Gets or sets the count of items at which throttling begins for list operations.	Read, write (update only)
`MaxItemsPerThrottledOperationOverride`	Single-valued, integer	Gets or sets the maximum count of items for which throttling is not enabled if the current user is an administrator or auditor.	Read, write (update only)

Attribute	Type	Description	Supported operations
MaxItemsPerThrottledOperationWarningLevel	Single-valued, integer	Gets or sets the warning level for the number of items in list operations.	Read, write (update only)
MaxQueryLookupFields	Single-valued, integer	Gets or sets the maximum number of lookup fields that may be included in a list item query.	Read, write (update only)
MaxSizeForSelfServiceEvalSiteCreationMB	Single-valued, LargeInteger	Gets or sets the maximum possible size (in MB) of a site collection for which the creation of evaluation sites is permitted through self-service.	Read, write (update only)
MaxUniquePermScopesPerList	Single-valued, integer	Gets or sets the maximum number unique scopes that can be in a list.	Read, write (update only)
MetaWeblogAuthenticationEnabled	Single-valued, Boolean	Gets or sets whether authentication via the MetaWeblog API is enabled for the web application.	Read, write (update only)
MetaWeblogEnabled	Single-valued, Boolean	Gets or sets whether the MetaWeblog API is enabled for the web application.	Read, write (update only)
OfficialFileName	Single-valued, string	Gets or sets the name of the Records Repository Web Service that is used to get the official file.	Read, write (update only)

Attribute	Type	Description	Supported operations
`OfficialFileUrl`	Multivalued, string	Gets the URL of the Recovery Repository Web Service that is used to get the official file.	Read
`OutboundMailCodePage`	Single-valued, integer	Gets or sets the default code page that is used for sending emails.	Read, write (update only)
`OutboundMailReplyToAddress`	Single-valued, string	Gets or sets the default reply email address to be used in email messages.	Read, write (update only)
`OutboundMailSenderAddress`	Single-valued, string	Gets or sets the default sender's email address to be displayed in the **From** field of outgoing email messages.	Read, write (update only)
`Parent`	Single-valued, string	Gets or sets the parent of the object.	Read, write
`PresenceEnabled`	Single-valued, Boolean	Gets or sets whether presence information is enabled in the web application.	Read, write (update only)
`ReadOnlyMaintenanceLink`	Single-valued, string	Gets or sets a link to the upgrade maintenance page.	Read, write (update only)
`RecycleBinCleanupEnabled`	Single-valued, Boolean	Gets or sets whether recycle bin cleanup is enabled.	Read, write (update only)
`RecycleBinEnabled`	Single-	Gets or sets	Read,

Attribute	Type	Description	Supported operations
	valued, Boolean	whether the Recycle Bin is enabled.	write (update only)
`RecycleBinRetentionPeriod`	Single-valued, integer	Gets or sets the period (in days) during which deleted items are retained in the Recycle Bin.	Read, write (update only)
`RenderingFromMetainfoEnabled`	Single-valued, Boolean	Gets or sets whether page roundtrip optimization is enabled.	Read, write (update only)
`RequireContactForSelfServiceSiteCreation`	Single-valued, Boolean	Gets or sets whether self-service site creation requires contact information of the site owner.	Read, write (update only)
`ScopeExternalConnectionsToSiteSubscriptions`	Single-valued, Boolean	No description available.	Read, write (update only)
`SecondStageRecycleBinQuota`	Single-valued, integer	Gets or sets the storage quota (in per cent) available to the second stage Recycle Bin.	Read, write (update only)
`SelfServiceCreateIndividualSite`	Single-valued, Boolean	Gets or sets whether self-service should create an individual site or a site collection.	Read, write (update only)
`SelfServiceCreationParentSiteUrl`	Single-valued, string	Gets or sets the parent site URL under which children sites are to be created.	Read, write (update only)

Attribute	Type	Description	Supported operations
`SelfServiceCreationQuotaTemplate`	Single-valued, string	Gets or sets the quota template to be used when creating site collections.	Read, write (update only)
`SelfServiceSiteCreationEnabled`	Single-valued, Boolean	Gets or sets whether sites can be created by using self-service in the Web application.	Read, write (update only)
`SelfServiceSiteCustomFormUrl`	Single-valued, string	Gets or sets the custom form URL to be used when creating sites through self-service.	Read, write (update only)
`SendLoginCredentialsByEmail`	Single-valued, Boolean	Gets or sets whether the login credentials of newly-created users are sent to them via email.	Read, write (update only)
`SendSiteUpgradeEmails`	Single-valued, Boolean	Gets or sets whether to send an email notification once a site upgrade completes.	Read, write (update only)
`SendUnusedSiteCollectionNotifications`	Single-valued, Boolean	Gets or sets whether to sent notifications to the owners of unused sites.	Read, write (update only)
`ShowStartASiteMenuItem`	Single-valued, Boolean	Gets or sets whether the **Start a new site** menu command is available.	Read, write (update only)
`ShowURLStructure`	Single-valued,	Gets or sets	Read,

Attribute	Type	Description	Supported operations
	Boolean	whether the users are allowed to see the file structure of the websites.	write (update only)
`StorageMetricsProcessingDuration`	Single-valued, integer	Gets or sets the maximum duration (in second) for the processing of metric changes for documents.	Read, write (update only)
`SuiteBarBrandingElementHtml`	Single-valued, string	Gets or sets the HTML snippet that is displayed in the `SuiteBarBrandingElement` control.	Read, write (update only)
`SyndicationEnabled`	Single-valued, Boolean	Gets or sets whether syndication is enabled.	Read, write (update only)
`TypeName`	Single-valued, string	Gets the type of object for the web application.	Read
`UnthrottledPrivilegedOperationWindowEnabled`	Single-valued, Boolean	Gets or sets whether to enable unthrottled daily time window. When this attribute is set to `TRUE`, large list operations are not throttled when they occur within the time window.	Read, write (update only)
`UnusedSiteNotificationPeriod`	Single-valued, string (`TimeSpan`)	Gets the time period during which the site was unused.	Read
`UnusedSiteNotificationsBeforeDeletion`	Single-valued,	Gets or sets the	Read,

Attribute	Type	Description	Supported operations
	integer	number of site deletion notifications that must be sent before an unused site gets deleted.	write (update only)
`UpgradeEvalSitesRetentionDays`	Single-valued, integer	Gets or sets the period (in days) since the evaluation site creation date after which the evaluation site gets deleted.	Read, write (update only)
`UpgradeMaintenanceLink`	Single-valued, string	Gets or sets a link pointing to the upgrade maintenance page.	Read, write (update only)
`UpgradeReminderDelay`	Single-valued, integer	Gets or sets the number of days by which the site collection administrator can put off the upgrade reminder. When this attribute value is set to 0, the status notification shows that an upgrade is required.	Read, write (update only)
`UseClaimsAuthentication`	Single-valued, Boolean	Gets or sets whether claims authentication is enabled.	Read, write (update only)
`UseExternalUrlZoneForAlerts`	Single-valued, Boolean	Gets or sets whether to use an external URL zone in emails providing information about alerts.	Read, write (update only)

Attribute	Type	Description	Supported operations
		If this attribute is set to `TRUE` and a cross-firewall URL zone is configured, then that zone is used in the emails containing alerts. If this attribute is set to `TRUE`, and no cross-firewall URL zone is configured, then the emails containing alerts use the default zone URL for the web application.	
`UserDefinedWorkflowMaximumComplexity`	Single-valued, integer	Gets or sets the maximum number of activities and bindings that a user-defined workflow can have.	Read, write (update only)
`UserDefinedWorkflowsEnabled`	Single-valued, Boolean	Gets or sets whether the users can create workflows in the web application.	Read, write (update only)
`UserPhotoErrorExpiration`	Single-valued, string (`Double`)	Gets or sets the period (in hours) upon which the error window for photos expires.	Read, write (update only)
`UserPhotoExpiration`	Single-valued, string (`Double`)	Gets or sets the period (in hours) upon which the photo expires.	Read, write (update only)
`UserPhotoImportEnabled`	Single-valued, Boolean	Gets or sets whether photo import is enabled.	Read, write (update

Attribute	Type	Description	Supported operations
			only)
`UserPhotoOnlineImportEnabled`	Single-valued, Boolean	Gets or sets whether photo import is enabled for Exchange Online.	Read, write (update only)
`WebFileExtensions`	Multivalued, string	Gets the list of file name extensions that identify web files.	Read

WebTemplate object attributes

In a SharePoint connection, the Synchronization Service supports the following attributes of the `WebTemplate` object with the following synchronization operations.

Table 70: WebTemplate object attributes

Attribute	Type	Description	Supported operations
`AllowGlobalFeatureAssociations`	Single-valued, Boolean	Gets whether global feature associations are allowed on sites created with the web template.	Read
`CompatibilityLevel`	Single-valued, integer	Gets the web template compatibility level.	Read
`Description`	Single-valued, string	Gets the web template description.	Read
`DisplayCategory`	Single-valued, string	Gets the name of the category to which the web template belongs.	Read
`Id`	Single-valued, string	Gets or sets the object ID.	Read, write (create only)
`IDWebTemplate`	Single-	Gets the web template	Read

Attribute	Type	Description	Supported operations
	valued, integer	ID.	
`IsCustomTemplate`	Single-valued, Boolean	Gets whether this is a custom web template.	Read
`IsFarmWideTemplate`	Single-valued, Boolean	Gets whether the web template is a farm-wide template and can be used to create sites across the entire SharePoint farm.	Read
`IsHidden`	Single-valued, Boolean	Gets whether the web template is hidden from the user interface.	Read
`IsRootWebOnly`	Single-valued, Boolean	Gets whether the web template can only be applied to the root site in the site collection.	Read
`IsSubWebOnly`	Single-valued, Boolean	Gets whether the web template is only applicable to subsites within the site collection.	Read
`IsUnique`	Single-valued, Boolean	Gets whether the site created from the web template inherits from its parent.	Read
`Lcid`	Single-valued, integer	Gets the locale identifier of the web template.	Read
`Name`	Single-valued, string	Gets the internal name of the web template.	Read
`Parent`	Single-valued, string, reference (`Web` object)	Gets or sets the parent of the object.	Read, write (create only)

Attribute	Type	Description	Supported operations
`ProvisionAssembly`	Single-valued, string	Gets the name of the assembly that implements the class which contains logic for provisioning sites created through the web template.	Read
`ProvisionClass`	Single-valued, string	Gets the name of the class which provides logic for provisioning sites created through the web template.	Read
`ProvisionData`	Single-valued, string	Gets the data that is passed to the site provisioning handler when creating sites.	Read
`SupportsMultilingualUI`	Single-valued, Boolean	Gets whether it is possible to enable alternate user interface languages for the sites created from the web template.	Read
`Title`	Single-valued, string	Gets the display name of the web template.	Read
`UserLicensingId`	Single-valued, string	Gets the per-user license.	Read
`VisibilityFeatureDependencyId`	Single-valued, string	Gets the GUID of the feature on which the web template depends.	Read

Considerations for creating objects in SharePoint

When creating objects in SharePoint, consider the following:

- **RoleAssignment object**: To create this object, you must populate the value of the **Member** attribute for the object. Since **Member** is a reference attribute, you can only populate its value by configuring a value generation rule. For more information about value generation rules, see Using value generation rules.

- **Site object**: To create this object, you must populate the values of attributes **URL** and **Owner** for the object.

Working with Microsoft 365

To create a connection to Microsoft 365, you must use Synchronization Service in conjunction with a special connector called Microsoft 365 Connector (formerly known as **Office 365 Connector**). This connector is included in the Synchronization Service package.

The Microsoft 365 Connector supports the following features:

Table 71: Microsoft 365 Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No
Secure Sockets Layer (SSL) data encryption Specifies whether the connector can use SSL to encrypt data transmitted between Active Roles Synchronization Service and the connected data system.	Yes

Creating a Microsoft 365 connection

With the **Microsoft 365 connector**, you can configure data synchronization connections for the Microsoft 365 service.

You can create an M365 connector by configuring an Azure application in the Synchronization Service Console:

- To create and configure an M365 connector with manual configuration, see Creating a Microsoft 365 connector with manual configuration.
- To create and configure an M365 connector with automatic configuration, see Creating a Microsoft 365 connector with automatic configuration.

Creating a Microsoft 365 connector with manual configuration

With the **Microsoft 365 connector**, you can configure data synchronization connections for the Microsoft 365 service.

You can create an M365 connector by configuring an Azure application manually in the Synchronization Service Console. One Identity recommends using **Manual configuration** if you want to use an existing Azure application for the connection.

IMPORTANT: If you are upgrading from an older version of Active Roles Synchronization Service to Active Roles Synchronization Service 8.1.3 or later, and the connector was configured manually, then you must update the authentication data to be able to run a synchronization workflow.

To update the authentication data, you can:

- Use **Auto configuration**. One Identity recommends this approach, as the process is handled automatically by the Active Roles Synchronization Service.

  For more information on automatic configuration, see Modifying the automatic configuration settings of a Microsoft 365 connector.
- Enter the **Certificate thumbprint** of the Azure tenant manually, and select the **Tenant Environment Type**.

To create a new M365 connector with manual configuration

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Microsoft 365 Connector**.
3. Click **Next**.
4. To use an existing Azure application, select **Manual configuration**.

   NOTE: Alternatively, to use and update an existing Azure application, you can also select **Auto configuration**. Under **Auto configuration**, click **Log in to Azure**, then select the **Tenant environment type** of the Azure tenant. After logging in to Azure with your tenant, the **Tenant ID**, **Application ID**, **Certificate thumbprint** and **Tenant environment type** parameters will be automatically filled in.
5. Enter the **Tenant ID**, **Application ID** and **Certificate thumbprint** of the Azure tenant as they appear on the Azure portal. Then, select the **Tenant Environment Type** of the Azure tenant.
6. To test the connection with the new parameters, click **Test connection**.
7. To finish creating a connection to Microsoft 365, click **Finish**.

Creating a Microsoft 365 connector with automatic configuration

With the **Microsoft 365 connector**, you can configure data synchronization connections for the Microsoft 365 service.

You can create an M365 connector by configuring an Azure application automatically in the Synchronization Service Console. One Identity recommends using **Auto configuration** if you want to create a new Azure application for the connection.

To create a new M365 connector with automatic configuration

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Microsoft 365 Connector**.
3. Click **Next**.
4. To create a new Azure application or update an existing one, select **Auto configuration**.

   NOTE: If you have more than one Azure Active Directory (Azure AD) service in your Azure tenant, select **I have more than one Azure AD in my Azure tenant**, and use the **Tenant ID** field to specify the GUID of the Azure AD for which you want to set up synchronization. For more information, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.
5. Select one of the following options based on the number of Azure AD services in your Azure tenant:
   - **I have one Azure AD in my Azure tenant.**
   - **I have more than one Azure AD in my Azure tenant.**
6. Authenticate your access to Azure AD:
   - If you have selected **I have one Azure AD in my Azure tenant**, to authenticate your access to Azure AD, click **Log in to Azure**, and from the **Select Environment Type** drop-down, select the environment type of your Azure tenant.

     NOTE: Active Roles Synchronization Service supports Azure Cloud, Azure GCC and Azure GCC-H government tenants.
   - If you have selected **I have more than one Azure AD in my Azure tenant**, in **Tenant ID**, enter the GUID of the Azure AD for which you want to set up synchronization.

     TIP: For more information on how to find the GUID of an Azure AD service, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.

After specifying the tenant ID, to authenticate your access to Azure AD, click **Log in to Azure**, and in the **Select Environment Type** drop-down, select the environment of your Azure tenant.

> NOTE: If you select **I have more than one Azure AD in my Azure tenant**, the **Log in to Azure** button will be enabled only if you specify a well-formed Azure AD GUID in the **Tenant ID** text box.

You can now create or update the Azure application in Azure AD.

7. **Azure application name**: Enter the name of the new or existing Azure application.
8. To create or update the Azure application in Azure AD, click **Create or update Azure application**.

   The created or updated Azure application has the following directory roles assigned to it:

   - Directory Writers
   - Exchange Administrator
   - User Administrator

   The following permissions are also added, for which you must give admin consent:

   - Sign in and read user profile
   - Manage Exchange As Application

   > NOTE: You may need to set additional permissions depending on your needs.

9. To give admin consent for the permissions of the Azure application, click **Consent**. Then, in the **Azure Tenant Consent** dialog, click **Accept**.
10. To test the connection with the new parameters, click **Test connection**.
11. To finish creating a connection to Microsoft 365, click **Finish**.

Modifying a Microsoft 365 connection

With the **Microsoft 365 connector**, you can configure data synchronization connections for the Microsoft 365 service.

You can modify the settings of an existing M365 connector in the Synchronization Service Console:

- To modify the manually configured settings of an M365 connector, see Modifying the manual configuration settings of a Microsoft 365 connector.
- To modify the automatically configured settings of an M365 connector, see Modifying the automatic configuration settings of a Microsoft 365 connector.

Modifying the manual configuration settings of a Microsoft 365 connector

You can modify the manual configuration settings of an existing M365 connector in the Synchronization Service Console.

IMPORTANT: If you are upgrading from an older version of Active Roles Synchronization Service to Active Roles Synchronization Service 8.1.3 or later, and the connector was configured manually, then you must update the authentication data to be able to run a synchronization workflow.

To update the authentication data, you can:

- Use **Auto configuration**. One Identity recommends this approach, as the process is handled automatically by the Active Roles Synchronization Service.

  For more information on automatic configuration, see Modifying the automatic configuration settings of a Microsoft 365 connector.

- Enter the **Certificate thumbprint** of the Azure tenant manually, and select the **Tenant Environment Type**.

To modify the manual configuration settings of an M365 connector

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** under the existing Microsoft 365 connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
4. To use an existing Azure application, select **Manual configuration**.

   NOTE: Alternatively, to use and update an existing Azure application, you can also select **Auto configuration**. Under **Auto configuration**, click **Log in to Azure**, then select the **Tenant environment type** of the Azure tenant. After logging in to Azure with your tenant, the **Tenant ID**, **Application ID**, **Certificate thumbprint** and **Tenant environment type** parameters will be automatically filled in.

5. Enter the **Tenant ID**, **Application ID** and **Certificate thumbprint** of the Azure tenant as they appear on the Azure portal. Then, select the **Tenant Environment Type** of the Azure tenant.
6. To test the connection with the new parameters, click **Test connection**.
7. To modify the connection settings, click **Save**.

Modifying the automatic configuration settings of a Microsoft 365 connector

You can modify the automatic configuration settings of an existing M365 connector in the Synchronization Service Console.

To modify the auto configuration settings of an M365 connector

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** under the existing Microsoft 365 connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
4. To create a new Azure application or update an existing one, select **Auto configuration**.

   NOTE: If you have more than one Azure Active Directory (Azure AD) service in your Azure tenant, select **I have more than one Azure AD in my Azure tenant**, and use the **Tenant ID** field to specify the GUID of the Azure AD for which you want to set up synchronization. For more information, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.

5. Select one of the following options based on the number of Azure AD services in your Azure tenant:
   - **I have one Azure AD in my Azure tenant.**
   - **I have more than one Azure AD in my Azure tenant.**
6. Authenticate your access to Azure AD:
   - If you have selected **I have one Azure AD in my Azure tenant**, to authenticate your access to Azure AD, click **Log in to Azure**, and from the **Select Environment Type** drop-down, select the environment type of your Azure tenant.

     NOTE: Active Roles Synchronization Service supports Azure Cloud, Azure GCC and Azure GCC-H government tenants.

   - If you have selected **I have more than one Azure AD in my Azure tenant**, in **Tenant ID**, enter the GUID of the Azure AD for which you want to set up synchronization.

     TIP: For more information on how to find the GUID of an Azure AD service, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.

     After specifying the tenant ID, to authenticate your access to Azure AD, click **Log in to Azure**, and in the **Select Environment Type** drop-down, select the environment of your Azure tenant.

     NOTE: If you select **I have more than one Azure AD in my Azure tenant**, the **Log in to Azure** button will be enabled only if you specify a well-formed Azure AD GUID in the **Tenant ID** text box.

7. **Azure application name**: Enter the name of the new or existing Azure application.
8. To create or update the Azure application in Azure AD, click **Create or update Azure application**.

   The created or updated Azure application has the following directory roles assigned to it:

- Directory Writers
- Exchange Administrator
- User Administrator

The following permissions are also added, for which you must give admin consent:

- Sign in and read user profile
- Manage Exchange As Application

NOTE: You may need to set additional permissions depending on your needs.

9. To give admin consent for the permissions of the Azure application, click **Consent**. Then, in the **Azure Tenant Consent** dialog, click **Accept**.
10. To test the connection with the new parameters, click **Test connection**.
11. To modify the connection settings, click **Save**.

Microsoft 365 data supported out of the box

The next table lists the Microsoft 365 object types supported by the Microsoft 365 Connector out of the box and provides information about the operations you can perform on these objects by using the Microsoft 365 Connector.

Table 72: Supported objects and operations

Object	Read	Create	Delete	Update
ClientPolicy Allows you to work with client policies in Skype for Business Online. You can use client policies to determine the features of Skype for Business Online that are available to users. For more information on what data you can read and write, see ClientPolicy object attributes.	Yes	No	No	No
ConferencingPolicy Allows you to work with conferencing policies in Skype for Business Online. You can use conferencing policies to determine the features available to the users participating in a conference. For more information on what data you can read and write, see ConferencingPolicy object attributes.	Yes	No	No	No
Contact Allows you to work with external contact	Yes	Yes	Yes	Yes

Object	Read	Create	Delete	Update
properties in Microsoft 365. For more information on what data you can read and write, see Contact object attributes.				
DistributionGroup Allows you to work with distribution group properties in Microsoft 365. For more information on what data you can read and write, see DistributionGroup object attributes.	Yes	Yes	Yes	Yes
Domain Allows you to retrieve information about domains in Microsoft 365. For more information on what data you can retrieve, see Domain object attributes.	Yes	No	No	No
DynamicDistributionGroup Allows you to work with dynamic distribution group properties in Microsoft 365. For more information on what data you can read and write, see DynamicDistributionGroup object attributes.	Yes	Yes	Yes	Yes
ExternalAccessPolicy Allows you to work with external access policies in Skype for Business Online. For more information on what data you can read and write, see ExternalAccessPolicy object attributes.	Yes	No	No	No
HostedVoicemailPolicy Allows you to work with voice mail policies in Skype for Business Online. For more information on what data you can read and write, see HostedVoicemailPolicy object attributes.	Yes	No	No	No
LicensePlanService Allows you to retrieve information related to the license plans and services that are currently in use in Microsoft 365. For more information on what data you can	Yes	No	No	No

Object	Read	Create	Delete	Update
read and write, see LicensePlanService object attributes.				
Mailbox Allows you to work with Exchange Online mailboxes in Microsoft 365. For more information on what data you can read and write, see Mailbox object attributes.	Yes	Yes	Yes	Yes
MailUser Allows you to work with mail user properties in Microsoft 365. For more information on what data you can read and write, see MailUser object attributes.	Yes	Yes	Yes	Yes
PresencePolicy Allows you to work with presence policies in Skype for Business Online. For more information on what data you can read and write, see PresencePolicy object attributes.	Yes	No	No	No
SecurityGroup Allows you to work with security group properties in Microsoft 365. For more information on what data you can read and write, see SecurityGroup objects attributes.	Yes	Yes	Yes	Yes
SPOSite Allows you to work with the properties of site collections in SharePoint Online. For more information on what data you can read and write, see SPOSite object attributes.	Yes	Yes	Yes	Yes
SPOSiteGroup Allows you to work with groups inside site collections in SharePoint Online. For more information on what data you can read and write, see SPOSiteGroup object attributes.	Yes	Yes	Yes	Yes
SPOWebTemplate	Yes	No	No	No

Object	Read	Create	Delete	Update
Allows you to work with web templates in SharePoint Online. For more information on what data you can read and write, see SPOWebTemplate object attributes.				
SPOTenant Allows you to work with SharePoint Online organization. For more information on what data you can read and write, see SPOTenant object attributes.	Yes	No	No	Yes
User Allows you to read and write user properties in Microsoft 365. For more information on what data you can read and write, see User object attributes.	Yes	Yes	Yes	Yes
VoicePolicy Allows you to read or write data related to voice policies in Skype for Business Online. For more information on what data you can read and write, see VoicePolicy object attributes.	Yes	No	No	No
Microsoft 365 Group Allows you to read or write data related to Microsoft 365 group. For more information on what data you can read and write, see Microsoft 365 group attributes.	Yes	Yes	Yes	Yes

ClientPolicy object attributes

Table 73: ClientPolicy object attributes

Attribute	Description	Supported operations
`Anchor`	Gets the Anchor property value of the policy.	Read
`Description`	Gets the policy description.	Read

Attribute	Description	Supported operations
`Identity`	Gets the unique identifier assigned to the policy.	Read
`Members`	Gets the users who have been assigned the policy.	Read
`ObjectID`	Gets the unique object identifier (GUID).	Read

ConferencingPolicy object attributes

Table 74: ConferencingPolicy object attributes

Attribute	Description	Supported operations
`Anchor`	Gets the Anchor property value of the policy.	Read
`Description`	Gets the policy description.	Read
`Identity`	Gets the unique identifier assigned to the policy.	Read
`Members`	Gets the users who have been assigned the policy.	Read
`ObjectID`	Gets the unique object identifier (GUID).	Read

Contact object attributes

Table 75: Contact object attributes

Attribute	Description	Supported operations
`AcceptMessagesOnlyFrom`	Gets or sets the senders that can send email messages to the contact. This reference attribute can take senders in any of the following formats: • Alias • Canonical name • Display name • DN • Exchange DN • GUID	Read, Write

Attribute	Description	Supported operations
	• Name • Primary SMTP email address This reference attribute accepts the following object types: • `MailUser` • `Mailbox` • `Contact`	
`AcceptMessagesOnlyFromDLMembers`	Gets or sets the distribution groups whose members are allowed to send email messages to the contact. This reference attribute can take distribution groups in any of the following formats: • Canonical name • Display name • DN • GUID • Legacy Exchange DN • Name • Primary SMTP email address This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	Read, Write
`AcceptMessagesOnlyFromSendersOrMembers`	Gets or sets the senders who can send email messages to the contact. This reference attribute can take senders in any of the following formats: • Canonical name • Display name • Distinguished name (DN) • GUID	Read, Write

Attribute	Description	Supported operations
	• Legacy Exchange DN • Name • Primary SMTP email address This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	
`Alias`	Gets or sets the alias of the mail-enabled contact.	Read, Write
`AllowUMCallsFromNonUsers`	Gets or sets whether to exclude or include the contact in directory searches. This attribute can take one of the following values: • **`None`**: Specifies to exclude the contact from directory searches. • **`SearchEnabled`**: Specifies to include the contact in directory searches.	Read, Write
`AssistantName`	Gets or sets the name of the contact's assistant.	Read, Write
`BypassModerationFromSendersOrMembers`	Gets or sets the senders whose messages bypass moderation for the contact. This reference attribute can take any of the following values for the senders: • Canonical name • Display name • Distinguished name (DN) • GUID	Read, Write

Attribute	Description	Supported operations
	• Name • Legacy Exchange DN • Primary SMTP email address • Moderation does not apply to the senders designated as moderators for the contact. • This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	
`City`	Gets or sets the city of the contact.	Read, Write
`Company`	Gets or sets the company of the contact.	Read, Write
`CountryOrRegion`	Gets or sets the country or region of the contact.	Read, Write
`CreateDTMFMap`	Gets or sets whether to create a dual-tone multi-frequency (DTMF) map for the contact. This attribute can take one of the following values: • **`TRUE`**: Specifies to create a DTMF map for the contact. • **`FALSE`**: Specifies not to create a DTMF map for the contact.	Read, Write

Attribute	Description	Supported operations
CustomAttribute1	Get or set the additional custom values you specify.	Read, Write
CustomAttribute2		
CustomAttribute3		
CustomAttribute4		
CustomAttribute5		
CustomAttribute6		
CustomAttribute7		
CustomAttribute8		
CustomAttribute9		
CustomAttribute10		
CustomAttribute11		
CustomAttribute12		
CustomAttribute13		
CustomAttribute14		
CustomAttribute15		
Department	Gets or sets the department of the contact.	Read, Write
DisplayName	Gets or sets the name displayed in Microsoft 365 for the mail-enabled contact.	Read, Write
EmailAddresses	Gets or sets the email alias of the contact.	Read, Write
ExtensionCustomAttribute1	Get or set the additional custom values you specify. These attributes are multivalued. To specify multiple values, use a comma as a separator.	Read, Write
ExtensionCustomAttribute2		
ExtensionCustomAttribute3		
ExtensionCustomAttribute4		
ExtensionCustomAttribute5		
ExternalDirectoryObjectId	Gets the GUID of the contact.	Read
ExternalEmailAddress	Gets or sets the contact's e-mail	Read,

Attribute	Description	Supported operations
	address.	Write
`Fax`	Gets or sets the fax number of the contact.	Read, Write
`FirstName`	Gets or sets the first name of the mail-enabled contact.	Read, Write
`GrantSendOnBehalfTo`	Gets or sets the distinguished name (DN) of other senders that can send messages on behalf of the contact. This reference attribute only accepts the following object type: • `Mailbox`	Read, Write
`HiddenFromAddressListsEnabled`	Gets or sets whether or not Microsoft 365 hides the contact from the address lists. This attribute can take one of the following values: • **`TRUE`**: Specifies to hide the contact from the address lists. • **`FALSE`** (default): Specifies to display the contact in the address lists.	Read, Write
`HomePhone`	Gets or sets the home phone number of the contact.	Read, Write
`Initials`	Gets or sets the initials of the mail-enabled contact.	Read, Write
`LastName`	Gets or sets the last name of the mail-enabled contact.	Read, Write
`MacAttachmentFormat`	Gets or sets the Apple Macintosh operating system attachment format for messages sent to the contact. This attribute can take the following values: • `BinHex` • `UuEncode`	Read, Write

Attribute	Description	Supported operations
	• AppleSingle • AppleDouble	
MailTip	Gets or sets the message displayed to senders when they start writing an email message to the contact.	Read, Write
MailTipTranslations	Gets or sets the MailTip message translations in additional languages. This attribute accepts the following format: <LanguageLocale>:<MailTipMessageTranslation> A MailTip message translation cannot exceed 250 characters.	Read, Write
Manager	Gets or sets the manager of the contact.	Read, Write
MaxRecipientPerMessage	Gets or sets the maximum number of recipients to which the contact can address a message.	Read, Write
MessageBodyFormat	Gets or sets the message body format for messages sent to the contact. The values this attribute can write depend on the value in the MessageFormat attribute. When the value in the MessageFormat is Mime, the MessageBodyFormat attribute can write the following values: • Text • Html • TextAndHtml When the value in the MessageFormat is Text, the MessageBodyFormat attribute can only write the Text value.	Read, Write
MessageFormat	Gets or sets the message format for messages sent to the contact.	Read, Write

Attribute	Description	Supported operations
	This attribute can take the following values: • `Text` • `Mime`	
`MobilePhone`	Gets or sets the mobile phone number of the contact.	Read, Write
`ModeratedBy`	Gets or sets the moderators who are moderating the messages sent to the contact. To specify multiple moderators, use a comma as a separator. This reference attribute is required if you set the value of the `ModerationEnabled` attribute to **`TRUE`**. This reference attribute accepts the following object types: • `Mailbox` • `MailUser`	Read, Write
`ModerationEnabled`	Gets or sets whether moderation is enabled for the contact. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`Name`	Gets or sets the name of the mail-enabled contact.	Read, Write
`Notes`	Gets or sets notes about the contact.	Read, Write
`ObjectID`	Gets the unique object identifier (GUID).	Read
`Office`	Gets or sets the office of the contact.	Read, Write
`OtherFax`	Gets or sets the alternate fax number of the contact.	Read, Write

Attribute	Description	Supported operations
`OtherHomePhone`	Gets or sets the alternate home phone number of the contact.	Read, Write
`Pager`	Gets or sets the pager of the contact.	Read, Write
`Phone`	Gets or sets the work phone number of the contact.	Read, Write
`PhoneticDisplayName`	Gets or sets a phonetic pronunciation of the value specified in the `DisplayName` attribute for the contact.	Read, Write
`PostalCode`	Gets or sets the postal code of the contact.	Read, Write
`PostOfficeBox`	Gets or sets the post office box number of the contact.	Read, Write
`RejectMessagesFrom`	Gets or sets the senders whose messages to the contact are rejected. This attribute can take senders in one of the following formats: • Alias • Canonical name • Display name • Distinguished name (DN) • GUID • Name • Legacy Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `Contact` • `Mailbox`	Read, Write
`RejectMessagesFromDLMembers`	Gets or sets the distribution groups whose members cannot send email messages to the contact (their messages are rejected).	Read, Write

Attribute	Description	Supported operations
	This reference attribute can take distribution groups in one of the following formats: • Alias • Canonical name • Display name • Distinguished name (DN) • GUID • Legacy Exchange DN • Name • Primary SMTP email address This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	
`RejectMessagesFromSendersOrMembers`	Gets or sets the senders that cannot send email messages to the contact (their messages are rejected). This reference attribute can take any of the following values for the senders: • Alias • Canonical name • Display name • Distinguished name (DN) • GUID • Name • Legacy Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `Contact` • `DistributionGroup`	Read, Write

Attribute	Description	Supported operations
	• `DynamicDistributionGroup` • `Mailbox`	
`RequireSenderAuthenticationEnabled`	Gets or sets whether the senders that send messages to this contact must be authenticated. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`SecondaryAddress`	Gets or sets the secondary address for the contact if it has Unified Messaging enabled.	Read, Write
`SecondaryDialPlan`	Gets or sets a secondary Unified Messaging dial plan for the contact.	Read, Write
`SendModerationNotifications`	Gets or sets whether to send status notifications to users when a message they sent to the moderated distribution group is rejected by a moderator. This attribute can take one of the following values: • **`Always`**: Specifies that notifications are sent to all senders. • **`Internal`**: Specifies that notifications are only sent to the senders internal to your organization. • **`Never`**: Specifies that all status notifications are disabled.	Read, Write
`SimpleDisplayName`	Gets or sets an alternate description of the contact in a situation where a limited set of characters is allowed. The limited set of characters includes ASCII characters from 26 to 126.	Read, Write
`StateOrProvince`	Gets or sets the state or province of the	Read,

Attribute	Description	Supported operations
	contact.	Write
`StreetAddress`	Gets or sets the street address of the contact.	Read, Write
`TelephoneAssistant`	Gets or sets the phone number of the contact's assistant.	Read, Write
`Title`	Gets or sets the title of the contact.	Read, Write
`UMCallingLineIds`	Gets or sets telephone numbers or telephone extensions that can be mapped to the contact if it has Unified Messaging enabled. To specify multiple telephone numbers use a comma as a separator. This attribute only accepts values that have less than 128 characters.	Read, Write
`UMDtmfMap`	Gets or sets whether to create a user-defined DTMF map for the contact if it has Unified Messaging enabled.	Read, Write
`UseMapiRichTextFormat`	Gets or sets a format for the MAPI Rich Text Format messages sent to the contact. This attribute can take one of the following values: • **`Never`**: Specifies to convert all messages sent to the contact to the plain text format. • **`Always`**: Specifies to always use the MAPI Rich Text Format (RTF) for the messages sent to the contact. • **`UseDefaultSettings`**: Specifies to use the message format set in the MAPI client that sent the message to the contact.	Read, Write
`UsePreferMessageFormat`	Gets or sets whether the message format specified for the contact	Read, Write

Attribute	Description	Supported operations
	overrides any global settings (such as those configured for the remote domain). This attribute can take one of the following values: • **TRUE**: Specifies that the message format set for the mail user overrides any global settings. • **FALSE**: Specifies that global settings have precedence over the mail format set for the mail user.	
`WebPage`	Gets or sets the web page address of the contact.	Read, Write
`WindowsEmailAddress`	Gets or sets the email address of the contact stored in Active Directory.	Read, Write

DistributionGroup object attributes

Table 76: DistributionGroup object attributes

Attribute	Description	Supported operations
`AcceptMessagesOnlyFrom`	Gets or sets the senders that can send email messages to the distribution group. This reference attribute can take senders in any of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID	Read, Write

Attribute	Description	Supported operations
	• Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `MailUser` • `Mailbox` • `Contact`	
`AcceptMessagesOnlyFromDLMembers`	Gets or sets the distribution groups whose members are allowed to send email messages to the distribution group. This reference attribute can take distribution groups in any of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	Read, Write
`AcceptMessagesOnlyFromSendersOrMembers`	Gets or sets the senders who can send email messages to the distribution	Read, Write

Attribute	Description	Supported operations
	group. This attribute can take senders in any of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	
`Alias`	Gets or sets the alias of the distribution group.	Read, Write
`BypassModerationFromSendersOrMembers`	Gets or sets the senders whose messages bypass moderation for the distribution group. This reference attribute can take senders in any of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN)	Read, Write

Attribute	Description	Supported operations
	• Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	
`BypassNestedModerationEnabled`	Gets or sets whether moderators of parent groups are allowed to moderate nested groups for which moderation is enabled. This attribute can take one of the following values: • **`TRUE`**: Specifies that email messages approved by parent group moderators bypass any moderation in nested groups. • **`FALSE`**: Specifies that email messages approved by parent group moderators still can be moderated in nested groups.	Read, Write
`CreateDTMFMap`	Sets whether to create a dual-tone multi-frequency (DTMF) map for the distribution group. This attribute can take one of the following values: • **`TRUE`**: Specifies to create a DTMF map for the distribution group.	Write

Attribute	Description	Supported operations
	• `FALSE`: Specifies not to create a DTMF map for the distribution group.	
`CustomAttribute1`	Get or set the additional custom values you specify.	Read, Write
`CustomAttribute2`		
`CustomAttribute3`		
`CustomAttribute4`		
`CustomAttribute5`		
`CustomAttribute6`		
`CustomAttribute7`		
`CustomAttribute8`		
`CustomAttribute9`		
`CustomAttribute10`		
`CustomAttribute11`		
`CustomAttribute12`		
`CustomAttribute13`		
`CustomAttribute14`		
`CustomAttribute15`		
`Description`	Gets or sets the description of the distribution group.	Read, Write
`DisplayName`	Gets or sets the display name of the distribution group.	Read, Write
`EmailAddresses`	Gets or sets the email alias of the distribution group.	Read, Write
`ExtensionCustomAttribute1`	Get or set the additional custom values you specify. These attributes are multivalued. To specify multiple values, use a comma as a separator.	Read, Write
`ExtensionCustomAttribute2`		
`ExtensionCustomAttribute3`		
`ExtensionCustomAttribute4`		
`ExtensionCustomAttribute5`		

Attribute	Description	Supported operations
`GrantSendOnBehalfTo`	Gets or sets the senders that can send messages on behalf of the distribution group. This reference attribute can take senders in any of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute only accepts the following object type: • `Mailbox`	Read, Write
`HiddenFromAddressListsEnabled`	Gets or sets whether or not Microsoft 365 hides the distribution group from the address lists. This attribute can take one of the following values: • **`TRUE`**: Specifies to hide the distribution group from the address lists. • **`FALSE`** (default): Specifies to display the distribution group in the address lists.	Read, Write
`IgnoreNamingPolicy`	Sets whether or not to ignore the naming policy applicable to the distribution groups created in the organization. This attribute can take one of the	Write

Attribute	Description	Supported operations
	following values: • **TRUE**: Specifies to ignore the applicable naming policy. • **FALSE**: Specifies to use the applicable naming policy.	
`IsSecurity`	Gets or sets whether the distribution group is a security distribution group.	Read, Write NOTE: This attribute allows you to write data only when you use the Microsoft 365 Connector to perform a create operation in Microsoft 365.
`MailTip`	Gets or sets the message displayed to senders when they start writing an email message to the distribution group.	Read, Write
`MailTipTranslations`	Gets or sets the MailTip message translations in additional languages. This attribute accepts the following format: `<LanguageLocale>:<MailTipMessageTranslation>`	Read, Write

Attribute	Description	Supported operations
	A MailTip message translation cannot exceed 250 characters.	
`ManagedBy`	Gets or sets the owner of the distribution group. This reference attribute accepts the following object types: • `Mailbox` • `MailUser`	Read, Write
`Member`	Gets or sets the members of the distribution group by using their Object IDs.	Read, Write NOTE: This attribute only allows you to write data when you use the Microsoft 365 Connector to perform an update operation in Microsoft 365.
`MemberDepartRestriction`	Gets or sets the restrictions applicable to the members who want to leave the distribution group. This attribute can take one of the following values: • `Open`	Read, Write

Attribute	Description	Supported operations
	• `Closed` • `ApprovalRequired`	
`MemberJoinRestriction`	Gets or sets the restrictions applicable to the members who want to join the distribution group. This attribute can take one of the following values: • `Open` • `Closed` • `ApprovalRequired`	Read, Write
`Member`	Gets or sets the members of the distribution group	Read, Write
`ModeratedBy`	Gets or sets the users who are moderating the messages sent to the distribution group. To specify multiple users, use a comma as a separator. This reference attribute can take users in any of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This attribute is required if you set the value of the `ModerationEnabled` attribute to **`TRUE`**. This reference attribute accepts the following object types:	Read, Write

Attribute	Description	Supported operations
	• `Mailbox` • `MailUser`	
`ModerationEnabled`	Gets or sets whether moderation is enabled for the distribution group. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`Name`	Gets or sets the name of the distribution group.	Read, Write
`Notes`	Gets or sets notes about the distribution group.	Read, Write NOTE: This attribute allows you to write data only when you use the Microsoft Office 365 Connector to create an object in Microsoft 365.
`ObjectID`	Gets the unique object identifier (GUID).	Read
`PrimarySmtpAddress`	Gets or sets primary SMTP address of the distribution group.	Read, Write

Attribute	Description	Supported operations
`PrimarySmtpAddress`	Gets or sets the primary email address of the distribution group.	Read, Write
`RejectMessagesFrom`	Gets or sets the senders whose messages to the distribution group are rejected. This attribute can take senders in one of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `Contact` • `Mailbox`	Read, Write
`RejectMessagesFromDLMembers`	Gets or sets the distribution groups whose members cannot send email messages to the distribution group (their messages are rejected). This reference attribute can take distribution groups in one of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account	Read, Write

Attribute	Description	Supported operations
	• GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	
`RejectMessagesFromSendersOrMembers`	Gets or sets the senders that cannot send email messages to the distribution group (their messages are rejected). This reference attribute can take senders in one of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `Contact` • `DynamicDistributionGroup` • `DistributionGroup` • `Mailbox`	Read, Write
`ReportToManagerEnabled`	Gets or sets whether delivery reports	Read,

Attribute	Description	Supported operations
	are sent to the manager of the distribution group. This attribute can take one of the following values: • TRUE • FALSE	Write
ReportToOriginatorEnabled	Gets or sets whether delivery reports are sent to the senders who sent email messages to the distribution group.	Read, Write
RequireSenderAuthenticationEnabled	Gets or sets whether the senders that send messages to this distribution group must be authenticated. This attribute can take one of the following values: • TRUE • FALSE	Read, Write
SendModerationNotifications	Gets or sets whether to send status notifications to senders when a message they send to the moderated distribution group is rejected by a moderator. This attribute can take one of the following values: • **Always**: Specifies that notifications are sent to all senders. • **Internal**: Specifies that notifications are only sent to the senders internal to your organization. • **Never**: Specifies that all status notifications are disabled.	Read, Write
SendOofMessageToOriginatorEnabled	Gets or sets a value that specifies whether or not to deliver out-of-office messages to the user who sent an email message to the distribution group.	Read, Write

Attribute	Description	Supported operations
	This attribute can take one of the following values: • `TRUE` • `FALSE`	
`SimpleDisplayName`	Gets or sets an alternate description of the distribution group in a situation where a limited set of characters is allowed. The limited set of characters includes ASCII characters from 26 to 126.	Read, Write
`UMDtmfMap`	Gets or sets whether to create a user-defined DTMF map for the distribution group if it has Unified Messaging enabled.	Read, Write
`WindowsEmailAddress`	Gets or sets the email address of the distribution group stored in Active Directory.	Read, Write

Domain object attributes

Table 77: Domain object attributes

Attribute	Description	Supported operations
Authentication	Gets the authentication method with which the domain in Microsoft 365 authenticates users. This attribute can take one of the following values: • **`Managed`**: Indicates that the domain uses Microsoft 365 authentication. • **`Federated`**: Indicates that the domain uses Single Sign-on (SSO) to authenticate users.	Read
DomainName	Gets the domain name in Microsoft 365.	Read
DomainServices	Gets the Microsoft 365 services available in the domain.	Read
IsDefault	Gets whether the domain is default in Microsoft 365.	Read

Attribute	Description	Supported operations
IsInitial	Gets whether the domain is initial in Microsoft 365.	Read
ObjectID	Gets the unique object identifier (GUID).	Read
Status	Gets whether the domain is verified with Microsoft 365. This attribute can take one of the following values: • **`Verified`**: Indicates that the domain is verified. • **`Unverified`**: Indicates that the domain is not verified.	Read

DynamicDistributionGroup object attributes

Table 78: DynamicDistributionGroup object attributes

Attribute	Description	Supported operations
`AcceptMessagesOnlyFrom`	Gets or sets the senders that can send email messages to the dynamic distribution group. This reference attribute can take senders in any of the following formats: • Alias • Canonical name • Display name • DN • Exchange DN • GUID • Name • Primary SMTP email address This reference attribute accepts the following object types: • `MailUser`	Read, Write

Attribute	Description	Supported operations
	• `Mailbox` • `Contact`	
`AcceptMessagesOnlyFromDLMembers`	Gets or sets the distribution groups whose members are allowed to send email messages to the dynamic distribution group. This reference attribute accepts any of the following values for the distribution groups: • DN • Canonical name • GUID • Name • Display name • Legacy Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	Read, Write
`AcceptMessagesOnlyFromSendersOrMembers`	Gets or sets the senders who can send email messages to the dynamic distribution group. This reference attribute can take any of the following values for the senders:	Read, Write

Attribute	Description	Supported operations
	• DN • Canonical name • GUID • Name • Display name • Alias • Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	
`Alias`	Gets or sets the alias of the dynamic distribution group.	Read, Write
`BypassModerationFromSendersOrMembers`	Gets or sets the senders whose messages bypass moderation for the dynamic distribution group. This reference attribute can take any of the following values for the senders: • DN • Canonical name • GUID • Name	Read, Write

Attribute	Description	Supported operations
	• Display name • Legacy Exchange DN • Primary SMTP email address The values in this attribute do not apply to the senders that are the moderators of the dynamic distribution group. This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	

Attribute	Description	Supported operations
`ConditionalCustomAttribute1`	Allow you to get or set recipients based on the corresponding `CustomAttribute<Number>` value. For example, `ConditionalCustomAttribute1` corresponds to `CustomAttribute1`, `ConditionalCustomAttribute2` corresponds to `CustomAttribute2`, and so on.	Read, Write
`ConditionalCustomAttribute2`		
`ConditionalCustomAttribute3`		
`ConditionalCustomAttribute4`		
`ConditionalCustomAttribute5`		
`ConditionalCustomAttribute6`		
`ConditionalCustomAttribute7`		
`ConditionalCustomAttribute8`		
`ConditionalCustomAttribute9`		
`ConditionalCustomAttribute10`		
`ConditionalCustomAttribute11`		
`ConditionalCustomAttribute12`		
`ConditionalCustomAttribute13`		
`ConditionalCustomAttribute14`		
`ConditionalCustomAttribute15`		
`ConditionalDepartment`	Uses the **Department** field to get or set the recipients used to build the dynamic distribution group. A comma that separates values of this multivalued attribute acts as the OR operator.	Read, Write NOTE: When writing data using this attribute, you cannot use the `RecipientFilter` attribute to write data.
`ConditionalStateOrProvince`	Uses the **State/Province** field to get or set the recipients used to build the dynamic	Read, Write

Attribute	Description	Supported operations
	distribution group. A comma that separates values of this multivalued attribute acts as the OR operator.	
`CustomAttribute1`	Get or set the additional custom values you specify.	Read, Write
`CustomAttribute2`		
`CustomAttribute3`		
`CustomAttribute4`		
`CustomAttribute5`		
`CustomAttribute6`		
`CustomAttribute7`		
`CustomAttribute8`		
`CustomAttribute9`		
`CustomAttribute10`		
`CustomAttribute11`		
`CustomAttribute12`		
`CustomAttribute13`		
`CustomAttribute14`		
`CustomAttribute15`		
`DisplayName`	Gets or sets the display name of the dynamic distribution group.	Read, Write
`EmailAddresses`	Gets or sets the email addresses of the dynamic distribution group. When specifying two or more email addresses in this multivalued attribute, use a comma as a separator.	Read, Write
`GrantSendOnBehalfTo`	Gets or sets the	Read, Write

Attribute	Description	Supported operations
	distinguished name (DN) of other senders that can send messages on behalf of the dynamic distribution group. This reference attribute only accepts the following object type: • `Mailbox`	
`IncludedRecipients`	Gets or sets the recipient types used to build the dynamic distribution group. This attribute can take the following values: • `AllRecipients` • `MailContacts` • `MailGroups` • `MailUsers` • `MailboxUsers` • `Resources` • `None` NOTE: You can use combinations of these values, except the `AllRecipients` value. No other value can be used along with the `AllRecipients` value.	Read, Write
`LdapRecipientFilter`	Gets the recipient filter that was created by using the `RecipientFilter` attribute.	Read
`ManagedBy`	Gets or sets the owner of the dynamic distribution group.	Read, Write

Attribute	Description	Supported operations
	This reference attribute accepts the following object types: • `Mailbox` • `MailUser`	
`ManagedBy`	Gets or sets the name of the mail-enabled user, group, or contact displayed on the **Managed by** tab of the Active Directory object. This reference attribute accepts the name in one of the following formats: • Alias • Canonical DN • Display Name • Distinguished Name (DN) • Domain\Account • GUID • Immutable ID • Legacy Exchange DN • SMTP Address • User Principal Name This reference attribute accepts the following object types: • `Mailbox` • `MailUser`	Read, Write
`ModeratedBy`	Gets or sets the users who are moderating the messages sent to the dynamic	Read, Write

Attribute	Description	Supported operations
	distribution group. To specify multiple users, use a comma as a separator. This reference attribute is required if you set the value of the `ModerationEnabled` attribute to **`TRUE`**. This reference attribute accepts the following object types: • `Mailbox` • `MailUser`	
`ModerationEnabled`	Gets or sets whether moderation is enabled for the dynamic distribution group. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`Name`	Gets or sets the name of the dynamic distribution group.	Read, Write
`Notes`	Gets or sets comments for the dynamic distribution group.	Read, Write
`ObjectID`	Gets the unique object identifier (GUID).	Read
`PhoneticDisplayName`	Gets or sets a phonetic pronunciation of the value specified in the `DisplayName` attribute.	Read, Write
`PrimarySmtpAddress`	Gets or sets the primary return SMTP email address of the	Read, Write

Attribute	Description	Supported operations
	dynamic distribution group. You can use this attribute if the group has two or more SMTP email addresses.	
`RecipientContainer`	Gets or sets the recipients used to build the dynamic distribution group, based on their location in Active Directory. This attribute can take the canonical name of the Active Directory Organizational Unit (OU) or domain where the recipients reside. When this attribute is omitted, the local container is used.	Read, Write
`RecipientFilter`	Gets or sets the mail-enabled recipients to be included in the dynamic distribution group. This attribute accepts OPATH filtering syntax. Syntax example: `((Company -eq 'MyCompany') -and (City -eq 'London'))`	Read, Write When writing data using this attribute, you cannot use any of the following attributes to write data: • `IncludedRecipients` • `ConditionalCompany` • `ConditionalCustomAttribute<Number>` • `ConditionalDepartment` • `ConditionalStateOrProvince`
`RejectMessagesFrom`	Gets or sets the senders whose messages to the dynamic distribution group are rejected. This reference attribute can take senders in one of the following	Read, Write

Attribute	Description	Supported operations
	formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `Contact` • `Mailbox`	
`RejectMessagesFromDLMembers`	Gets or sets the distribution groups whose members cannot send email messages to the dynamic distribution group (their messages are rejected). This reference attribute can take distribution groups in one of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account	Read, Write

Attribute	Description	Supported operations
	• GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal name This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	
`RejectMessagesFromSendersOrMembers`	Gets or sets the senders that cannot send email messages to the dynamic distribution group (their messages are rejected). This reference attribute can take senders in one of the following formats: • Alias • Canonical DN • Display name • Distinguished name (DN) • Domain\account • GUID • Immutable ID • Legacy Exchange DN • SMTP address • User principal	Read, Write

Attribute	Description	Supported operations
	name This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox`	
`ReportToManagerEnabled`	Gets or sets a value that specifies whether or not to send delivery reports to the dynamic distribution group manager. This Boolean attribute can take one of the following values: • **TRUE**: Indicates that delivery reports are enabled. • **FALSE** (default): Indicates that delivery reports are disabled.	Read, Write
`ReportToOriginatorEnabled`	Gets or sets a value that specifies whether or not to send a delivery reports to the user who sent an email message to the dynamic distribution group. This Boolean attribute can take one of the following values: • **TRUE**: Indicates that delivery	Read, Write

Attribute	Description	Supported operations
	reports are enabled. • **FALSE** (default): Indicates that delivery reports are disabled.	
`SendModerationNotifications`	Gets or sets whether or not to send a notification to the sender whose message to the moderated dynamic distribution group is rejected by a moderator. This attribute can take one of the following values: • **Always**: Indicates that moderation notifications are sent to all senders. • **Internal**: Indicates that moderation notifications are sent to the internal senders only. • **Never**: Indicates that moderation notifications are disabled.	Read, Write
`SendOofMessageToOriginatorEnabled`	Gets or sets a value that specifies whether or not to deliver out-of-office messages to the user who sent an e-mail message to the dynamic distribution group.	Read, Write

Attribute	Description	Supported operations
	This attribute can take one of the following values: • TRUE • FALSE	

ExternalAccessPolicy object attributes

Table 79: ExternalAccessPolicy object attributes

Attribute	Description	Supported operations
Anchor	Gets the Anchor property value of the policy.	Read
Description	Gets the policy description.	Read
Identity	Gets the unique identifier assigned to the policy.	Read
Members	Gets the users who have been assigned the policy.	Read
ObjectID	Gets the unique object identifier (GUID).	Read

HostedVoicemailPolicy object attributes

Table 80: HostedVoicemailPolicy object attributes

Attribute	Description	Supported operations
Anchor	Gets the Anchor property value of the policy.	Read
Description	Gets the policy description.	Read
Identity	Gets the unique identifier assigned to the policy.	Read
Members	Gets the users who have been assigned the policy.	Read
ObjectID	Gets the unique object identifier (GUID).	Read

LicensePlanService object attributes

Table 81: LicensePlanService object attributes

Attribute	Description	Supported operations
`AssignedLicenses`	Gets the number of used licenses in Microsoft 365. This number includes both valid and expired licenses that are currently assigned.	Read
`ExpiredLicenses`	Gets the number of expired licenses in Microsoft 365.	Read
`ObjectID`	Gets the unique object identifier (GUID).	Read
`PlanDisplayName`	Gets the name of the currently used license plan name in the form it is displayed on the Microsoft 365 graphical user interface.	Read
`PlanName`	Gets the name of the currently used license plan in the form it is returned by the Windows PowerShell cmdlets for Microsoft 365.	Read
`ReducedFunctionalityLicenses`	Gets the number of licenses that are currently in the reduced functionality mode (RFM).	Read
`RelatedAttributeName`	Gets the name of the attribute in the Microsoft 365 Connector schema that allows you to work (for example, read and write) with the specified Microsoft 365 service.	Read
`ServiceDisplayName`	Gets the license service name in the form it is displayed on the Microsoft 365 graphical user interface. Service names are the names of the check boxes displayed under a license plan.	Read
`ServiceName`	Gets the license service name in the form it is returned by the Windows PowerShell cmdlets for Microsoft 365.	Read
`ValidLicenses`	Gets the number of valid licenses in Microsoft 365. This number includes both assigned and available licenses.	Read

Mailbox object attributes

Table 82: Mailbox object attributes

Attribute	Description	Supported operations
`AcceptMessagesOnlyFrom`	Gets or sets the senders that can send email messages to the specified mailbox. This reference attribute accepts any of the following values for the distribution groups: • DN • Canonical name • GUID • Name • Display name • Alias • Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `MailUser` • `Mailbox` • `Contact`	Read, Write
`AcceptMessagesOnlyFromDLMembers`	Gets or sets the distribution groups whose members are allowed to send email messages to the specified mailbox. This reference attribute accepts any of the following values for the distribution groups: • DN • Canonical name • GUID • Name • Display name	Read, Write

Attribute	Description	Supported operations
	• Legacy Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	
`AcceptMessagesOnlyFromSendersOrMembers`	Gets or sets the senders who can send email messages to the specified mailbox. This reference attribute can take any of the following values for the senders: • DN • Canonical name • GUID • Name • Display name • Alias • Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	Read, Write
`Alias`	Gets or sets the alias of the mailbox user.	Read, Write
`ApplyMandatoryProperties`	Sets whether to modify the mandatory properties of a legacy mailbox. For example, you can use this attribute to remove the `legacyMailbox` tag from a	Write

Attribute	Description	Supported operations
	legacy mailbox residing on an Exchange Server 2010 or check whether this tag exists on the mailbox. This attribute can take one of the following values: • **TRUE**: Specifies that the legacyMailbox tag does not exist on the mailbox. • **FALSE**: Specifies that the legacyMailbox tag exists on the mailbox.	
`ArchiveName`	Gets or sets the name of the archive mailbox. This is the name displayed on the user interface in Microsoft Office Outlook Web App and Microsoft Outlook.	Read, Write
`AuditAdmin`	Gets or sets the operations to log for administrators. This attribute can take the following values: • `None` • `Update` • `Copy` • `Move` • `MoveToDeletedItems` • `SoftDelete` • `HardDelete` • `FolderBind` • `SendAs` • `SendOnBehalf` • `MessageBind` To enable mailbox audit logging, set the value of the `AuditEnabled` attribute to **TRUE**.	Read, Write

Attribute	Description	Supported operations
`AuditDelegate`	Gets or sets the operations to log for delegate users. This attribute can take the following values: • `None` • `Update` • `Move` • `MoveToDeletedItems` • `SoftDelete` • `HardDelete` • `FolderBind` • `SendAs` • `SendOnBehalf` To enable mailbox audit logging, set the value of the `AuditEnabled` attribute to **`TRUE`**.	Read, Write
`AuditEnabled`	Gets or sets whether mailbox audit logging is enabled or disabled. If mailbox audit logging is enabled, the operations specified in the `AuditAdmin`, `AuditDelegate`, and `AuditOwner` attributes are logged. This attribute can take one of the following values: • **`TRUE`**: Specifies that mailbox audit logging is enabled. • **`FALSE`**: Specifies that mailbox audit logging is disabled.	Read, Write
`AuditLogAgeLimit`	Gets or sets the retention period for the mailbox audit logs. Logs whose age exceeds the specified retention period are deleted. This attribute accepts the following format for the retention period:	Read, Write

Attribute	Description	Supported operations

	`DD.HH:MM:SS`
	The maximum value this attribute can accept is `24855.03:14:07`
	
	Example 1
	`30.05:00:00`
	Specifies to retain the mailbox audit logs for 30 days and 5 hours.
	
	Example 2
	`00.00:00:00`
	The mailbox audit logs are never deleted.	

`BypassModerationFromSendersOrMembers`	Gets or sets the senders whose messages bypass moderation for the mailbox. This reference attribute can take any of the following values for the senders: • DN • Canonical name • GUID • Name • Display name • Legacy Exchange DN • Primary SMTP email address The values in this attribute do not apply to the senders that are the moderators of the mailbox. This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup`	Read, Write

Attribute	Description	Supported operations
	• `Mailbox` • `MailUser`	
`CalendarRepairDisabled`	Gets or sets whether the calendar items in the mailbox can be repaired by the Calendar Repair Assistant. This attribute can take one of the following values: • **`TRUE`**: Specifies that repair operations are enabled. • **`FALSE`**: Specifies that repair operations are disabled.	Read, Write
`CalendarVersionStoreDisabled`	Gets or sets whether to log calendar changes in the mailbox. This attribute can take one of the following values: • **`TRUE`**: Specifies that calendar changes are logged. • **`FALSE`**: Specifies that calendar changes are not logged.	Read, Write
`CreateDTMFMap`	Sets whether to create a dual-tone multi-frequency map for the mailbox user.	Write

Attribute	Description	Supported operations
`CustomAttribute1`	Get or set the additional custom values you specify.	Read, Write
`CustomAttribute2`		
`CustomAttribute3`		
`CustomAttribute4`		
`CustomAttribute5`		
`CustomAttribute6`		
`CustomAttribute7`		
`CustomAttribute8`		
`CustomAttribute9`		
`CustomAttribute10`		
`CustomAttribute11`		
`CustomAttribute12`		
`CustomAttribute13`		
`CustomAttribute14`		
`CustomAttribute15`		
`DeliverToMailboxAndForward`	Gets or sets whether this mailbox receives forwarded messages in case message forwarding to another address is configured for the mailbox. This attribute can take one of the following values: • **`TRUE`**: Specifies that messages are delivered to this mailbox and to the forwarding address. • **`FALSE`**: Specifies that messages are delivered to the forwarding address only and not to this mailbox.	Read, Write
`DisplayName`	Gets or sets the display name of the user account associated with the mailbox.	Read, Write

Attribute	Description	Supported operations
`EmailAddresses`	Gets or sets all the proxy addresses of the mailbox. The proxy addresses also include the primary SMTP address. When writing proxy addresses using this attribute, make sure the specified addresses are valid, because the addresses are not validated by Exchange.	Read, Write
`EndDateForRetentionHold`	Gets or sets the retention hold end date for messaging records management (MRM). To enable or disable retention hold, use the `RetentionHoldEnabled` attribute.	Read, Write
`ExternalDirectoryObjectId`	Gets the GUID of the user to whom the mailbox belongs.	Read
`ExternalOofOptions`	Gets or sets whether Out of Office message is sent to external senders. This attribute can take one of the following values: • `External` • `InternalOnly`	Read, Write
`ExtensionCustomAttribute1`	Get or set the additional custom values you specify. These attributes are multivalued.	Read, Write
`ExtensionCustomAttribute2`		
`ExtensionCustomAttribute3`		
`ExtensionCustomAttribute4`		
`ExtensionCustomAttribute5`		
`ForwardingAddress`	Gets or sets a forwarding address for the mailbox.	Read, Write
`ForwardingSmtpAddress`	Gets or sets a forwarding SMTP address for the mailbox.	Read, Write
`GrantSendOnBehalfTo`	Gets or sets the distinguished name (DN) of other senders that can send messages on behalf of the mailbox. This reference attribute only accepts the	Read, Write

Attribute	Description	Supported operations
	following object type: • `Mailbox`	
`HiddenFromAddressListsEnabled`	Gets or sets whether this mailbox is hidden from address lists. This attribute can take one of the following values: • **`TRUE`**: Specifies that the mailbox is hidden from address lists. • **`FALSE`**: Specifies that the mailbox is shown in address lists.	Read, Write
`ImmutableId`	Gets or sets a unique immutable ID in the form of an SMTP address.	Read, Write
`IsEquipment`	Gets or sets whether the mailbox belongs to a piece of equipment. This attribute can take one of the following values: • **`TRUE`**: Indicates that the mailbox is an equipment mailbox. • **`FALSE`**: Indicates that the mailbox is not an equipment mailbox.	Read, Write
`IsRegular`	Gets or sets whether the mailbox belongs to a user. This attribute can take one of the following values: • **`TRUE`**: Indicates that the mailbox belongs to a user. • **`FALSE`**: Indicates that the mailbox does not belong to a user.	Read, Write
`IsRoom`	Gets or sets whether the mailbox belongs to a room. This attribute can take one of the following values: • **`TRUE`**: Indicates that the mailbox belongs to a room.	Read, Write

Attribute	Description	Supported operations
	• **FALSE**: Indicates that the mailbox does not belong to a room.	
IsShared	Gets or sets whether the mailbox is shared. This attribute can take one of the following values: • **TRUE**: Indicates that the mailbox is shared. • **FALSE**: Indicates that the mailbox is not shared.	Read, Write
IssueWarningQuota	Gets or sets the mailbox size at which a warning message is sent to the mailbox user. To specify a mailbox size, use an integer value. To disable warning, set the value of this attribute to **Unlimited**. The value set on a mailbox by using this attribute overrides the value specified for the entire mailbox database.	Read, Write
IsValid	Gets whether or not the mailbox object is configured correctly. This attribute can take one of the following values: • **TRUE**: Indicates that the mailbox object is configured correctly. • **FALSE**: Indicates that the mailbox object is not configured correctly.	Read
Languages	Gets or sets preferred languages for the mailbox in the order of their priority.	Read, Write
LitigationHoldDate	Gets or sets the date when the mailbox is placed on litigation hold. This date is only used for informational or reporting purposes.	Read, Write
LitigationHoldDuration	Gets or sets the litigation hold duration for the mailbox in days.	Read, Write

Attribute	Description	Supported operations
`LitigationHoldEnabled`	Gets or sets whether litigation hold is enabled for the mailbox. When a mailbox is on litigation hold, messages cannot be deleted from the mailbox. This attribute can take one of the following values: • **`TRUE`**: Specifies that litigation hold is enabled. • **`FALSE`**: Specifies that litigation hold is not enabled.	Read, Write
`LitigationHoldOwner`	Gets or sets the user who put the mailbox on litigation hold.	Read, Write
`MailboxPlan`	Gets or sets the mailbox plan name associated with the mailbox. When setting a mailbox plan name, make sure that plan name exists for the organization in which the mailbox resides.	Read, Write
`MailTip`	Gets or sets the message displayed to senders when they start writing an email message to this recipient.	Read, Write
`MailTipTranslations`	Gets or sets the `MailTip` message translations in additional languages. This attribute accepts the following format: `<LanguageLocale>:<MailTipMessageTranslation>` A `MailTip` message translation cannot exceed 250 characters.	Read, Write
`MessageTrackingReadStatusEnabled`	Gets or sets whether the read status of sent messages is provided to the senders who sent messages to this mailbox. This attribute can take one of the following values: • `TRUE`	Read, Write

Attribute	Description	Supported operations
	• FALSE	
ModeratedBy	Gets or sets the users who are moderating the messages sent to the mailbox. To specify multiple users, use a comma as a separator. This reference attribute is required if you set the value of the ModerationEnabled attribute to **TRUE**. This reference attribute accepts the following object types: • Mailbox • MailUser	Read, Write
ModerationEnabled	Gets or sets whether moderation is enabled for the mailbox. This attribute can take one of the following values: • TRUE • FALSE	Read, Write
Name	Gets or sets the name of the mailbox user. This is the name that displays in the Active Directory Users and Computers tool.	Read, Write
ObjectID	Gets the unique object identifier (GUID).	Read
Office	Gets or sets the Microsoft Office attribute for the mailbox.	Read, Write
Password	Sets the password for the user account associated with the mailbox.	Write
PrimarySmtpAddress	Gets or sets the originating email address displayed to the external recipients of a message sent from the mailbox.	Read, Write
ProhibitSendQuota	Gets or sets the mailbox size at which the mailbox user can no longer send messages.	Read, Write

Attribute	Description	Supported operations
	To specify a mailbox size, use an integer value. To disable the send quota, set the value of this attribute to **`Unlimited`**. The value set on a mailbox by using this attribute overrides the value specified for the entire mailbox database.	
`ProhibitSendReceiveQuota`	Gets or sets the mailbox size at which the mailbox user can no longer send or receive messages. To specify a mailbox size, use an integer value. To disable the send and receive quota, set the value of this attribute to **`Unlimited`**. The value set on a mailbox by using this attribute overrides the value specified for the entire mailbox database.	Read, Write
`RejectMessagesFrom`	Gets or sets the senders whose messages are rejected by the mailbox. This reference attribute accepts the following object types: • `Contact` • `Mailbox`	Read, Write
`RejectMessagesFromDLMembers`	Gets or sets the distribution groups whose members cannot send email messages to the mailbox (their messages are rejected). This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	Read, Write
`RejectMessagesFromSendersOrMembers`	Gets or sets the senders that cannot send email messages to the mailbox (their messages are rejected). This attribute can take any of the following values for the recipients:	Read, Write

Attribute	Description	Supported operations
	• DN • Canonical name • GUID • Name • Display name • Alias • Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox`	
`RequireSenderAuthenticationEnabled`	Gets or sets whether senders must be authenticated. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`ResourceCapacity`	Gets or sets the maximum number of people that can be accommodated by the room to which the mailbox belongs.	Read, Write
`ResourceCustom`	Gets or sets additional information about the resource.	Read, Write
`RetainDeletedItemsFor`	Gets or sets for how long to keep deleted items. This attribute accepts the following format: `DD.HH:MM:SS` **Example**	Read, Write

Attribute	Description	Supported operations
	`10.00:00:00` Specifies to retain deleted items for 10 days 00 hours 00 minutes and 00 seconds.	
`RetentionComment`	Gets or sets a comment on user's hold status. This comment is displayed in Outlook. You can only write the value of this attribute if the value of the `RetentionHoldEnabled` attribute is set to `TRUE`.	Read, Write
`RetentionHoldEnabled`	Gets or sets whether retention hold is enabled for messaging retention policies. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`RetentionPolicy`	Gets or sets the name of a retention policy to be applied to the folders and mail items in this mailbox.	Read, Write
`RetentionUrl`	Gets or sets the URL of a Web site providing additional details about the organization's messaging retention policies.	Read, Write
`RoleAssignmentPolicy`	Gets or sets the management role assignment policy to assign to the mailbox when it is created or enabled. If the assignment policy name you want to specify contains spaces, use quotation marks around the name. If you omit this attribute when creating or enabling a mailbox, the default assignment policy is used. If you do not want to assign an assignment policy, set an empty value	Read, Write

Attribute	Description	Supported operations
	in this attribute.	
`RulesQuota`	Gets or sets the limit for the size of rules for the mailbox. Qualify the value you specify in this attribute by appending `B` (bytes) or `KB` (kilobytes). Unqualified values are treated as bytes. The maximum value this attribute accepts is 256 KB.	Read, Write
`SecondaryAddress`	Sets the secondary address used by the UM-enabled user.	Write
`SecondaryDialPlan`	Sets a secondary UM dial plan to use.	Write
`SendModerationNotifications`	Gets or sets whether to send status notifications to users when a message they sent to the moderated distribution group is rejected by a moderator. This attribute can take one of the following values: • **`Always`**: Specifies that notifications are sent to all senders. • **`Internal`**: Specifies that notifications are only sent to the internal senders in your organization. • **`Never`**: Specifies that all status notifications are disabled.	Read, Write
`SharingPolicy`	Gets or sets the sharing policy associated with the mailbox.	Read, Write
`SimpleDisplayName`	Gets or sets an alternate description of the mailbox in a situation where a limited set of characters is allowed. The limited set of characters includes ASCII characters 26 through 126.	Read, Write
`SingleItemRecoveryEnabled`	Gets or sets whether to enable or disable the purging of recovery items. This attribute can take one of the	Read, Write

Attribute	Description	Supported operations
	following values: • **TRUE**: Specifies to disable the purging of recovery items. • **FALSE**: Specifies to enable the purging of recovery items.	
`UMDtmfMap`	Gets or sets whether to create a user-defined DTMF map for the user if it has Unified Messaging enabled.	Read, Write
`UsageLocation`	Gets a two-letter country code that defines the location of the user. Usage location determines the services available to the user. For example: • FR • GB • NL	Read
`UserCertificate`	Gets or sets the digital certificate used to sign email messages of the user.	Read, Write
`UserPrincipalName`	Gets or sets the logon name of the mailbox user.	Read, Write
`UserSMimeCertificate`	Gets or sets the SMIME certificate used to sign email messages of the user.	Read, Write

MailUser object attributes

Table 83: MailUser object attributes

Attribute	Description	Supported operations
`AcceptMessagesOnlyFrom`	Gets or sets the senders that can send email messages to the specified mail user. This reference attribute can take senders in any of the following formats:	Read, Write

Attribute	Description	Supported operations
	• Alias • Canonical name • Display name • DN • Exchange DN • GUID • Name • Primary SMTP email address This reference attribute accepts the following object types: • `MailUser` • `Mailbox` • `Contact`	
`AcceptMessagesOnlyFromDLMembers`	Gets or sets the distribution groups whose members are allowed to send email messages to the specified mail user. This reference attribute can take distribution groups in any of the following formats: • Canonical name • Display name • DN • GUID • Legacy Exchange DN • Name • Primary SMTP email address This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	Read, Write
`AcceptMessagesOnlyFromSendersOr`	Gets or sets the senders who can send	Read,

Attribute	Description	Supported operations
`Members`	email messages to the mail user. This reference attribute can take senders in any of the following formats: • Alias • Canonical name • Display name • DN • GUID • Name • Legacy Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	Write
`Alias`	Gets or sets the alias of the mail user.	Read, Write
`ArchiveName`	Gets the name of the archive mailbox. This is the name displayed on the user interface in Microsoft Office Outlook Web App and Microsoft Outlook.	Read
`BypassModerationFromSendersOrMembers`	Gets or sets the senders whose messages bypass moderation for the mail user. This reference attribute can take any of the following values for the senders: • Alias • Canonical name • Display name	Read, Write

Attribute	Description	Supported operations
	• DN • GUID • Name • Legacy Exchange DN • Primary SMTP email address Moderation does not apply to the senders designated as moderators for the mail user. This reference attribute accepts the following object types: • `Contact` • `DistributionGroup` • `DynamicDistributionGroup` • `Mailbox` • `MailUser`	
`CalendarVersionStoreDisabled`	Gets or sets whether to log calendar changes for the mail user. This attribute can take one of the following values: • **`TRUE`**: Specifies that calendar changes are logged. • **`FALSE`**: Specifies that calendar changes are not logged.	Read, Write
`CreateDTMFMap`	Sets whether to create a dual-tone multi-frequency map for the mail user.	Write

Attribute	Description	Supported operations
`CustomAttribute1`	Get or set the additional custom values you specify.	Read, Write
`CustomAttribute2`		
`CustomAttribute3`		
`CustomAttribute4`		
`CustomAttribute5`		
`CustomAttribute6`		
`CustomAttribute7`		
`CustomAttribute8`		
`CustomAttribute9`		
`CustomAttribute10`		
`CustomAttribute11`		
`CustomAttribute12`		
`CustomAttribute13`		
`CustomAttribute14`		
`CustomAttribute15`		
`DeliverToMailboxAndForward`	Gets whether messages sent to the mail user are forwarded to another address in case message forwarding is configured. This attribute can take one of the following values: • **`TRUE`**: Specifies that messages are delivered to the mail user and to the forwarding address. • **`FALSE`**: Specifies that messages are delivered to the forwarding address only.	Read
`DisplayName`	Gets or sets the display name of the mail user.	Read, Write
`EmailAddresses`	Gets or sets the email alias of the mail user.	Read, Write

Attribute	Description	Supported operations
`EndDateForRetentionHold`	Gets the retention hold end date for messaging records management (MRM). To enable or disable retention hold, use the `RetentionHoldEnabled` attribute.	Read

`ExtensionCustomAttribute1`
`ExtensionCustomAttribute2`
`ExtensionCustomAttribute3`
`ExtensionCustomAttribute4`
`ExtensionCustomAttribute5`
	Get or set the additional custom values you specify. These attributes are multivalued. To specify multiple values, use a comma as a separator.	Read, Write

`ExternalDirectoryObjectId`	Gets the GUID of the mail user.	Read
`ExternalEmailAddress`	Gets or sets an email address outside of the mail user's organization. Messages sent to the mail user are delivered to this external address.	Read, Write
`FederatedIdentity`	Allows you to associate an on-premises Active Directory user with the Microsoft 365 mail user.	Write
`ForwardingAddress`	Gets the forwarding address for the mail user.	Read
`GrantSendOnBehalfTo`	Gets or sets the distinguished name (DN) of other senders that can send messages on behalf of the mail user. This reference attribute only accepts the following object type: • `Mailbox`	Read, Write
`HiddenFromAddressListsEnabled`	Gets or sets whether the mail user is hidden from address lists. This attribute can take one of the following values: • **`TRUE`**: Specifies that the mail user is hidden from address lists. • **`FALSE`**: Specifies that the mail user	Read, Write

Attribute	Description	Supported operations
	is shown in address lists.	
`ImmutableId`	Gets or sets a unique immutable ID in the form of an SMTP address.	Read, Write
`LitigationHoldDate`	Gets the date when the mail user's mailbox is placed on litigation hold.	Read
`LitigationHoldEnabled`	Gets whether litigation hold is enabled for the mail user's mailbox. When a mailbox is on litigation hold, messages cannot be deleted from the mailbox. This attribute can take one of the following values: • **`TRUE`**: Specifies that litigation hold is enabled. • **`FALSE`**: Specifies that litigation hold is not enabled.	Read
`LitigationHoldOwner`	Gets the user who enabled litigation hold on the mailbox. This attribute can only be used for informational or reporting purposes.	Read
`MacAttachmentFormat`	Gets or sets the Apple Macintosh operating system attachment format for messages sent to the mail user. This attribute can take the following values: • `BinHex` • `UuEncode` • `AppleSingle` • `AppleDouble`	Read, Write
`MailTip`	Gets or sets the message displayed to senders when they start writing an email message to the mail user.	Read, Write
`MailTipTranslations`	Gets or sets the MailTip message translations in additional languages. This attribute accepts the following format:	Read, Write

Attribute	Description	Supported operations
	`<LanguageLocale>:<MailTipMessageTranslation>` A MailTip message translation cannot exceed 250 characters.	
`MessageBodyFormat`	Gets or sets the message body format for messages sent to the mail user. The values this attribute can take depend on the value in the `MessageFormat` attribute. When the value in the `MessageFormat` is `Mime`, the `MessageBodyFormat` attribute can take the following values: • `Text` • `Html` • `TextAndHtml` When the value in the `MessageFormat` is `Text`, the `MessageBodyFormat` attribute can only take the `Text` value.	Read, Write
`MessageFormat`	Gets or sets the message format for messages sent to the mail user. This attribute can take the following values: • `Text` • `Mime`	Read, Write
`ModeratedBy`	Gets or sets the moderators who are moderating the messages sent to the distribution group. To specify multiple moderators, use a comma as a separator. This reference attribute is required if you set the value of the `ModerationEnabled` attribute to `TRUE`. This reference attribute accepts the following object types: • `Mailbox`	Read, Write

Attribute	Description	Supported operations
	• `MailUser`	
`ModerationEnabled`	Gets or sets whether moderation is enabled for the distribution group. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`Name`	Gets or sets the name of the mail user.	Read, Write
`ObjectID`	Gets the unique object identifier (GUID).	Read
`Password`	Sets the password for the mail user.	Write
`RejectMessagesFrom`	Gets or sets the senders whose messages to the mail user are rejected. This attribute can take senders in one of the following formats: • Alias • Canonical name • Display name • DN • GUID • Name • Legacy Exchange DN • Primary SMTP email address This reference attribute accepts the following object types: • `Contact` • `Mailbox`	Read, Write
`RejectMessagesFromDLMembers`	Gets or sets the distribution groups whose members cannot send email messages to the mail user (such messages are rejected).	Read, Write

Attribute	Description	Supported operations
	This reference attribute can take distribution groups in one of the following formats: • Alias • Canonical name • Display name • DN • GUID • Legacy Exchange DN • Name • Primary SMTP email address This reference attribute accepts the following object types: • `DistributionGroup` • `DynamicDistributionGroup`	
`RequireSenderAuthenticationEnabled`	Gets or sets whether the senders that send messages to this mail user must be authenticated. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write

`RetainDeletedItemsFor`	Gets for how long to keep deleted items for the mail user.
	This attribute accepts the following format:
	`DD.HH:MM:SS`
	For example:
	`10.00:00:00`
	Specifies to retain deleted items for 10 days 00 hours 00 minutes and 00 seconds.
		Read

Attribute	Description	Supported operations
`RetentionComment`	Gets the comment on the mail user's hold status. This comment is displayed in Outlook. You can only write the value of this attribute if the value of the `RetentionHoldEnabled` attribute is set to `TRUE`.	Read
`RetentionHoldEnabled`	Gets whether retention hold is enabled for messaging retention policies. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read
`RetentionUrl`	Gets the URL of a web page providing additional details about the organization's messaging retention policies.	Read
`SecondaryAddress`	Sets the secondary address used by the Unified Messaging-enabled user.	Write
`SecondaryDialPlan`	Sets a secondary Unified Messaging dial plan for the mail user.	Write
`SendModerationNotifications`	Gets or sets whether to send status notifications to users when a message they sent to the moderated distribution group is rejected by a moderator. This attribute can take one of the following values: • **`Always`**: Specifies that notifications are sent to all senders. • **`Internal`**: Specifies that notifications are only sent to the senders internal to your organization. • **`Never`**: Specifies that all status notifications are disabled.	Read, Write

Attribute	Description	Supported operations
`SimpleDisplayName`	Gets or sets an alternate description of the mailbox in a situation where a limited set of characters is allowed. The limited set of characters includes ASCII characters from 26 to 126.	Read, Write
`SingleItemRecoveryEnabled`	Gets whether the purging of recovery items is enabled. This attribute can take one of the following values: • **`TRUE`**: Specifies to disable the purging of recovery items. • **`FALSE`**: Specifies to enable the purging of recovery items.	Read
`StartDateForRetentionHold`	Gets the start date for retention hold. To use this attribute, you must set the `RetentionHoldEnabled` attribute to `TRUE`.	Read
`UMDtmfMap`	Gets or sets whether to create a user-defined DTMF map for the mail user if it has Unified Messaging enabled.	Read, Write
`UsageLocation`	Gets a two-letter country code that defines the location of the mail user. Usage location determines the services available to the mail user. For example: • FR • GB • NL	Read
`UseMapiRichTextFormat`	Gets or sets a format for the MAPI Rich Text Format messages sent to the mail user. • **`Never`**: Specifies to convert all messages sent to the mail user to the plain text format. • **`Always`**: Specifies to always use the MAPI Rich Text Format (RTF)	Read, Write NOTE: You can only write data by using

Attribute	Description	Supported operations
	for the messages sent to the mail user. • **`UseDefaultSettings`**: Specifies to use the message format set in the MAPI client that sent the message to the mail user.	this attribute when updating an existing object in Microsoft 365.
`UsePreferMessageFormat`	Gets or sets whether the message format specified for the mail user overrides any global settings (such as those configured for the remote domain). This attribute can take one of the following values: • **`TRUE`**: Specifies that the message format set for the mail user overrides any global settings. • **`FALSE`**: Specifies that global settings have precedence over the mail format set for the mail user.	Read, Write
`UserPrincipalName`	Gets or sets the user principal name (UPN) of the mail user.	Read, Write
`WindowsEmailAddress`	Gets or sets the email address for the mail user stored in Active Directory.	Read, Write

PresencePolicy object attributes

Table 84: PresencePolicy object attributes

Attribute	Description	Supported operations
`Anchor`	Gets the Anchor property value of the policy.	Read
`Description`	Gets the policy description.	Read
`Identity`	Gets the unique identifier assigned to the policy.	Read

Attribute	Description	Supported operations
Members	Gets the users who have been assigned the policy.	Read
ObjectID	Gets the unique object identifier (GUID).	Read

SecurityGroup objects attributes

Table 85: SecurityGroup attributes

Attribute	Description	Supported operations
Description	Gets or sets the description of the security group.	Read, Write
DisplayName	Gets or sets the display name of the security group.	Read, Write
Members	Gets or sets the members of the security group.	Read, Write
ObjectID	Gets the unique object identifier (GUID).	Read

SPOSite object attributes

Table 86: SPOSite attributes

Attribute	Description	Supported operations
AllowSelfServiceUpgrade	Gets or sets whether the site collection administrators can upgrade this site collection.	Read, Write
CompatibilityLevel	Gets the major version number of the site collection. This version number is used to perform compatibility checks.	Read
Groups	Gets or sets the site collection groups. This attribute is required to create a site collection in SharePoint Online.	Read, Write
LastContentModifiedDate	Gets the date when the site collection content was last modified.	Read
LocaleId	Gets or sets the Locale ID (LCID) for the site collection.	Read, Write

Attribute	Description	Supported operations
`LockIssue`	Gets or sets the comment that was written when the site collection was locked.	Read
`LockState`	Gets or sets a lock state for the site collection. This attribute can take one of the following values: • **NoAccess**: All traffic to the site collection is blocked. Traffic to sites that have this lock state is redirected to the URL set in the `NoAccessRedirectUrl` attribute of the `SPOTenant` object. If no URL is set in that attribute, a 404 error is returned. • **Unlock**: All traffic to the site collection is allowed.	Read, Write
`ObjectID`	Gets the unique object identifier (GUID).	Read
`Owner`	Gets or sets the owner of the site collection. This attribute is required to create a site collection in SharePoint Online.	Read, Write
`ResourceQuota`	Gets or sets the server resource quota for the site collection.	Read, Write
`ResourceQuotaWarningLevel`	Gets or sets the warning level for the site collection. When the resource usage for the site collection reaches the specified warning level, a notification email is sent.	Read, Write
`ResourceUsageAverage`	Gets average resource usage for the site collection.	Read
`ResourceUsageCurrent`	Gets the current resource usage for the site collection.	Read
`Status`	No description available.	Read, Write
`StorageQuota`	Gets or sets the storage quota limit for the site collection. This attribute is required to create a site collection in SharePoint Online.	Read, Write
`StorageQuotaWarningLevel`	Gets or sets the storage warning level for the site collection.	Read, Write

Attribute	Description	Supported operations
	In SharePoint Online, you can view the current storage warning level in the site collection properties.	
`StorageUsageCurrent`	Gets the current storage usage for the site collection.	Read
`Template`	Gets or sets the template for the site collection.	Read, Write
`TimeZoneId`	Gets or sets the identifier of the time zone for the site collection.	Read, Write
`Title`	Gets or sets the title of the site collection.	Read, Write
`Url`	Gets or sets the website address (URL). In SharePoint Online, you can view the website address in the site collection properties. This attribute is required to create a site collection in SharePoint Online.	Read, Write
`WebsCount`	No description available.	Read

SPOSiteGroup object attributes

Table 87: SPOSiteGroup attributes

Attribute	Description	Supported operations
`LoginName`	Gets or sets the name of the group.	Read, Write
`ObjectID`	Gets the unique object identifier (GUID).	Read
`Owner`	Gets or sets the owner in the group.	Read, Write
`PermissionLevels`	Gets or sets permission levels for the group.	Read, Write
`Site`	Gets or sets the name of the site collection to which the group belongs.	Read, Write
`Users`	Gets or sets users in the group.	Read, Write

SPOWebTemplate object attributes

Table 88: SPOWebTemplate attributes

Attribute	Description	Supported operations
`CompatibilityLevel`	Gets the compatibility level of the web template.	Read
`Description`	Gets the description of the web template.	Read
`DisplayCategory`	Gets the name of the category to which the web template belongs.	Read
`LocaleID`	Gets the Locale ID (LCID) of the web template.	Read
`Name`	Gets the name of the web template.	Read
`ObjectID`	Gets the unique object identifier (GUID).	Read
`Title`	Gets the title of the web template.	Read

SPOTenant object attributes

Table 89: SPOTenant attributes

Attribute	Description	Supported operations
`ExternalServicesEnabled`	Gets or sets the maximum compatibility level for new sites.	Read, Write (update only)
`MinCompatibilityLevel`	Gets or sets the minimum compatibility level for new sites.	Read, Write (update only)
`NoAccessRedirectUrl`	Gets or sets the redirect URL for the `SPOSite` object whose `LockState` attribute value is set to `NoAccess`.	Read, Write (update only)
`ObjectID`	Gets the unique object identifier (GUID).	Read
`ResourceQuota`	Gets or sets the server resource quota available to the organization.	Read, Write (update only)
`ResourceQuotaAllocated`	Gets or sets the server resource quota limit for the organization.	Read, Write (update only)
`StorageQuota`	Gets or sets the storage quota available to the organization.	Read, Write (update only)
`StorageQuotaAllocated`	Gets or sets the storage quota limit for the organization.	Read, Write (update only)

User object attributes

The Microsoft 365 Connector provides the following attributes for the User object in Microsoft 365:

- Attributes Related to License Plans and Services
- Other attributes

Attributes Related to License Plans and Services

These attributes allow you to get or set the license plans and services available to the user in Microsoft 365. The attributes support Read and Write operations.

The names and display names of these attributes are formed dynamically according to the following patterns:

Table 90: Naming patterns for attributes

Item	Naming pattern	Examples
Attribute display name	`<LicensePlanNameOnGUI> - <ServiceNameOnGUI>` In this pattern: `LicensePlanNameOnGUI` is the license plan name as it is displayed on the Microsoft 365 user interface. `ServiceNameOnGUI` is the service name as it is displayed below the corresponding license plan on the Microsoft 365 user interface.	Microsoft 365 Plan E3 - Office Web Apps Microsoft 365 Plan K2 - Exchange Online Kiosk
Attribute name	`<LicensePlanName>-<ServiceName>` In this pattern: `LicensePlanName` is the license plan name in the form used by the Microsoft 365 cmdlets for Windows PowerShell. `ServiceName` is the service name in the corresponding license plan. The service name is displayed in the form used by the Microsoft 365 cmdlets for Windows PowerShell.	`ENTERPRISEPACK-SHAREPOINTWAC` `DESKLESSWOFFPACK-EXCHANGE_S_DESKLESS`

These attributes can take one of the following values:

- **`True`**: Specifies that the service is selected in the corresponding license plan in Microsoft 365.
- **`False`**: Specifies that the service is selected in the corresponding license plan in Microsoft 365.

If necessary, you can modify the display names of Microsoft 365 license plans and services that appear in the Synchronization Service Console. These display names are part of the Office 365 Connector schema and saved in the `O365LicensePlansServices.xml` file located

in the Synchronization Service installation folder (by default, this is `%ProgramFiles%\One Identity\Active Roles\7.4\SyncService`).

For example, you may need to modify the name of a license plan or service in the Microsoft 365 Connector schema when the corresponding name changes in the Microsoft 365 user interface and therefore the related attribute display name becomes outdated in the Synchronization Service Console.

To modify the display names of attributes in the Microsoft 365 Connector schema

1. Open the `O365LicensePlansServices.xml` file located in the Synchronization Service installation folder.
2. In the appropriate XML elements, modify the values of the **`PlanDisplayName`** and **`ServiceDisplayName`** attributes as necessary. See the table below for more information about the XML elements used in the file.
3. When you are finished, click **OK**.

Table 91: XML elements

XML element	Description	Example
`<Plan>`	Defines the name and display name of the attribute related to a particular Microsoft 365 license plan in the Microsoft 365 Connector schema. This element has the following attributes: • **PlanName**: The license plan name in the form used by the Microsoft 365 cmdlets for Windows PowerShell. • **PlanDisplayName**: The license plan name as it displays in the Synchronization Service Console.	**`<Plan PlanName="STANDARDPACK" PlanDisplayName="Microsoft Office 365 Plan E1"/>`**
`<Service>`	Defines the name and display name of the attribute related to a particular Microsoft 365 service in the Microsoft 365 Connector schema. This element has the following attributes:	**`<Service ServiceName="OFFICESUBSCRIPTION" ServiceDisplayName="Office Professional Plus" />`**

XML element	Description	Example
	• **ServiceName**: The service name in the form used by the Microsoft 365 cmdlets for Windows PowerShell. • **ServiceDisplayName**: The service name as it displays in the Synchronization Service Console.	

Other attributes

Table 92: Other attributes

Attribute	Description	Supported operations
`AllowUMCallsFromNonUsers`	Gets or sets whether to exclude or include the user in directory searches. This attribute can take one of the following values: • **`None`**: Specifies to exclude the user from directory searches. • **`SearchEnabled`**: Specifies to include the user in directory searches.	Read, Write
`AlternateEmailAddresses`	Gets or sets the alternate email addresses of the user.	Read, Write
`AssistantName`	Gets or sets the name of the user's assistant.	Read, Write
`BlockCredential`	Gets or sets whether or not the user can sign in and use Microsoft 365 services. This attribute can take one of the following values: • **`TRUE`**: Specifies that user's Microsoft Online Services ID is disabled and the user cannot sign in and use Microsoft 365 services. • **`FALSE`** (default): Specifies that user's Microsoft Online Services ID is enabled and the user can sign in and use Microsoft 365 services.	Read, Write

Attribute	Description	Supported operations
`City`	Gets or sets the user's city.	Read, Write
`Company`	Gets or sets the name of user's company.	Read, Write
`Country`	Gets or sets the user's country.	Read, Write
`CountryOrRegion`	Gets or sets the country or region of the user.	Read, Write
`Department`	Gets or sets the user's department.	Read, Write
`DisplayName`	Gets or sets the display name of the user.	Read, Write
`Fax`	Gets or sets the user's fax number.	Read, Write
`FirstName`	Gets or sets the first name of the user.	Read, Write
`ForceChangePassword`	Gets or sets whether or not the user is forced to change their password the next time the user signs in to Microsoft 365. • **`TRUE`**: Specifies that the user must change their password the next time the user signs in to Microsoft 365. • **`FALSE`** (default): Specifies that the user does not have to change their password the next time the user signs in to Microsoft 365.	Write NOTE: To write data by using this attribute, you must at the same time write data by using the **Password** attribute.
`HomePhone`	Gets or sets the home phone number of the user.	Read, Write
`ImmutableId`	Gets or sets the GUID of the user in Microsoft 365. This GUID is used to verify the identity of the Active Directory user when the user accesses Microsoft 365 by using single sign-on. Note that in order the Microsoft 365 Connector could read the `ImmutableId` attribute value stored in Microsoft 365, that value must be in base64 encoding format. If the `ImmutableId` attribute value has any other encoding format, the Microsoft 365 Connector returns an error when reading that value.	Read, Write
`Initials`	Gets or sets the initials of the user.	Read, Write

Attribute	Description	Supported operations
`LastName`	Gets or sets the last name of the user.	Read, Write
`LiveID`	Gets the user's unique login ID.	Read
`MailboxId`	Gets the GUID of the user's mailbox.	Read
`Manager`	Gets or sets the name of the user's manager.	Read, Write
`MobilePhone`	Gets or sets the user's mobile phone number.	Read, Write
`Name`	Gets or sets the name of the user.	Read, Write
`Notes`	Gets or sets notes about the user.	Read, Write
`ObjectID`	Gets the unique object identifier (GUID).	Read
`Office`	Gets or sets the user's office.	Read, Write
`OtherFax`	Gets or sets the alternate fax number of the user.	Read, Write
`OtherHomePhone`	Gets or sets the alternate home phone number of the user.	Read, Write
`OtherTelephone`	Gets or sets the alternate phone number of the user.	Read, Write
`Pager`	Gets or sets the pager of the user.	Read, Write
`Password`	Sets a password for the user.	Write
`PasswordNeverExpires`	Gets or sets whether or not the user's password periodically expires. This attribute can take one of the following values: • **`TRUE`** (default): Specifies that the user's password never expires. • **`FALSE`**: Specifies that the user's password periodically expires.	Read, Write
`Phone`	Gets or sets the phone number of the user.	Read, Write
`PhoneNumber`	Gets or sets the user's phone number.	Read, Write
`PhoneticDisplayName`	Gets or sets a phonetic pronunciation of the value specified in the `DisplayName` attribute for the user.	Read, Write
`PostalCode`	Gets or sets the user's postal code.	Read, Write
`PostOfficeBox`	Gets or sets the post office box number of	Read, Write

Attribute	Description	Supported operations
	the user.	
`PreferredLanguage`	Gets or sets the preferred language for the user.	Read, Write
`RemotePowerShellEnabled`	Gets or sets whether remote Windows PowerShell cmdlets are available to the user. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`ResetPasswordOnNextLogon`	Gets or sets whether the user must reset their password at next logon. This attribute can take one of the following values: • `TRUE` • `FALSE`	Read, Write
`SimpleDisplayName`	Gets or sets an alternate description of the user in a situation where a limited set of characters is allowed. The limited set of characters includes ASCII characters from 26 to 126.	Read, Write
`State`	Gets or sets the state where the user is located.	Read, Write
`StateOrProvince`	Gets or sets the state or province of the user.	Read, Write
`StreetAddress`	Gets or sets the user's street address.	Read, Write
`Title`	Gets or sets the user's title.	Read, Write
`UMDtmfMap`	Gets or sets whether to create a user-defined DTMF map for the user if it has Unified Messaging enabled.	Read, Write
`UsageLocation`	Gets or sets the two-letter ISO country designation. This attribute specifies the user's country where services are consumed.	Read, Write
`UserPrincipalName`	Gets or sets the user's Microsoft Online Services ID.	Read, Write
`WebPage`	Gets or sets the web page address of the	Read, Write

Attribute	Description	Supported operations
	user.	
`WindowsEmailAddress`	Gets or sets the email address of the user stored in Active Directory.	Read, Write

VoicePolicy object attributes

Table 93: VoicePolicy object attributes

Attribute	Description	Supported operations
`Anchor`	Gets the Anchor property value of the policy.	Read
`Description`	Gets the policy description.	Read
`Identity`	Gets the unique identifier assigned to the policy.	Read
`Members`	Gets the users who have been assigned the policy.	Read
`ObjectID`	Gets the unique object identifier (GUID).	Read

Microsoft 365 group attributes

Table 94: Microsoft 365 group attributes

Attribute	Description	Supported operations
`AcceptMessagesOnlyFromSendersOrMembers`	Gets or sets the senders who can send email messages to the Microsoft 365 group. This attribute can take senders in any of the following formats. For example: • Name • Alias • Distinguished name (DN) • Email address	Read, Write

Attribute	Description	Supported operations
`AccessType`	The `AccessType` parameter specifies the privacy type for the Microsoft 365 group. The acceptable values are: • `Public` • `Private`	Read, Write
`Alias`	Gets or sets the alias of the Microsoft 365 group.	Read, Write
`AlwaysSubscribeMembersToCalendarEvents`	Controls the default subscription settings of the new members that are added to the Microsoft 365 group.	Read, Write
`AuditLogAgeLimit`	Gets or sets the retention period for the mailbox audit logs. Logs whose age exceeds the specified retention period are deleted.	Read, Write
`AutoSubscribeNewMembers`	Specifies if you have to automatically subscribe new members that are added to the Microsoft 365 Group to conversations and calendar events.	Read, Write
`CalendarMemberReadOnly`	Specifies if you have to set read-only Calendar permissions to the Microsoft 365 group for members of the group.	Read
`Classification`	Specifies the classification for the Microsoft 365 Group.	Read

Attribute	Description	Supported operations
CustomAttribute1	Get or set the additional custom values you specify.	Read, Write
CustomAttribute2		
CustomAttribute3		
CustomAttribute4		
CustomAttribute5		
CustomAttribute6		
CustomAttribute7		
CustomAttribute8		
CustomAttribute9		
DataEncryptionPolicy	Specifies the data encryption policy that is applied to the Microsoft 365 group.	Read
DisplayName	Gets or sets the display name of the Microsoft 365 group.	Read, Write
EmailAddresses	Get all the Microsoft 365 proxy addresses of the mailbox. The proxy addresses also include the primary SMTP address.	Read
ExtensionCustomAttribute1	Get or set the additional custom values you specify. These attributes are multivalued.	Read, Write
ExtensionCustomAttribute2		
ExtensionCustomAttribute3		
ExtensionCustomAttribute4		
ExtensionCustomAttribute5		
GrantSendOnBehalfTo	Specifies the sender who can send on behalf of this Microsoft 365 group.	Read, Write
HiddenFromAddressListsEnabled	Gets or sets whether this mailbox is hidden from address lists.	Read, Write
HiddenFromExchangeClientsEnabled	Specifies if the Microsoft	Read, Write

Attribute	Description	Supported operations
	365 Group is hidden from the Outlook clients connected to Microsoft 365.	
`Language`	Gets or sets preferred languages for the Microsoft 365 group.	Read, Write
`MailboxRegion`	This is reserved for internal Microsoft use.	Read
`MailTip`	Gets or sets the message displayed to senders when they start writing an email message to this recipient.	Read
`MailTipTranslations`	Gets or sets the MailTip message translations in additional languages.	Read
`MaxReceiveSize`	Specifies the maximum size of an email message that can be sent to this group	Read, Write
`MaxSendSize`	Specifies the maximum size of an email message that can be sent by this group.	Read, Write
`ModeratedBy`	Gets or sets the users who are moderating the messages sent to the Microsoft 365 group.	Read, Write
`ModerationEnabled`	Gets or sets whether moderation is enabled for the Microsoft 365 group.	Read, Write
`Notes`	Gets or sets notes about the user.	Read, Write
`PrimarySmtpAddress`	Gets or sets primary SMTP address of the Microsoft 365 group.	Read, Write
`RejectMessagesFromSendersOrMembers`	Gets or sets the senders that cannot send email messages to the Microsoft 365 group. The messages sent are rejected.	Read, Write

Attribute	Description	Supported operations
RequireSenderAuthenticationEnabled	Gets or sets if the senders that send messages to this Microsoft 365 group must be authenticated.	Read, Write
SubscriptionEnabled	Specifies if the subscriptions to conversations and calendar events are enabled for the Microsoft 365 group.	Read, Write
UnifiedGroupWelcomeMessageEnabled	Specifies if the option to send the system-generated welcome messages to users who are added as members to the Microsoft 365 group should be enable or disabled.	Read, Write

Objects and attributes specific to Microsoft 365 services

In the Microsoft 365 connection settings, you can select the services you want to work with, such as SharePoint Online, Exchange Online, or Skype for Business Online.

The next table describes the object types and attributes that become available in the Synchronization Service Console user interface when you select a particular check box in the connection settings. The objects and object attributes not mentioned in the table are always available in the Synchronization Service Console user interface.

Table 95: Objects and attributes specific to Microsoft 365 services

Check box	Related objects	Related attributes
SharePoint Online	SPOSiteGroup	All
	SPOWebTemplate	All
	SPOTenant	All
Exchange Online	Contact	All
	DistributionGroup	All
	DynamicDistributionGroup	All
	User	Manager

Check box	Related objects	Related attributes
Skype for Business Online	`ClientPolicy`	All
	`ConferencingPolicy`	All
	`ExternalAccessPolicy`	All
	`HostedVoicemailPolicy`	All
	`VoicePolicy`	All
	`PresencePolicy`	All
	`User`	• `AudioVideoDisabled` • `ClientPolicy` • `ConferencingPolicy` • `Enabled` • `EnterpriseVoiceEnabled` • `ExchangeArchivingPolicy` • `ExternalAccessPolicy` • `HostedVoicemailPolicy` • `LineURI` • `LineServerURI` • `PresencePolicy` • `PrivateLine` • `RegistrarPool` • `RemoteCallControlTelephonyEnabled` • `SipAddress` • `VoicePolicy`

How the Microsoft 365 Connector works with data

To read and write data in Microsoft 365, the **Microsoft 365 Connector** relies on the cmdlets of the ExchangeOnlineManagement Windows PowerShell module. As a result, the connector can only work with data supported by the cmdlets of that module.

Working with Microsoft Azure Active Directory

Synchronization Service reads and writes data in Microsoft Azure Active Directory by using an Azure application in your Microsoft Azure Active Directory environment. To create a connection to Microsoft Azure Active Directory, use the **Microsoft Azure AD Connector** of the Active Roles Synchronization Service.

The Microsoft Azure AD Connector supports the following features:

Table 96: Microsoft Azure AD Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	Yes
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No
Secure Sockets Layer (SSL) data encryption Specifies whether the connector can use SSL to encrypt data transmitted between Active Roles Synchronization Service and the connected data system.	Yes

Creating a Microsoft Azure Active Directory connection

Synchronization Service reads and writes data in Microsoft Azure Active Directory by using an Azure application in your Microsoft Azure Active Directory environment. To create a connection to Microsoft Azure Active Directory, use the **Microsoft Azure AD Connector** of the Active Roles Synchronization Service.

You can create an Azure AD connector by configuring an Azure application in the Synchronization Service Console:

- To create and configure an Azure AD connector manual configuration, see Creating a Microsoft Azure Active Directory connector with manual configuration.

- To create and configure an Azure AD connector with automatic configuration, see Creating a Microsoft Azure Active Directory connector with automatic configuration.
- To configure an Azure application for an Azure AD connector using a script, see Configuring an Azure application for a Microsoft Azure Active Directory connection using a script.

Creating a Microsoft Azure Active Directory connector with manual configuration

Synchronization Service reads and writes data in Microsoft Azure Active Directory by using an Azure application in your Microsoft Azure Active Directory environment. To create a connection to Microsoft Azure Active Directory, use the **Microsoft Azure AD Connector** of the Active Roles Synchronization Service.

You can create an Azure AD connector by configuring an Azure application manually in the Synchronization Service Console. One Identity recommends using **Manual configuration** if you want to use an existing Azure application for the connection.

IMPORTANT: If you are upgrading from an older version of Active Roles Synchronization Service to Active Roles Synchronization Service 8.1.3 or later, and the connector was configured manually, then you must update the authentication data to be able to run a synchronization workflow.

To update the authentication data, you can:

- Use **Auto configuration**. One Identity recommends this approach, as the process is handled automatically by the Active Roles Synchronization Service.

  For more information on automatic configuration, see Modifying the automatic configuration settings of a Microsoft Azure Active Directory connector.
- Enter the **Certificate thumbprint** of the Azure tenant manually, and select the **Tenant Environment Type**.

To create a new Azure AD connector with manual configuration

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Azure AD Connector**.
3. Click **Next**.
4. To use an existing Azure application, select **Manual configuration**.

   NOTE: Alternatively, to use and update an existing Azure application, you can also select **Auto configuration**. Under **Auto configuration**, click **Log in to Azure**, then select the **Tenant environment type** of the Azure tenant. After logging in to Azure with your tenant, the **Tenant ID**, **Application ID**, **Certificate thumbprint** and **Tenant environment type** parameters will be automatically filled in.

5. Enter the **Tenant ID**, **Application ID** and **Certificate thumbprint** of the Azure tenant as they appear on the Azure portal. Then, select the **Tenant Environment Type** of the Azure tenant.
6. To test the connection with the new parameters, click **Test connection**.
7. To finish creating a connection to Azure AD, click **Finish**.

Creating a Microsoft Azure Active Directory connector with automatic configuration

Synchronization Service reads and writes data in Microsoft Azure Active Directory by using an Azure application in your Microsoft Azure Active Directory environment. To create a connection to Microsoft Azure Active Directory, use the **Microsoft Azure AD Connector** of the Active Roles Synchronization Service.

You can create an Azure AD connector by configuring an Azure application automatically in the Synchronization Service Console. One Identity recommends using **Auto configuration** if you want to create a new Azure application for the connection.

To create a new Azure AD connector with automatic configuration

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **Azure AD Connector**.
3. Click **Next**.
4. To create a new Azure application or update an existing one, select **Auto configuration**.

   NOTE: If you have more than one Azure Active Directory (Azure AD) service in your Azure tenant, select **I have more than one Azure AD in my Azure tenant**, and use the **Tenant ID** field to specify the GUID of the Azure AD for which you want to set up synchronization. For more information, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.
5. Select one of the following options based on the number of Azure AD services in your Azure tenant:
   - **I have one Azure AD in my Azure tenant.**
   - **I have more than one Azure AD in my Azure tenant.**
6. Authenticate your access to Azure AD:
   - If you have selected **I have one Azure AD in my Azure tenant**, to authenticate your access to Azure AD, click **Log in to Azure**, and from the **Select Environment Type** drop-down, select the environment type of your Azure tenant.

NOTE: Active Roles Synchronization Service supports Azure Cloud, Azure GCC and Azure GCC-H government tenants.

- If you have selected **I have more than one Azure AD in my Azure tenant**, in **Tenant ID**, enter the GUID of the Azure AD for which you want to set up synchronization.

  TIP: For more information on how to find the GUID of an Azure AD service, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.

  After specifying the tenant ID, to authenticate your access to Azure AD, click **Log in to Azure**, and in the **Select Environment Type** drop-down, select the environment of your Azure tenant.

  NOTE: If you select **I have more than one Azure AD in my Azure tenant**, the **Log in to Azure** button will be enabled only if you specify a well-formed Azure AD GUID in the **Tenant ID** text box.

7. **Azure application name**: Enter the name of the new or existing Azure application.
8. To create or update the Azure application in Azure AD, click **Create or update Azure application**.

   The created or updated Azure application has the following directory roles assigned to it:

   - Directory Writers
   - Exchange Administrator
   - User Administrator

   The following permissions are also added, for which you must give admin consent:

   - Sign in and read user profile
   - Manage Exchange As Application

   NOTE: You may need to set additional permissions depending on your needs.
9. To give admin consent for the permissions of the Azure application, click **Consent**. Then, in the **Azure Tenant Consent** dialog, click **Accept**.
10. To test the connection with the new parameters, click **Test connection**.
11. To finish creating a connection to Azure AD, click **Finish**.

Configuring an Azure application for a Microsoft Azure Active Directory connection using a script

Synchronization Service reads and writes data in Microsoft Azure Active Directory by using an Azure application in your Microsoft Azure Active Directory environment.

To create an Azure AD connection by configuring an Azure application using a Windows PowerShell script, perform the following steps.

To configure an Azure application for an Azure Active Directory connection using a Windows PowerShell script

1. Create an application in any domain of your Microsoft Azure Active Directory environment. The application must have sufficient permissions to read and write data in Microsoft Azure Active Directory.

   You can assign the required permissions to the application by running a Windows PowerShell script. To run the script, you need to install Microsoft Azure PowerShell on your computer.

   Script example

   ```
   # Replace <ClientId> with the Client ID of the Active Roles Azure AD Connector Application (example format: 455ad643-332g-32h7-q004-8ba89ce65ae26)

   $Id = “<ClientId>”

   # Prompt for Microsoft Azure AD Global Admin credentials.

   Connect-AzureAD

   # Get the Principal ID of the Active Roles Azure AD Connector Application and save it to the $servicePrincipal variable

   $servicePrincipal = Get-AzureADServicePrincipal -All $true | Where-Object {$_.AppId -eq $Id}

   # Get the required role ID from the Active Roles Azure AD Connector Application and save it to the $roleId variable

   $roleId = (Get-AzureADDirectoryRole | Where-Object {$_.displayName -eq 'Company Administrator'}).ObjectId

   # Assign the required permissions to the Active Roles Azure AD Connector Application

   Add-AzureADDirectoryRoleMember -ObjectId $roleId -RefObjectId $servicePrincipal.ObjectId
   ```

2. Open the application properties and copy the following information:
   - Tenant ID
   - Application ID

- Certificate thumbprint

3. When creating a new Microsoft Azure Active Directory connection or modifying an existing one in the Synchronization Service Console, enter the **Tenant ID**, **Application ID**, and **Certificate thumbprint** of the Azure tenant as they appear on the Azure portal. For more information, see Creating a Microsoft Azure Active Directory connection.

Modifying a Microsoft Azure Active Directory connection

Synchronization Service reads and writes data in Microsoft Azure Active Directory by using an Azure application in your Microsoft Azure Active Directory environment. To create a connection to Microsoft Azure Active Directory, use the **Microsoft Azure AD Connector** of the Active Roles Synchronization Service.

You can modify the settings of an existing Azure AD connector in the Synchronization Service Console.

- To modify the manually configured settings of an Azure AD connector, see Modifying the manual configuration settings of a Microsoft Azure Active Directory connector.
- To modify the automatically configured settings of an Azure AD connector, see Modifying the automatic configuration settings of a Microsoft Azure Active Directory connector.

Modifying the manual configuration settings of a Microsoft Azure Active Directory connector

You can modify the manual configuration settings of an existing Azure AD connector in the Synchronization Service Console.

IMPORTANT: If you are upgrading from an older version of Active Roles Synchronization Service to Active Roles Synchronization Service 8.1.3 or later, and the connector was configured manually, then you must update the authentication data to be able to run a synchronization workflow.

To update the authentication data, you can:

- Use **Auto configuration**. One Identity recommends this approach, as the process is handled automatically by the Active Roles Synchronization Service.

  For more information on automatic configuration, see Modifying the automatic configuration settings of a Microsoft Azure Active Directory connector.
- Enter the **Certificate thumbprint** of the Azure tenant manually, and select the **Tenant Environment Type**.

To modify the manual configuration settings of a Azure AD connector

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** under the existing Azure AD connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
4. To use an existing Azure application, select **Manual configuration**.

   NOTE: Alternatively, to use and update an existing Azure application, you can also select **Auto configuration**. Under **Auto configuration**, click **Log in to Azure**, then select the **Tenant environment type** of the Azure tenant. After logging in to Azure with your tenant, the **Tenant ID**, **Application ID**, **Certificate thumbprint** and **Tenant environment type** parameters will be automatically filled in.
5. Enter the **Tenant ID**, **Application ID** and **Certificate thumbprint** of the Azure tenant as they appear on the Azure portal. Then, select the **Tenant Environment Type** of the Azure tenant.
6. To test the connection with the new parameters, click **Test connection**.
7. To modify the connection settings, click **Save**.

Modifying the automatic configuration settings of a Microsoft Azure Active Directory connector

You can modify the automatic configuration settings of an existing Azure AD connector in the Synchronization Service Console.

To modify the automatic configuration settings of an Azure AD connector

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** under the existing Azure AD connection you want to modify.
3. On the **Connection Settings** tab, click **Specify connection settings** to expand it and use the following options.
4. To create a new Azure application or update an existing one, select **Auto configuration**.

   NOTE: If you have more than one Azure Active Directory (Azure AD) service in your Azure tenant, select **I have more than one Azure AD in my Azure tenant**, and use the **Tenant ID** field to specify the GUID of the Azure AD for which you want to set up synchronization. For more information, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.
5. Select one of the following options based on the number of Azure AD services in your Azure tenant:

- **I have one Azure AD in my Azure tenant.**
- **I have more than one Azure AD in my Azure tenant.**

6. Authenticate your access to Azure AD:
   - If you have selected **I have one Azure AD in my Azure tenant**, to authenticate your access to Azure AD, click **Log in to Azure**, and from the **Select Environment Type** drop-down, select the environment type of your Azure tenant.

     NOTE: Active Roles Synchronization Service supports Azure Cloud, Azure GCC and Azure GCC-H government tenants.
   - If you have selected **I have more than one Azure AD in my Azure tenant**, in **Tenant ID**, enter the GUID of the Azure AD for which you want to set up synchronization.

     TIP: For more information on how to find the GUID of an Azure AD service, see Finding the GUID (Tenant ID) of an Azure AD for Azure BackSync.

     After specifying the tenant ID, to authenticate your access to Azure AD, click **Log in to Azure**, and in the **Select Environment Type** drop-down, select the environment of your Azure tenant.

     NOTE: If you select **I have more than one Azure AD in my Azure tenant**, the **Log in to Azure** button will be enabled only if you specify a well-formed Azure AD GUID in the **Tenant ID** text box.
7. **Azure application name**: Enter the name of the new or existing Azure application.
8. To create or update the Azure application in Azure AD, click **Create or update Azure application**.

   The created or updated Azure application has the following directory roles assigned to it:
   - Directory Writers
   - Exchange Administrator
   - User Administrator

   The following permissions are also added, for which you must give admin consent:
   - Sign in and read user profile
   - Manage Exchange As Application

   NOTE: You may need to set additional permissions depending on your needs.
9. To give admin consent for the permissions of the Azure application, click **Consent**. Then, in the **Azure Tenant Consent** dialog, click **Accept**.
10. To test the connection with the new parameters, click **Test connection**.
11. To modify the connection settings, click **Save**.

Microsoft Azure Active Directory data supported for synchronization

The next table lists the Microsoft Azure Active Directory object types supported by the Microsoft Azure AD Connector. The table also provides information about the operations you can perform on these objects by using the Microsoft Azure AD Connector.

Table 97: Supported objects and operations

Object	Read	Create	Delete	Update
User	Yes	Yes	Yes	Yes
Group	Yes	Yes	Yes	Yes

The following sections describe the attributes provided by the Microsoft Azure AD Connector. By using these attributes, you can read and/or write data related to a particular object in Microsoft Azure Active Directory.

Microsoft Azure AD user attributes supported for data synchronization

The **Microsoft Azure AD Connector** of the Active Roles Synchronization Service supports the following Azure Active Directory (Azure AD) user attributes for data synchronization.

NOTE: When configuring a data synchronization mapping rule with the **Microsoft Azure AD Connector**, consider that the following user attributes are currently not supported and cannot be queried via the Microsoft Graph API:

- `aboutMe`
- `birthday`
- `contacts`
- `hireDate`
- `interests`
- `mySite`
- `officeLocation`
- `pastProjects`
- `preferredName`
- `responsibilites`
- `schools`
- `skills`

This means that although these user attributes are visible, they cannot be set in a mapping rule.

Table 98: Azure AD user attributes supported for data synchronization

Attribute	Description	Supported operations
accountEnabled	Gets or sets whether the user account is enabled. NOTE: This attribute is required when creating a user.	Read, Write
city	Gets or sets the user city.	Read, Write
country	Gets or sets the user country.	Read, Write
department	Gets or sets the user department.	Read, Write
dirSyncEnabled	Gets or sets whether the user was synchronized from the on-premises Active Directory Domain Services (AD DS).	Read, Write
directReports	Gets the direct reports of the user.	Read
displayName	Gets or sets the user name in the address book. NOTE: This attribute is required when creating a user.	Read, Write
facsimileTelephoneNumber	Gets or sets the user fax number.	Read, Write
givenName	Gets or sets the given name of the user.	Read, Write
jobTitle	Gets or sets the user job title.	Read, Write
lastDirSyncTime	Gets the time when the user was last synchronized with the on-premises AD DS.	Read
mail	Gets or sets the primary e-mail address of the user.	Read, Write
mailNickName	Gets or sets the mail alias of the user. NOTE: This attribute is required when creating a user.	Read, Write
manager	Gets or sets the manager of the user.	Read, Write
memberOf	Gets the group membership of the user.	Read
mobile	Gets or sets the mobile phone number o the user.	Read, Write
objectId	Gets the unique identifier of the user.	Read

Attribute	Description	Supported operations
`objectType`	Gets the object type of the user.	Read
`otherMails`	Gets or sets other e-mail addresses for the user.	Read, Write
`passwordPolicies`	Gets or sets password policies applicable to the user.	Read, Write
`passwordProfile`	Gets or sets the password profile of the user. NOTE: This attribute is required when creating a user.	Read, Write
`physicalDeliveryOfficeName`	Gets or sets the office location of the user.	Read, Write
`postalCode`	Gets or sets the postal code of the user.	Read, Write
`preferredLanguage`	Gets or sets the preferred language of the user.	Read, Write
`provisionedPlans`	Gets the provisioned plans of the user.	Read
`provisioningErrors`	Gets the errors encountered when provisioning the user.	Read
`proxyAddresses`	Gets the known address entries of the user.	Read
`state`	Gets or sets the state or province of the user.	Read, Write
`streetAddress`	Gets or sets the street address of the user.	Read, Write
`surname`	Gets or sets the family name of the user.	Read, Write
`telephoneNumber`	Gets or sets the telephone number of the user.	Read, Write
`thumbnailPhoto`	Gets or sets the thumbnail photo of the user.	Read, Write
`usageLocation`	Gets or sets the usage location, that is the geographical location where the user is located and operating from.	Read, Write
`userPrincipalName`	Gets or sets the user principal name of the user. NOTE: This attribute is required when creating a user.	Read, Write

Microsoft Azure AD group attributes supported for data synchronization

The **Microsoft Azure AD Connector** of the Active Roles Synchronization Service supports the following Azure Active Directory (Azure AD) group attributes for data synchronization.

NOTE: When configuring a data synchronization mapping rule with the **Microsoft Azure AD Connector**, consider that the following group attributes are currently not supported and cannot be queried via the Microsoft Graph API:

- `acceptedSenders`
- `allowExternalSenders`
- `autoSubscribeNewMembers`
- `hasMembersWithLicenseErrors`
- `hideFromAddressLists`
- `hideFromOutlookClients`
- `isSubscribedByMail`
- `membersWithLicenseErrors`
- `rejectedSenders`
- `unseenCount`

This means that although these group attributes are visible, they cannot be set in a mapping rule.

Table 99: Azure AD group attributes supported for data synchronization

Attribute	Description	Supported operations
`description`	Gets or sets the group description.	Read, Write
`dirSyncEnabled`	Gets whether the group was synchronized from the on-premises Active Directory Domain Services (AD DS).	Read
`displayName`	Gets or sets the display name of the group. NOTE: This attribute is required when creating a group.	Read, Write
`lastDirSyncTime`	Gets the time when the group was last synchronized with the on-premises AD DS.	Read
`mail`	Gets or sets the e-mail address of the group.	Read, Write
`mailEnabled`	Gets or sets whether the group is mail-enabled. NOTE: This attribute is required when creating a group.	Read, Write

Attribute	Description	Supported operations
`mailNickName`	Gets or sets the mail alias of the group. NOTE: This attribute is required when creating a group.	Read, Write
`members`	Gets or sets the members of the group.	Read, Write
`objectId`	Gets the unique identifier of the group.	Read
`objectType`	Gets the object type of the group.	Read
`provisioningErrors`	Gets the errors encountered when provisioning the group.	Read
`proxyAddresses`	Gets the known address entries of the group.	Read
`securityEnabled`	Gets or sets whether the group is a security group. NOTE: This attribute is required when creating a group.	Read, Write

Configuring data synchronization with the SCIM Connector

With the **SCIM Connector**, you can configure inbound data synchronization connections for the following SCIM-based One Identity Starling Connect connectors:

- PingOne
- Workday HR

NOTE: Consider the following when planning to configure a connection with the **SCIM Connector**:

- The **SCIM Connector** is tested to support the Starling Connect PingOne and Workday HR connectors. To configure a connection for import-based workflows to the SCIM 2.0-based SuccessFactors HR 8.0 or ServiceNow 2.0 Starling connectors, use the **Generic SCIM Connector** instead. For more information, see Configuring data synchronization with the Generic SCIM Connector.
- The **SCIM Connector** supports only the standard schema of the SCIM protocol. It does not support extended schemas, and therefore cannot handle user-made custom attributes.

For the list of Active Roles Synchronization Service connector features that the **SCIM Connector** supports or does not support, see the following table.

Table 100: SCIM Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	No
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No
Secure Sockets Layer (SSL) data encryption Specifies whether the connector can use SSL to encrypt data transmitted between Active Roles Synchronization Service and the connected data system.	Yes

For more information on the SCIM protocol, see the official SCIM site, or the following IETF RFC documents:

- IETF RFC-7642: *System for Cross-domain Identity Management: Definitions, Overview, Concepts, and Requirements*
- IETF RFC-7643: *System for Cross-domain Identity Management: Core Schema*
- IETF RFC-7644: *System for Cross-domain Identity Management: Protocol*

Objects and operations supported by the SCIM Connector

This section lists the data objects supported by the SCIM Connector, along with the data operations you can perform on those objects via the SCIM Connector.

Table 101: Supported objects and operations for SCIM v2.0

Object	Read	Create	Delete	Update
Core user	Yes	Yes	Yes	Yes
Group	Yes	Yes	Yes	Yes
Enterprise	Yes	Yes	Yes	Yes

Table 102: Supported objects and operations for SCIM v1.1

Object	Read	Create	Delete	Update
User	Yes	Yes	Yes	Yes
Group	Yes	Yes	Yes	Yes

Creating a SCIM connection with the SCIM Connector

You can configure an Active Roles Synchronization Service connection to the PingOne and Workday HR connectors of Starling Connect with the **SCIM Connector**.

To configure a connection to a Starling Connect connector with the SCIM Connector

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **SCIM Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, under **SCIM settings**, configure the following connection options:
   - **SCIM version**: Select the SCIM protocol version to use for the connection. The SCIM Connector supports protocol versions **V2** and **V1.1**.
   - **SCIM URL**: Specify the the base URL of the Starling Connect connector to which you want to connect.
   - **Authentication type**: Select the authentication type. The SCIM Connector supports **Basic**, **OAuth** and **API Key**-based authentication. The contents of the **Authentication parameters** section are populated dynamically based on the authentication type you select in this setting.
5. Under **Authentication parameters**, configure the applicable options:
   - If you use **Basic** authentication, provide a valid **User name** and **Password**.

     NOTE: The PingOne connector of Starling uses the API key as the **User name** and the API token as the **Password**.
   - If you use **OAuth** authentication, configure the settings applicable to the selected **Grant type**.

Table 103: SCIM Connector OAuth authentication settings

Grant type	Setting	Description
password	**Token URL**	Specify the URL of the token.
	User name	Specify the user name.
	Password	Specify the password.
	Client ID	Specify the client ID used for login.
	Client secret	Specify the client secret.
client_ credentials	**Token URL**	Specify the URL of the token.
	Client ID	Specify the client ID used for login.
	Client secret	Specify the client secret.
Bearer_ Token	**Bearer token**	Specify the bearer token for the connection. NOTE: Connections using a bearer token have a time-limit, specified by the token provider. Once this time limit is reached, the connection ends. To establish a new connection session, you must create a new bearer token.

- If you use **API Key** authentication, specify the API **Key** and **Token** for the connection.

6. (Optional) To connect to the Workday HR connector of Starling with the configured SCIM Connector, select **Load workday schema**. Selecting this option will result in the configured SCIM Connector using the Workday schema instead of the standard SCIM schema.

   NOTE: Select **Load workday schema** only if you want to connect to the Workday HR connector of Starling. Attempting to connect to the PingOne connector with this setting enabled will result in the SCIM Connector failing to synchronize data.

7. (Optional) To configure additional authentication parameters (such as a region ID or organization ID) for the SCIM Connector, click **Add additional parameters**. Then, use the **Additional authentication parameters** settings to specify additional **Plain text parameters** or **Masked parameters**. To save the parameters, click **OK**.
8. To verify that the specified settings are correct, click **Test Connection**.
9. To create the connection, click **Finish**.

Viewing or modifying the settings of a SCIM Connector

You can view or modify an existing connection based on the **SCIM Connector** with Active Roles Synchronization Service. Modifying a **SCIM Connector** is typically required if any change occurs in the SCIM-based Starling Connect connectors to which the Active Roles Synchronization Service connection was originally configured.

To view or modify an existing SCIM Connector connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Connection settings** below the existing SCIM connection you want to modify.
3. On the **Connection Settings** tab, click the **Specify connection settings** item to expand it and update the required settings.

   For more information on the available settings, see Creating a SCIM connection with the SCIM Connector.
4. To apply your changes, click **Save**.

Configuring data synchronization with the Generic SCIM Connector

With the **Generic SCIM Connector**, you can configure inbound data synchronization connections for a set of SCIM 2.0-based One Identity Starling Connect connectors:

NOTE: Consider the following when planning to configure a SCIM-based data synchronization connector:

- The **Generic SCIM Connector** was tested with the following Starling Connect connectors:
  - Pipedrive 1.0
  - ServiceNow 2.0
  - SuccessFactors HR 8.0 and 9.0
  - WorkdayHR 3.0
  - Zendesk 1.0

  While the **Generic SCIM Connector** may work with other SCIM 2.0-based Starling Connect connectors, One Identity tested it to work only with those connectors and connector versions.
- To configure a connection to the PingOne connector of Starling Connect, use the **SCIM Connector** of Active Roles Synchronization Service. For more information, see Configuring data synchronization with the SCIM Connector.

For the list of Active Roles Synchronization Service connector features that the **Generic SCIM Connector** supports or does not support, see the following table.

Table 104: Generic SCIM Connector – Supported features

Feature	Supported
Bidirectional synchronization Specifies whether you can both read and write data in the connected data system.	No
Delta processing mode Specifies whether the connection can process only the data that has changed in the connected data system since the last synchronization operation. This reduces the overall synchronization duration.	No
Password synchronization Specifies whether you can synchronize user passwords from an Active Directory (AD) domain to the connected data system.	No
Secure Sockets Layer (SSL) data encryption Specifies whether the connector can use SSL to encrypt data transmitted between Active Roles Synchronization Service and the connected data system.	Yes

For more information on the SCIM protocol, see the official SCIM site, or the following IETF RFC documents:

- IETF RFC-7642: *System for Cross-domain Identity Management: Definitions, Overview, Concepts, and Requirements*
- IETF RFC-7643: *System for Cross-domain Identity Management: Core Schema*
- IETF RFC-7644: *System for Cross-domain Identity Management: Protocol*

Configuring the Generic SCIM Connector for Starling Connect connections

You can configure an Active Roles Synchronization Service connection to the following Starling Connect connectors with the **Generic SCIM Connector**:

- Pipedrive 1.0
- ServiceNow 2.0
- SuccessFactors HR 8.0 and 9.0
- WorkdayHR 3.0
- Zendesk 1.0

Prerequisites

Before configuring the connection, make sure that the following conditions are met:

- Your organization must have an active Starling Connect account.
- The Starling connector to which you want to connect must be already configured in Starling Connect.
- If your organization is using a proxy server for outbound connections, make sure that the system level proxy settings are properly configured.

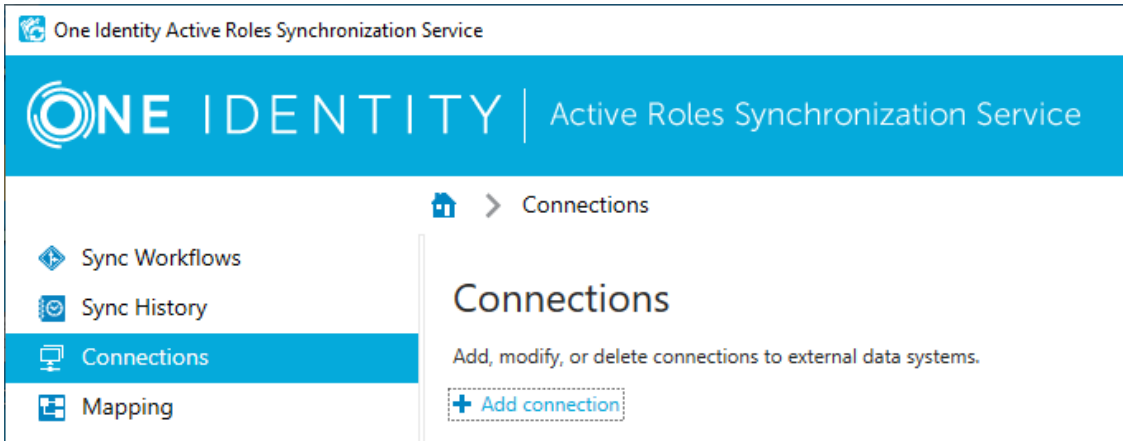
  To configure system-level proxy settings, navigate to one of the following Windows configuration pages:

  - **Control Panel** > **Internet Settings** > **Connections** > **LAN Settings**
  - **Settings** > **Network and Internet** > **Proxy**
- You are aware of the specific implementation details (such as the supported objects and operations) of the Starling Connect connector you want to connect to. For more information, see the connector-specific sections of the *Starling Connect Active Roles Synchronization Service Administration Guide*.

To configure a connection to a Starling Connect connector with the Generic SCIM Connector

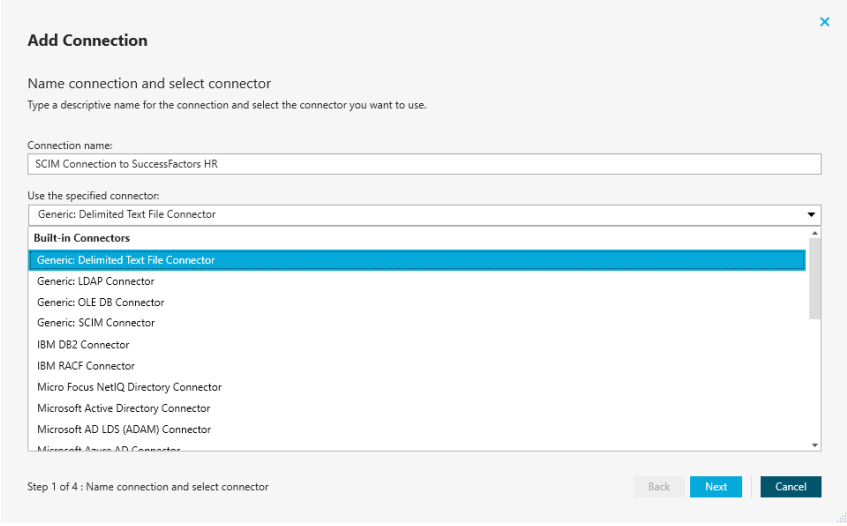
1. In the Active Roles Synchronization Service, navigate to **Connections** > **Add Connection**.

   Figure 4: Active Roles Synchronization Service Console – Adding a new connection via Connections > Add connection

2. In the **Name connection and select connector** step, specify a custom **Connection name**. Then, to load the SCIM-specific connector settings, from the **Use the specified connector** drop-down list, select **Generic: SCIM Connector**.

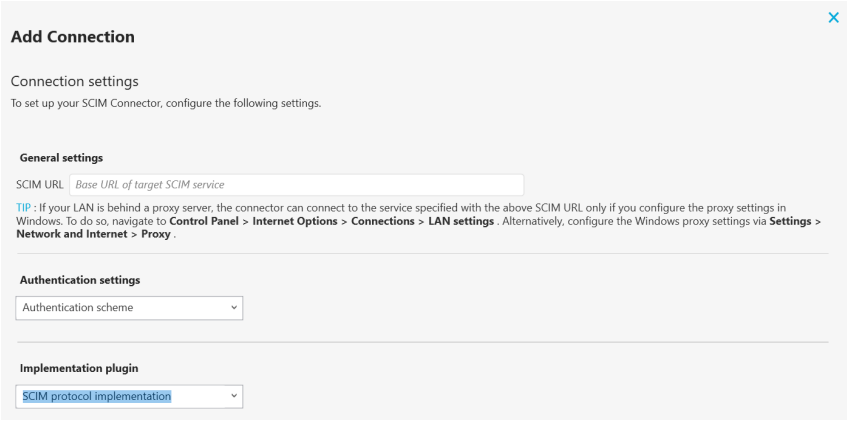
Figure 5: Add Connection – Specifying the connection name and connector type



3. (Optional) If you want to use a remote connector for the configured connection, configure **Remote connector access** as described in Creating a connection using a remotely installed connector. To continue, click **Next**.
4. To continue, click **Next**.

   The **Connection settings** step of the **Generic SCIM Connector** appears.

Figure 6: Generic SCIM Connector – General, authentication and implementation settings



5. Under **General settings**, specify the base **SCIM URL** of the Starling Connect connector to which you want to connect.

   > TIP: To check the base SCIM URL of the Starling Connect connector, in Starling Connect, navigate to **Connectors** > **Active Connectors**, select the SCIM-based connector to which you want to connect, then copy the value of the **SCIM URL** property.

6. Under **Authentication settings**, to enable the authentication scheme options required by the supported Starling Connect connectors, select the **Starling** authentication scheme, then configure the following settings:
   - **Token endpoint URL**: Specifies the full path of the Starling connector token endpoint.

     TIP: To find the token endpoint URL of the Starling Connect connector, in Starling Connect, navigate to **Connectors** > **Active Connectors**, and copy the value of the **SCIM Token Endpoint URL** property.
   - **Client ID**: Specifies the SCIM client ID.

     TIP: To find the SCIM client ID of the Starling Connect connector, in Starling Connect, navigate to **Connectors** > **Active Connectors**, and copy the value of the **SCIM Client ID** property.
   - **Client secret**: Specifies the SCIM client secret.

     TIP: To find the SCIM client secret of the Starling Connect connector, in Starling Connect, navigate to **Connectors** > **Active Connectors**, and copy the value of the **Show SCIM Client Secret** text box.
7. Under **Implementation plugin**, to enable the pre-made connection implementation for the supported Starling Connect connectors, select **Starling batch 1 - v2.0**.

   NOTE: The **Starling batch 1 - v2.0** implementation plugin is backwards compatible with **Starling batch 1 - v1.0**, originally released in Active Roles Synchronization Service 7.6.1.

   The **Generic SCIM Connector** was tested with the following Starling Connect connectors:
   - Pipedrive 1.0
   - ServiceNow 2.0
   - SuccessFactors HR 8.0 and 9.0
   - WorkdayHR 3.0
   - Zendesk 1.0

   While the **Generic SCIM Connector** may work with other SCIM 2.0-based Starling Connect connectors, One Identity tested it to work only with those connectors and connector versions.

   For the list of SCIM attributes supported by Starling Connect for these connectors, see the *Pipedrive*, *ServiceNow*, *SuccessFactorsHR* and *Zendesk* chapters of the *Starling Connect Active Roles Administration Guide*.
8. Configure the following Starling Connect connection settings as required by the Starling Connect connector to which you want to connect.
   - **Import uses direct query**: When selected, Active Roles Synchronization Service queries every synchronized object separately by their ID. Select this setting when configuring a connection to the Starling Connect ServiceNow 2.0, Zendesk 1.0, or similar connectors.

NOTE: Consider the following when using this setting:

- Selecting this setting decreases synchronization speed considerably. However, you must select this setting to read all object attributes for Starling Connect ServiceNow 2.0, or to read certain resource types or attributes for Zendesk 1.0.
- Do not enable this setting when configuring the **Generic SCIM Connector** for other supported Starling Connect connectors, as it has no effect on the results of import data synchronization.

- **Query only synced attributes**: To improve performance, certain Starling Connect connectors allow to query only parameters that are specifically defined for synchronization. If you enable this setting, Synchronization Service sets the `?attributes=attrName` query parameter according to IETF RFC-7644, so that Starling Connect will retrieve the attributes specified in the sync workflow.

  NOTE: Select this setting if you configure a connection for the Starling Connect Pipedrive 1.0 or Zendesk 1.0 connectors.

- **Starling cursor-based pagination**: Certain Starling Connect connectors use a cursor-based pagination method (as defined by Cursor-based Pagination of SCIM Resources) instead of the protocol-defined index-based pagination. When configuring a connection to such a Starling Connector, select this setting to override the standard pagination method.

  NOTE: Select this setting if you configure a connection to the Starling Connect Pipedrive 1.0, WorkdayHR 3.0 or Zendesk 1.0 connectors.

- **Max degree of parallelism**: If **Import uses direct query** is enabled, this setting specifies the maximum number of threads that Synchronization Service Console can run in parallel for the direct query of each object in the response list (that is, how many entries can Synchronization Service Console query simultaneously).

  TIP: One Identity recommends testing the value optimal for your environment, and setting it as low as possible. Specifying a value of 1 means no parallelism is configured.

  NOTE: Consider the following when using this setting:

  - This setting works only if **Import uses direct query** is enabled. Active Roles Synchronization Service will ignore any value specified for **Max degree of parallelism** if **Import uses direct query** is not selected.
  - Setting the value of **Max degree of parallelism** too high may result in connector service instability.

9. Check the implementation plugin information indicated on-screen. Make sure that the **Supported Features**, the **Target Service Providers** and the supported Starling Connect connector versions will meet the requirements of your planned mapping rule and/or synchronization workflow.

10. To verify that the specified authentication settings are correct, click **Test Connection**.

    NOTE: Clicking **Test Connection** verifies only if the authentication settings for the SCIM metadata endpoint connection are correct, and if Active Roles Synchronization Service can fetch the SCIM schemas and query the `resourceTypes` metadata from the configured SCIM service.

    When testing the connection, Active Roles Synchronization Service does not query any actual resource objects. Because of this, testing may finish successfully even if the connection is down between Starling Connect and the third-party service provider (for example, SuccessFactors HR), preventing the import of actual data during synchronization later.

    TIP: If testing fails, Active Roles Synchronization Service will highlight the settings that it detects as incorrect. Check and fix those settings, then try again. If testing fails again, then:

    - Check your network connectivity.
    - Check if the Starling Connect service is available.
    - Make sure that the Starling Connect connector you specified during configuration is still active and working.
    - If you use a proxy server, make sure that the system-level proxy settings are properly configured.

11. If testing completed successfully, create the new SCIM connection to the Starling Connect connector by clicking **Finish**.

After Active Roles Synchronization Service created the connection, you can use it to configure SCIM-based data synchronization by setting up one or more mapping rules and synchronization workflows.

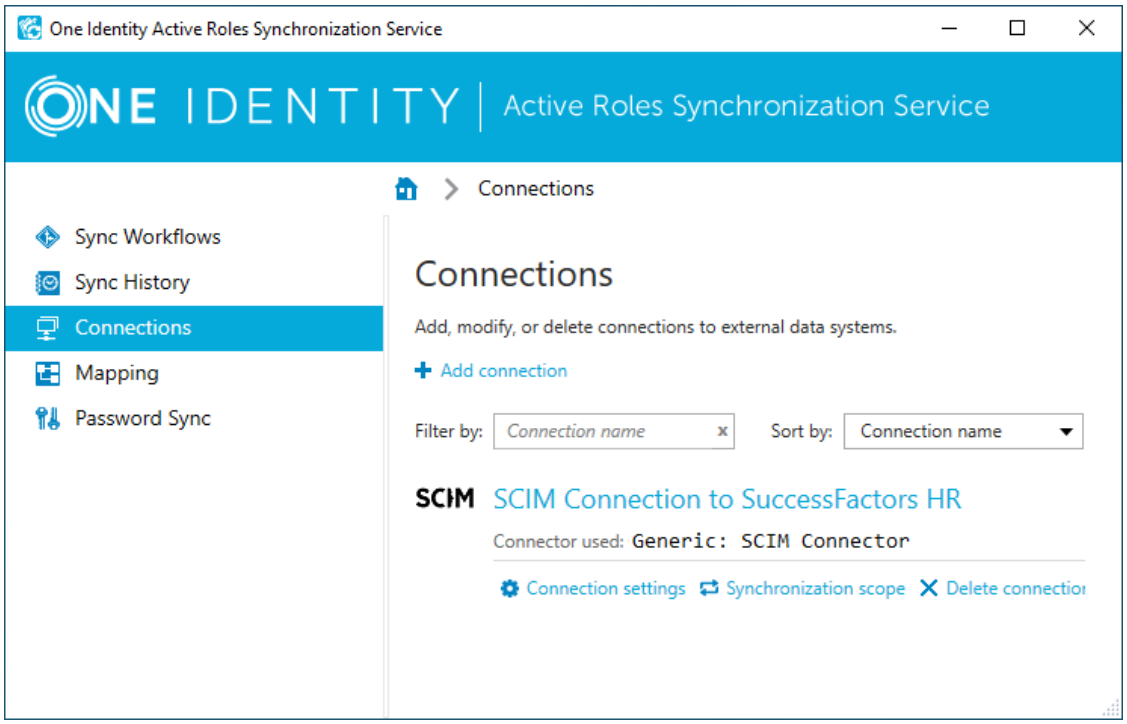
- For an example SCIM-based mapping rule, see Creating object mapping between a SCIM connection and an SQL connection.
- For an example SCIM-based synchronization workflow, see Creating a sync workflow for synchronizing data from a SCIM-based Starling Connect connector.
- For a PowerShell script example for synchronizing complex multi-value objects from a SCIM source system, see Synchronizing complex multi-value objects from a SCIM source system.

Viewing or modifying the settings of a Generic SCIM Connector connection

You can view or modify an existing connection based on the **Generic SCIM Connector** with the Synchronization Service Console. Modifying a **Generic SCIM Connector** connection is typically required if any change occurs in the SCIM-based Starling Connect connectors to which the Active Roles Synchronization Service connection was originally configured.

To view or modify an existing Generic SCIM Connector connection

1. In the Synchronization Service Console, click **Connections**.
2. In **Connections**, search for the connection you want to modify, then click **Connection settings**.



3. (Optional) In **General**, modify the custom **Connection name**.
4. (Optional) In **Connection Settings**, modify the following settings as you need:
   - **Token endpoint URL**: Specifies the full path of the Starling connector token endpoint.

     TIP: To find the token endpoint URL of the Starling Connect connector, in Starling Connect, navigate to **Connectors** > **Active Connectors**, and copy the value of the **SCIM Token Endpoint URL** property.
   - **Client ID**: Specifies the SCIM client ID.

     TIP: To find the SCIM client ID of the Starling Connect connector, in Starling Connect, navigate to **Connectors** > **Active Connectors**, and copy the value of the **SCIM Client ID** property.
   - **Client secret**: Specifies the SCIM client secret.

     TIP: To find the SCIM client secret of the Starling Connect connector, in Starling Connect, navigate to **Connectors** > **Active Connectors**, and copy the value of the **Show SCIM Client Secret** text box.
   - **Import uses direct query**: When selected, Active Roles Synchronization Service queries every synchronized object separately by their ID. Select this

setting when configuring a connection to the Starling Connect ServiceNow 2.0, Zendesk 1.0, or similar connectors.

> NOTE: Consider the following when using this setting:
>
> - Selecting this setting decreases synchronization speed considerably. However, you must select this setting to read all object attributes for Starling Connect ServiceNow 2.0, or to read certain resource types or attributes for Zendesk 1.0.
> - Do not enable this setting when configuring the **Generic SCIM Connector** for other supported Starling Connect connectors, as it has no effect on the results of import data synchronization.

- **Query only synced attributes**: To improve performance, certain Starling Connect connectors allow to query only parameters that are specifically defined for synchronization. If you enable this setting, Synchronization Service sets the `?attributes=attrName` query parameter according to IETF RFC-7644, so that Starling Connect will retrieve the attributes specified in the sync workflow.

  > NOTE: Select this setting if you configure a connection for the Starling Connect Pipedrive 1.0 or Zendesk 1.0 connectors.

- **Starling cursor-based pagination**: Certain Starling Connect connectors use a cursor-based pagination method (as defined by Cursor-based Pagination of SCIM Resources) instead of the protocol-defined index-based pagination. When configuring a connection to such a Starling Connector, select this setting to override the standard pagination method.

  > NOTE: Select this setting if you configure a connection to the Starling Connect Pipedrive 1.0, WorkdayHR 3.0 or Zendesk 1.0 connectors.

- **Max degree of parallelism**: If **Import uses direct query** is enabled, this setting specifies the maximum number of threads that Synchronization Service Console can run in parallel for the direct query of each object in the response list (that is, how many entries can Synchronization Service Console query simultaneously).

  > TIP: One Identity recommends testing the value optimal for your environment, and setting it as low as possible. Specifying a value of 1 means no parallelism is configured.

  > NOTE: Consider the following when using this setting:
  >
  > - This setting works only if **Import uses direct query** is enabled. Active Roles Synchronization Service will ignore any value specified for **Max degree of parallelism** if **Import uses direct query** is not selected.
  > - Setting the value of **Max degree of parallelism** too high may result in connector service instability.

5. (Optional) In **Scope**, modify the scope of objects included in the data synchronization process of the connection. For more information on the **Scope** settings, see Modifying synchronization scope for a connection.

6. (Optional) In **Connection Handlers**, create, update or remove any automated data synchronization operations for the connection. For more information on the **Connection Handlers** settings, see Using connection handlers.
7. To apply your changes, click **Save and Continue**.

Using connectors installed remotely

In some cases, you need to configure a connection to an external data system which is separated by a firewall from the computer running Synchronization Service. To implement this scenario, you can install an instance of Synchronization Service and built-in connectors on a remote computer and switch this Synchronization Service instance to remote mode. This will allow the Synchronization Service instance running in the local mode to communicate with the remotely installed instance and connectors via a single port.

Consider a scenario where you want to synchronize data between two Active Directory domains that are separated by a firewall. In this case, you can install one Synchronization Service instance in the local mode in the first domain, then deploy another Synchronization Service instance in the remote mode in the other domain. Then, ensure the firewall allows traffic on the port used for communications between the Synchronization Service instances.

Installing Synchronization Service and built-in connectors remotely

To use connectors remotely, you need to install Synchronization Service and built-in connectors on a required remote computer and switch the installed instance of Synchronization Service to remote mode. For more information on installing Synchronization Service, see Installing Synchronization Service.

To set Synchronization Service in remote mode

1. Start the Synchronization Service Console.
2. Follow the steps in the wizard that starts automatically to configure Synchronization Service.
3. On the **Service Account and Mode** page, do the following and click **Finish**:
    - Enter the account under which you want Synchronization Service to run.
    - Select the remote mode for this instance of Synchronization Service.

Creating a connection using a remotely installed connector

To create a connection using a remotely installed connector

1. Start the Synchronization Service Console.
2. On the **Connections** tab, click **Add connection**.
3. In the **Connection name** text box, type a descriptive name for the connection.
4. From the **Use the specified connector list**, select the connector you want to use.
5. Click to expand the **Remote connector access** element, and then use the following options:
    - **Use remote connector**: Select this check box to use the connector installed on a remote computer.
    - **Connector host**: Type the Fully Qualified Domain Name (FQDN) of the computer on which the Synchronization Service in the remote mode and the corresponding connector are installed.
    - **Port**: Type the port number on which you want the Synchronization Service to access the remote connector. By default, this is port **`8080`**.
    - **Connect using**: Specify an account under which to access the remote connector. The account must be a local administrator on the computer where the remote connector is installed. Select one of the following:
        - **Synchronization Service account**: Allows you to access the remote connector using the account under which Synchronization Service is running locally.
        - **Windows account**: Allows you to type the user name and password of the account with which you want to access the remote connector.
    - **Verify Settings**: Click this button to verify that Synchronization Service can access the remote connector using the settings you have specified.
6. Follow the instructions of the wizard to complete the connection creation.

Creating a connection

To create a connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**.
3. On the wizard page that opens, use the following options:

- **Connection name**: Type a descriptive name for the connection being created.
- **Use the specified connector**: From this list, select the connector you want to use.
- **Remote connector access**: Expand this element to specify settings to access the connector installed on a remote computer. For more information, see Using connectors installed remotely.

4. Follow the steps in the wizard to create a connection.

For information on the options you can use in the subsequent steps of the wizard, see the section for the connector you have selected.

Renaming a connection

You can rename any existing data connection in the Active Roles Synchronization Service Console.

To rename a connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click the name of the existing connection you want to rename.
3. On the **General** tab, edit the connection name in the **Connection name** box.
4. Click **Save**.

Deleting a connection

To delete a connection

1. In the Synchronization Service Console, open the **Connections** tab.
2. Locate the connection you want to delete, and then click **Delete connection** for that connection.
3. When prompted, confirm that you want to delete the connection.

Modifying synchronization scope for a connection

For each connected data system, you can modify the scope of objects participating in the data synchronization operations.

To modify the synchronization scope

1. In the Synchronization Service Console, open the **Connections** tab.
2. Locate the connection for which you want to modify the synchronization scope, then click **Synchronization scope**.
3. Use the following options to modify the synchronization scope:
   - **Include objects from selected containers only**: Select the check boxes next to the containers that hold the objects you want to participate in data synchronization operations.

     NOTE: This option may be unavailable for some types of connected data systems, such as Microsoft SQL Server or Oracle Database.
   - **Objects must meet these conditions**: Set up a list of conditions that objects must meet in order to participate in data synchronization operations.
4. When you are finished, click **Save**.

Using connection handlers

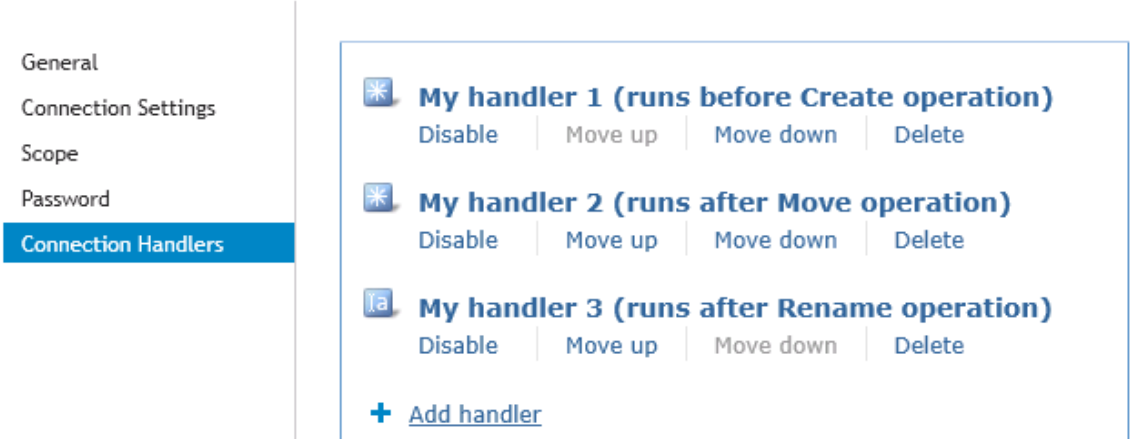
Connection handlers allow you to automatically perform specific actions on connected data systems before, after, or instead of specific data synchronization operations (such as create, modify, move, rename, delete, or password synchronization). When creating a connection handler, you can specify the action you want to perform and set the conditions for triggering the action.

By default, Synchronization Service includes only one built-in handler type that can run your custom PowerShell script to perform the action you want. However, you can also implement your own custom handler types.

IMPORTANT: If the predefined connection handler is configured to run your PowerShell script instead of a data synchronization operation, the script must return a system entry object.

To create, modify, or delete handlers for a connection, use the **Connection Handlers** tab in the connection settings:

Figure 7: Connection Handlers



This tab provides the following elements:

- **Add handler**: Starts a wizard that helps you add a new connection handler. By default, the wizard creates a new handler that allows you to run your PowerShell script.
- **Disable**: Disables the connection handler.
- **Enable**: Enables the connection handler.
- **Move up**: Moves the connection handler one position up in the list.
- **Move down**: Moves the connection handler one position down in the list.
- **Delete**: Deletes the connection handler.

To create a connection handler

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click the name of the connection for which you want to create a handler, then click the **Connection Handlers** tab.
3. Click **Add handler**, then follow the steps in the wizard to create your handler.

To modify a connection handler

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click the name of the connection for which you want to modify a handler, then click the **Connection Handlers** tab.
3. Click the name of the handler you want to modify, then modify the handler settings as necessary. When you are finished, click **OK**.
4. You can also do the following:
    - **Change the order in which handlers are activated**: Synchronization Service activates handlers in the order in which they appear in the list. To move a handler in the list, use the **Move up** and **Move down** links below the

handler.

- **Disable or enable handlers**: You can enable or disable existing handlers. To do so, use the **Enable** or **Disable** link below the handler.

5. When you are finished, click **Save**.

To delete a connection handler

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click the name of the connection for which you want to delete a handler, then click the **Connection Handlers** tab.
3. Click **Delete** below the handler you want to delete.

Specifying password synchronization settings for a connection

For each connected data system that supports password synchronization, you can set password synchronization settings. These settings allow you to enable or disable password synchronization and manage passwords in the data system by using One Identity Password Manager.

Optionally, you can use the password synchronization settings to specify a custom Windows PowerShell script you want to run each time the password synchronization completes for the connected data system.

To specify password synchronization settings

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click the name of the connection for which you want to modify password synchronization settings.
3. Open the **Password** tab, and use the following options to modify the password synchronization settings as necessary:
   - **Synchronize and manage passwords**: Allows you to enable or disable password synchronization for this connection. Selecting this check box also allows you to manage passwords in the connected data system by using One Identity Password Manager. For more information on Password Manager, see https://www.oneidentity.com/products/password-manager/.
   - **Synchronize passwords for objects of this type**: Allows you to specify an object type that will participate in password synchronization. Click **Select** next to this text box, then specify the object type you want.

     NOTE: This option is only available for certain types of connected systems, such as the LDAP directory service.

- **Password synchronization method**: Allows you to select a password synchronization method.

  NOTE: This option is only available for certain types of connected systems, such as LDAP directory service.

  You can select one of the following methods:

  - **Write password to this attribute**: Displays the object attribute in which the object password will be stored. To specify a different attribute, click **Select** next to the text box in this option.
  - **Use LDAP extended operation**: Allows you to automate the synchronization of user passwords in the connected data system regardless of the form of the authentication identity or the password storage mechanism used (for example, in the case of non-directory storage of passwords).
  - **Configure Query**: Allows you to use an SQL query to specify the data you want to participate in the password synchronization. Click **Configure**, then type your SQL query.

    NOTE: This option is only available for certain types of connected systems, such as LDAP directory service or Oracle Database.

4. When you are finished, click **Save**.

5

Synchronizing identity data

To synchronize identity data between connected data systems, you can use sync workflows and synchronization steps. A sync workflow is a set of data synchronization operations called synchronization steps. A sync workflow can include one or more steps. Each synchronization step defines a synchronization operation to be run between the source and target connected data systems. To manage sync workflows and their steps, you can use the **Sync Workflows** in the Synchronization Service Console.

You can configure a synchronization step to perform one of the following operations:

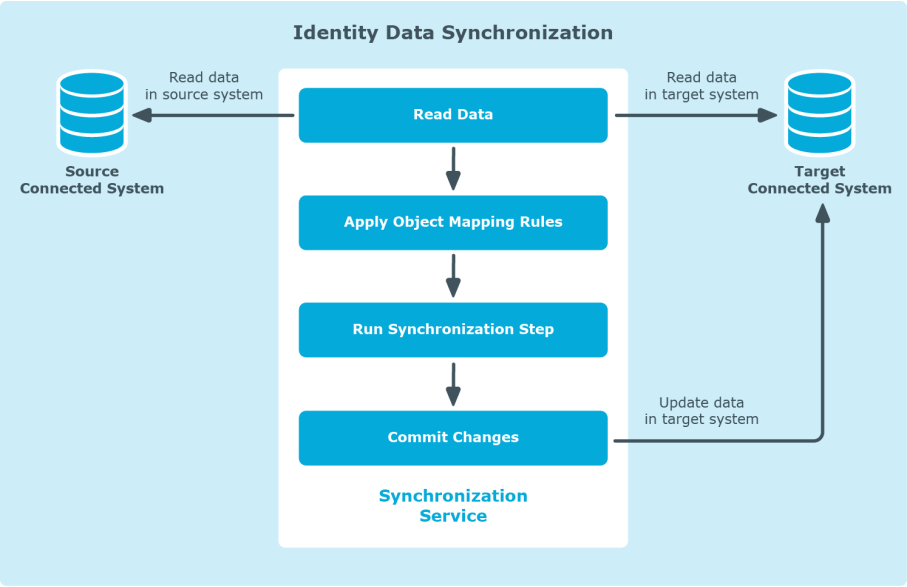
- **Creation**: Creates objects in the target data system based on the changes made to specific objects in the source data system. When creating a new object in the target data system, Synchronization Service generates initial values for the object attributes using the attribute population rules you have configured.
- **Update**: Modifies object attributes in the target data system based on the changes made to specific objects in the source data system. To specify the objects that will participate in the update operation you can use object mapping rules. For more information, see Mapping objects.
- **Deprovision**: Modifies or removes objects in the target data system after their counterparts have been disconnected from the source data system. Synchronization Service can be configured to remove target objects permanently or change them to a specific state. To specify the objects that will participate in the deprovision operation you can use object mapping rules. For more information, see Mapping objects.

When configuring a synchronization step you can specify the following:

- Containers to which you want to create or move objects.
- Settings to generate names for objects being created or modified.
- Settings to synchronize group memberships.
- Settings to synchronize attribute values.

To synchronize identity data between two data systems, you need to create a sync workflow, populate the workflow with synchronization steps, and then run the sync workflow manually or schedule the sync workflow run. The following figure illustrates how Synchronization Service synchronizes identity data in connected data systems:

Figure 8: Identity Data Synchronization



Running a sync workflow causes Synchronization Service to read data in the source and target data systems according to the settings in the sync workflow steps and prepare a list of changes to be made in the target system. Then, you can commit these changes to the target data system.

Running a sync workflow manually allows you to review a list of changes before committing them to the target data system. A scheduled sync workflow run always commits changes to the target data system automatically.

You can configure as many sync workflows as needed, each performing its own set of synchronization steps.

Creating a sync workflow

To create a sync workflow

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click **Add sync workflow**.
3. In the **Sync workflow name** text box, type a name for the sync workflow being created.
4. Click **OK**.

   The new workflow appears on the **Sync Workflows** tab.

   NOTE: After you created a sync workflow, you must populate it with one or more synchronization steps. For more information, see Synchronizing identity data.

Running a sync workflow

After you created a sync workflow and populated it with one or more steps, you can run the sync workflow. Before running a sync workflow, you can select the workflow steps you want to run. You can run a sync workflow either manually, or automatically on a recurring schedule.

Running a sync workflow manually

This method allows you to select specific steps in a sync workflow and run them. You can also specify how you want to commit the changes to the target data system: automatically or manually. With the manual method you can review a list of changes before committing them to decide whether or not you want these changes in the target system.

To run a sync workflow manually

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the sync workflow you want to run.
3. Click **Run now**.
4. Select the check boxes next to the sync workflow steps you want to run.
5. If you want to automatically commit the changes made by the sync workflow run, select the **Automatically commit changes** check box. If you want to review the changes before committing them, leave this check box cleared.
6. Click one of the following to run the sync workflow:
    - **Full Run**: With this option, Synchronization Service retrieves the data required to run the sync workflow from the connected data systems.
    - **Quick Run**: With this option, Synchronization Service first tries to run the sync workflow by using the data that is available in the local cache. If the local cache is missing or cannot be used to run the sync workflow, then Synchronization Service retrieves the required data from the connected data systems.

Running a sync workflow on a recurring schedule

This method allows you to create a recurring schedule to automatically run specific steps in a sync workflow.

When scheduling a sync workflow, you can choose the workflow steps to run, specify how frequently you want to run the steps, and set the date and time when you want the run schedule to come into effect. If you have two or more Synchronization Service instances

installed in your environment, you can also select a Synchronization Service instance to be used for running the sync workflow.

A scheduled sync workflow automatically commits changes to the target data system.

To run a sync workflow on a recurring schedule

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click **Schedule** below the name of the sync workflow you want to run on a recurring schedule.
3. In the dialog that opens, select **Schedule the task to run**, then specify a schedule.
4. If there are several Synchronization Service instances deployed in your environment, under **Run the task on**, select the computer that hosts the Synchronization Service instance you want to use for running the sync workflow.
5. Expand **Sync Workflow Steps**, and then select the workflow steps you want to run on the schedule.
6. To activate the schedule, click **OK**.

Disabling a sync workflow run schedule

To disable a sync workflow run schedule

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click **Schedule** below the sync workflow for which you want to disable the run schedule.
3. In the dialog that opens, clear the **Schedule the task to run** check box.
4. To disable the schedule, click **OK**.

Renaming a sync workflow

To rename a sync workflow

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click **Rename** below the sync workflow.
3. In the **Sync workflow name** text box, type a new workflow name.
4. Click **OK** to apply the change.

Deleting a sync workflow

To delete a sync workflow

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click **Delete** below the sync workflow.
3. When prompted, confirm that you want to delete the sync workflow.

Adding a creating step

You can create a synchronization connection between two object types of two connected data systems in a sync workflow with the **Add synchronization step** < **Creation** setting. Typically, you need to specify a creation step either when configuring a new sync workflow, or must configure a new synchronization connection between two new object types in an existing sync workflow.

To add a creating step

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the sync workflow in which you want to add a creating step.

   If necessary, create a new sync workflow. For more information, see Creating a sync workflow.
3. Click **Add synchronization step**.
4. Select **Creation**, then click **Next**.
5. Specify the source system by using these options:
   - **Source connected system**: Allows you to choose a source data system for the creation operation. Click **Specify** to select a data system connected earlier or add and select a new data system.
   - **Source object type**: Allows you to specify the object type you want to use as a source for the creation operation. Click **Select** to specify an object type.
   - **Creation Criteria**: Allows you to narrow the scope of source data system objects that participate in the creating step. To specify the containers that hold the source objects you want to participate in the step, expand **Creation Criteria**. You can also specify additional conditions to include objects into the scope.
6. Click **Next**.
7. Specify the creation target by using these options:
   - **Target connected system**: Allows you to choose a target data system for the creation operation. To select a data system connected earlier or add and select

a new data system, click **Specify**.

- **Target object type**: Allows you to specify the target data system object type to which you want to create objects from the source data system. To specify an object type, click **Select**.
- **Target container**: Allows you to specify the target data system container in which you want to create objects. Click the down arrow on the button, and then select one of the following:
  - **Browse**: Click to locate and select a single target container.
  - **PowerShell Script**: Click to compose a PowerShell script that calculates the target container name.
  - **Rule**: Click to configure a set of rules for selecting target containers.
  - **Use Mapping**: Click to define a target container based on the mapping of the source object.
  - **Clear**: Click to use an empty value.
- **Rules to generate unique object name**: Allows you to set up a list of rules to generate a unique name for each object being created. For more information, see Generating object names by using rules.

8. Click **Next**.
9. Specify rules to create objects into the target data system. You can use the following options:
   - **Initial Attribute Population Rules**: Expand this element to specify how you want to populate the attributes of created objects. For more information, see Modifying attribute values by using rules.
   - **Initial Password**: Expand this element to specify an initial password for each created object.
   - **User Account Options**: Expand this element to specify settings for the user accounts to be created.
10. To add the creating step, click **Finish**.

You can modify the settings of an existing synchronization step. For more information, see Modifying an existing sync workflow step.

Creating an update step

You can update an existing synchronization step between two object types of two connected data systems in a sync workflow with the **Add synchronization step** < **Update** setting.

To create an updating step

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the sync workflow in which you want to create an updating step.

   If necessary, create a new sync workflow. For more information, see Creating a sync workflow.
3. Click **Add synchronization step**.
4. Select **Update**, then click **Next**.
5. Specify the update operation source by using these options:
   - **Source connected system**: Allows you to choose a source data system for the update operation. To select a data system connected earlier or add and select a new data system, click **Specify**.
   - **Source object type**: Allows you to specify the data system object type you want to use as a source for the update operation. To specify an object type, click **Select**.
   - **Updating Criteria**: Allows you to narrow the scope of source data system objects that will participate in the updating step. To specify the containers that hold the source objects you want to participate in the step, expand **Updating Criteria**. You can also specify additional criteria for selecting source objects.
6. Click **Next**.
7. Specify an update target by using these options:
   - **Target connected system**: Allows you to choose a target connected system for the update operation. To select a data system connected earlier or add and select a new data system, click **Specify**.
   - **Target object type**: Allows you to specify what type of objects you want to update in the target data system. To specify an object type, click **Select**.
8. Click **Next**.
9. Specify rules to update objects in the target data system. You can use the following options:
   - **Rules to Modify Object Attributes**: Allows you to set up a list of rules to modify object attributes in the target data system. For more information, see Modifying attribute values by using rules.
   - **Rules to Move Objects**: Expand this option to specify the location to which you want to move objects. Click the down arrow on the button, and then select one of the following:
     - **Browse**: Click to locate and select a single target container.
     - **PowerShell Script**: Click to compose a PowerShell script that calculates the target container name.
     - **Rule**: Click to configure a set of rules for selecting target containers.

    - **Use Mapping**: Click to define a target container based on the mapping of the source object.
    - **Clear**: Click to use an empty value.
  - **Rules to Rename Objects**: Allows you to view or change the list of rules used to rename target objects. For more information, see Generating object names by using rules.
10. Click **Finish** to create the updating step.

You can modify the settings of an existing synchronization step. For more information, see Modifying an existing sync workflow step.

Creating a deprovisioning step

You can create a deprovisioning step between two object types of two connected data systems in a sync workflow with the **Add synchronization step** < **Deprovision** setting. Creating a deprovisioning step instead of a deletion step is typically recommended to allow you reprovisioning the affected data objects later.

To create a deprovisioning step

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the sync workflow in which you want to create a deprovisioning step.

   If necessary, create a new sync workflow. For more information, see Creating a sync workflow.
3. Click **Add synchronization step**.
4. Select **Deprovision**, then click **Next**.
5. Specify a deprovisioning source and criteria by using the following options:
   - **Source connected system**: Allows you to choose a source data system for the deprovision operation. To select a data system connected earlier or add and select a new data system, click **Specify**.
   - **Source object type**: Allows you to specify the data system object type you want to use as a source for the deprovision operation. To specify an object type, click **Select**.
   - **Deprovision target objects if**: Allows you to specify criteria for deprovisioning objects in the target data system.
6. Click **Next**.
7. Specify a deprovisioning target by using the following options:
   - **Target connected system**: Allows you to choose a target data system for the deprovision operation. To select a data system connected earlier or add and

select a new data system, click **Specify**.

- **Target object type**: Allows you to specify what type of objects you want to deprovision in the target data system. To specify an object type, click **Select**.

8. Click **Next**.
9. Select a method to deprovision objects in the target data system. You can select **Delete target objects** to delete target objects or **Modify target objects** to modify target objects using the rules configured in the following options:
   - **Rules to Modify Object Attributes**: Allows you to set up a list of rules to modify object attributes in the target data system. For more information, see Modifying attribute values by using rules.
   - **Rules to Move Objects**: Expand this option to specify the location to which you want to move objects. Click the down arrow on the button, and then select one of the following:
     - **Browse**: Click to locate and select a single target container.
     - **PowerShell Script**: Click to compose a PowerShell script that calculates the target container name.
     - **Rule**: Click to configure a set of rules for selecting target containers.
     - **Use Mapping**: Click to define a target container based on the mapping of the source object.
     - **Clear**: Click to use an empty value.
   - **Rules to Rename Objects**: Allows you to view or change the list of rules used to rename target objects. For more information, see Generating object names by using rules.
10. Click **Finish** to create the deprovisioning step.

You can modify the settings of an existing synchronization step. For more information, see Modifying an existing sync workflow step.

Modifying an existing sync workflow step

You can modify the existing steps of sync workflows in the Synchronization Service Console, including their general options, source and target data system settings, or synchronization rules.

To modify an existing step

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the sync workflow in which you want to modify a step.
3. Click the name of the step you want to modify.

4. Use the following tabs to modify the step as necessary:
    - General Options
    - Source
    - Target
    - Creation Rules
    - Deprovisioning Rules
    - Updating Rules
    - Step Handlers

    For more information on these tabs, see the next subsections.
5. When you are finished, click **Save** to apply your changes.

General Options

The general options allow you to rename the step, specify a method for processing data in the source and target connected systems, and specify conditions to stop data processing.

This tab has the following elements:

- **Step name**: Allows you to rename the step.
- **Specify how to process data in connected systems**: Allows you to select how to process data during synchronization. The available methods are the following:
    - **Process all data**: Each run of the step will process all data in the configured synchronization scope.
    - **Process delta from last run**: Each run of the step will process only the data that has changed in the configured synchronization scope since the last run.
- **Stop data processing if**: Allows you to specify conditions that will stop data processing in the source and target data systems when met.

Source

The **Source** setting allows you to view information about the source connected system and source object type specified for the synchronization step. You can also view or modify the criteria used to perform the creation, deprovision, or update operation in the step.

For all types of synchronization steps (creating, deprovisioning, and updating), this tab provides the following options:

- **Source connected system**: Displays the name of the source data system.
- **Source object type**: Displays the object type that is used as a source for the synchronization step.

The **Source** tab also provides the following options depending on the type of step you select:

- For deprovisioning steps, the **Deprovision target objects if** option. This option allows you to modify the criteria used for triggering the deprovision operation in the target data system.
- For creation steps,the **Creation Criteria** option. This option allows you to modify the scope of source data system objects that participate in the creation step. To modify the list of containers that hold the source objects you want to participate in the step, expand **Creation Criteria**. Also, you can specify additional criteria for selecting source objects.
- For update steps, the **Updating Criteria** option. This option allows you to modify the scope of the source data system objects that participate in the updating step. To specify the containers that hold the source objects you want to participate in the step, expand **Updating Criteria**. You can also specify additional criteria for selecting source objects.

Target

The **Target** setting allows you to view information about the target connected system and target object type specified for the synchronization step. For creating steps, you can use this tab to view and modify the target container to which objects are created and rules to generate unique names for created objects.

For all types of synchronization steps (creating, deprovisioning, and updating) this tab provides the following elements:

- **Target connected system**: Displays the name of the data system that is currently used as a target for the synchronization step.
- **Target object type**: Displays the object type that is currently used as a target for the synchronization step.

For creating steps related to certain types of target data systems, this tab may also provide any of the following additional elements:

- **Target container**: Allows you to specify the target data system container in which you want to create objects from the source data system. For more information, see Generating object names by using rules.
- **Rules to generate unique object name**: Allows you to set up a list of rules to generate a unique name for each object being created. For more information, see Generating object names by using rules.

Creation Rules

Creation rules allow you to view or modify the rules used for creating objects. This tab has the following elements:

- **Initial Attribute Population Rules**: Expand this element to view or modify the rules for populating the attributes of objects being created.
- **Initial Password**: Expand this element to view or modify how an initial password is generated for each object being created.
- **User Account Options**: Expand this element to view or modify the settings used for creating user accounts in the result of the creation operation.

You can use this tab to import or export initial attribute population rules.

To export a population rule to a file

1. In the list of configured attribute population rules, select the rule you want to export.
2. Click **More**, then click **Export**.
3. In the **Save As** dialog, specify an XML file to store the rule.

To import a population rule from a file

1. Expand **Initial Attribute Population Rules**, click **More**, then click **Import**.
2. Use the **Open** dialog to open the XML file that stores the population rule to import.

Deprovisioning Rules

Deprovisioning rules allow you to select a method for deprovisioning synchronized objects. As part of deprovisioning, you can either delete the target objects if the source objects meet the synchronization criteria configured in the wizard, or just modify the target objects using the following deprovisioning rules.

- **Rules to Modify Object Attributes**: Allows you to set up a list of rules to modify object attributes in the target data system. For more information, see Modifying attribute values by using rules.
- **Rules to Move Objects**: Expand this option to specify the location to which you want to move objects. Click the down arrow on the button, and then select one of the following:
    - **Browse**: Click to locate and select a single target container.
    - **PowerShell Script**: Click to compose a PowerShell script that calculates the target container name.
    - **Rule**: Click to configure a set of rules for selecting target containers.
    - **Use Mapping**: Click to define a target container based on the mapping of the source object.
    - **Clear**: Click to use an empty value.
- **Rules to Rename Objects**: Allows you to view or change the list of rules used to rename target objects. For more information, see Generating object names by using rules.

Updating Rules

Updating rules allow you to view or modify the rules used for updating objects. This tab has the following elements:

- **Rules to Modify Object Attributes**: Allows you to set up a list of rules to modify object attributes in the target data system. For more information, see Modifying attribute values by using rules.
- **Rules to Move Objects**: Expand this option to specify the location to which you want to move objects. Click the down arrow on the button, and then select one of the following:
  - **Browse**: Click to locate and select a single target container.
  - **PowerShell Script**: Click to compose a PowerShell script that calculates the target container name.
  - **Rule**: Click to configure a set of rules for selecting target containers.
  - **Use Mapping**: Click to define a target container based on the mapping of the source object.
  - **Clear**: Click to use an empty value.
- **Rules to Rename Objects**: Allows you to view or change the list of rules used to rename target objects. For more information, see Generating object names by using rules.

Step Handlers

Step handlers allow you to create, modify, or delete handlers for a sync workflow. For more information on how to use step handlers, see Using sync workflow step handlers. This tab has the following elements:

- **Add handler**: Starts a wizard that helps you add a new handler for the sync workflow step. By default, the wizard creates a new handler that runs your PowerShell script.
- **Disable**: Disables the step handler.
- **Enable**: Enables the step handler.
- **Move up**: Moves the step handler one position up in the list.
- **Move down**: Moves the step handler one position down in the list.
- **Delete**: Deletes the step handler.

Deleting a sync workflow step

You can delete steps in a sync workflow. This is typically required when performing maintenance and housekeeping on the configured sync workflows, making sure that they do not contain any outdated or unnecessary steps.

To delete a sync workflow step

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the sync workflow in which you want to delete a step.
3. Click **Delete** below the step you want to delete.
4. When prompted, confirm that you want to delete the step.

Changing the order of steps in a sync workflow

When you run a sync workflow, its steps are performed in the order they appear in the Synchronization Service Console. However, if necessary, you can change the order of these steps.

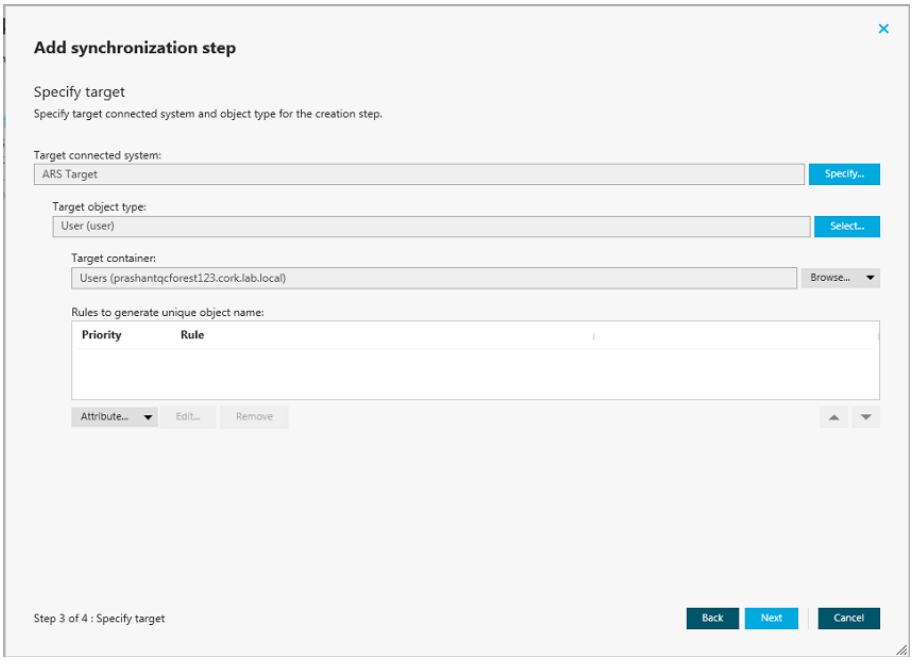
To change the order of steps in a sync workflow

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the sync workflow in which you want to change the order of steps.
3. Use the **Move up** and **Move down** links to arrange the steps as necessary.

Generating object names by using rules

When configuring a synchronization step, you can use the **Rules to generate unique object name** list to specify rules for creating or modifying object names in the target connected system. The **Rules to generate unique object name** list looks similar to the following:

Figure 9: Add synchronization step



To configure rules for generating object names

1. Click the down arrow on the leftmost button provided below the **Rules to generate unique object name** list.
2. Select a list item:
   - **Attribute**: Allows you to select the target object attribute whose value you want to use as the object name.
   - **Rule**: Allows you to configure a rule to generate target object names. For details, see Using value generation rules.
   - **PowerShell Script**: Allows you to type a PowerShell script to generate target object names.

When the **Rules to generate unique object name** list includes two or more entries, Synchronization Service uses the uppermost rule in the list to generate the target object name. If the generated object name is not unique, Synchronization Service uses the next rule in the list, and so on.

To copy and paste an existing rule

1. In the **Rules to generate unique object name** list, right-click a rule, then select **Copy** from the shortcut menu.
2. In the rules list, right-click an entry, then select **Paste** from the shortcut menu.

Modifying attribute values by using rules

In a sync workflow step you can configure a set of rules to automatically modify attribute values during the step run. By using these rules, you can select or generate an initial value, transform this value if necessary, and then assign the resulting value to the object attribute you want.

To create a rule to modify attribute values

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the appropriate sync workflow, then click the name of the sync workflow step.
3. Depending on the workflow step type, complete the corresponding actions:
    - **Creating step**: Click the **Creation Rules** tab, and then expand the **Initial Attribute Population Rules** element.
    - **Updating step**: Click the **Updating Rules** tab, and then expand the **Rules to Modify Object Attributes** element.
    - **Deprovisioning step**: Click the **Deprovisioning Rules** tab, and then expand the **Rules to Modify Object Attributes** element.
4. In the element you have expanded, click the down arrow on the leftmost button to select a rule type:
    - **Forward Sync Rule**: Allows you to create a rule that synchronizes attribute values from the source to the target data system. This type of rule is available in creating, updating, and deprovisioning steps. For more information, see Configuring a forward sync rule.
    - **Reverse Sync Rule**: Allows you to create a rule that synchronizes attribute values from the target to the source data system. This type of rule is available in creating, updating, and deprovisioning steps. For more information, see Configuring a reverse sync rule.
    - **Merge Sync Rule**: Allows you to create a rule that merges the values of specified attributes between the source and the target data systems. As a result, the attribute values in the source and the target become identical. This type of rule is only available in updating steps. For more information, see Configuring a merge sync rule.

Configuring a forward sync rule

A forward sync rule allows you to synchronize data from the source data system to the target data system. To create such a rule, follow the instructions in Modifying attribute values by using rules to select the **Forward Sync Rule** type. Then, configure your rule by using the options in the dialog that opens.

Forward sync rule source item

This option allows you to obtain an initial value for the synchronization operation. You can then transform the obtained initial value before assigning it to the attribute you want.

To get started, click the down arrow on the button in this option, and then select an item from the drop-down list:

- **Attribute**: Allows you to select the attribute whose value you want to use.
- **Rule**: Allows you to obtain a value by using a value generation rule. For more information, see Using value generation rules.
- **PowerShell script**: Allows you to obtain a value by running a Windows PowerShell script.
- **Text**: Allows you to type a text value.
- **Referenced object attribute**: Allows you select an attribute of a referenced object and use the value of the selected attribute.
- **Parent object attribute**: Allows you to select an attribute of a parent object and use the value of the selected attribute.
- **Empty**: Generates an empty value.

Once you have explicitly selected an attribute in this option, you can click the **Advanced** link to configure some advanced synchronization settings for the attribute.

For example, you can specify which characters to retrieve from the attribute value, how to modify the retrieved value (remove white-space characters or change the capitalization), or set how to process references in the attribute. The available settings depend on the attribute types selected in the **Source item** and **Target item** options.

Forward sync rule target item

This option allows you to select the target attribute whose value you want to modify.

To get started, click the down arrow on the button in this option, and then select an item from the drop-down list:

- **Attribute**: Allows you to select the object attribute whose value you want to modify.
- **Referenced object attribute**: Allows you to select the referenced object attribute whose value you want to modify.
- **Parent object attribute**: Allows you to modify attribute values of objects that are parents to the target object type selected in the sync workflow step settings.

Once you have explicitly selected an attribute in this option, you can click the **Advanced** link to configure some advanced synchronization settings for the attribute.

For example, you can select how to handle the existing attribute value (overwrite or append data to the value) or set how to process references in the attribute. The available settings depend on the attribute types selected in the **Source item** and **Target item** options.

Configuring a reverse sync rule

A reverse sync rule allows you to synchronize data from the target to the source data system.

To create such a rule, follow the instructions in Modifying attribute values by using rules to select the **Reverse Sync Rule** type. Then, configure your rule by using the options in the dialog that opens.

Reverse sync rule source item

This option allows you to select the source attribute whose value you want to modify.

To get started, click the down arrow on the button in this option, and then select an item from the drop-down list:

- **Attribute**: Allows you to select the object attribute whose value you want to modify.
- **Referenced object attribute**: Allows you to select the referenced object attribute whose value you want to modify.
- **Parent object attribute**: Allows you to modify attribute values of objects that are parents to the source object type selected in the sync workflow step settings.

Once you have explicitly selected an attribute in this option, you can click the **Advanced** link to configure some advanced synchronization settings for the attribute.

For example, you can select how to handle the existing attribute value (overwrite or append data to the value) or set how to process references in the attribute. The available settings depend on the attribute types selected in the **Source item** and **Target item** options.

Reverse sync rule target item

This option allows you to obtain an initial value for the synchronization operation. You can then transform the obtained initial value before assigning it to the attribute you want.

To get started, click the down arrow on the button in this option, and then select an item from the drop-down list:

- **Attribute**: Allows you to select the attribute whose value you want to use.
- **Rule**: Allows you to obtain a value by using a value generation rule. For more information, see Using value generation rules.
- **PowerShell script**: Allows you to obtain a value by running a Windows PowerShell script.
- **Text**: Allows you to type a text value.

- **Referenced object attribute**: Allows you select an attribute of a referenced object and use the value of the selected attribute.
- **Parent object attribute**: Allows you to select an attribute of a parent object and use the value of the selected attribute.
- **Empty**: Generates an empty value.

Once you have explicitly selected an attribute in this option, you can click the **Advanced** link to configure some advanced synchronization settings for the attribute.

For example, you can specify which characters to retrieve from the attribute value, how to modify the retrieved value (remove white-space characters or change the capitalization), or set how to process references in the attribute. The available settings depend on the attribute types selected in the **Source item** and **Target item** options.

Configuring a merge sync rule

A merge sync rule allows you to merge attribute values between the source and the target data system. As a result, these values become identical.

To create such a rule, follow the instructions in Modifying attribute values by using rules to select the **Merge Sync Rule** type. Then, configure your rule by using the options in the dialog that opens:

- **Source item**: Allows you to specify an attribute in the source data system. To select an attribute, click **Attribute**.
- **Target item**: Allows you to specify the attribute in the target data system. To select an attribute, click **Attribute**.
- **Merge Settings**: Allows you to select a method to merge the values of two multivalued attributes. This link is only available if both the source and the target attributes you have selected are multivalued.

When running a sync workflow step that has a merge sync rule configured for the first time, Synchronization Service synchronizes attribute values from the source to the target. In each subsequent run of the sync workflow step, the synchronization direction depends on which attribute value (source or target) is more recent, as follows:

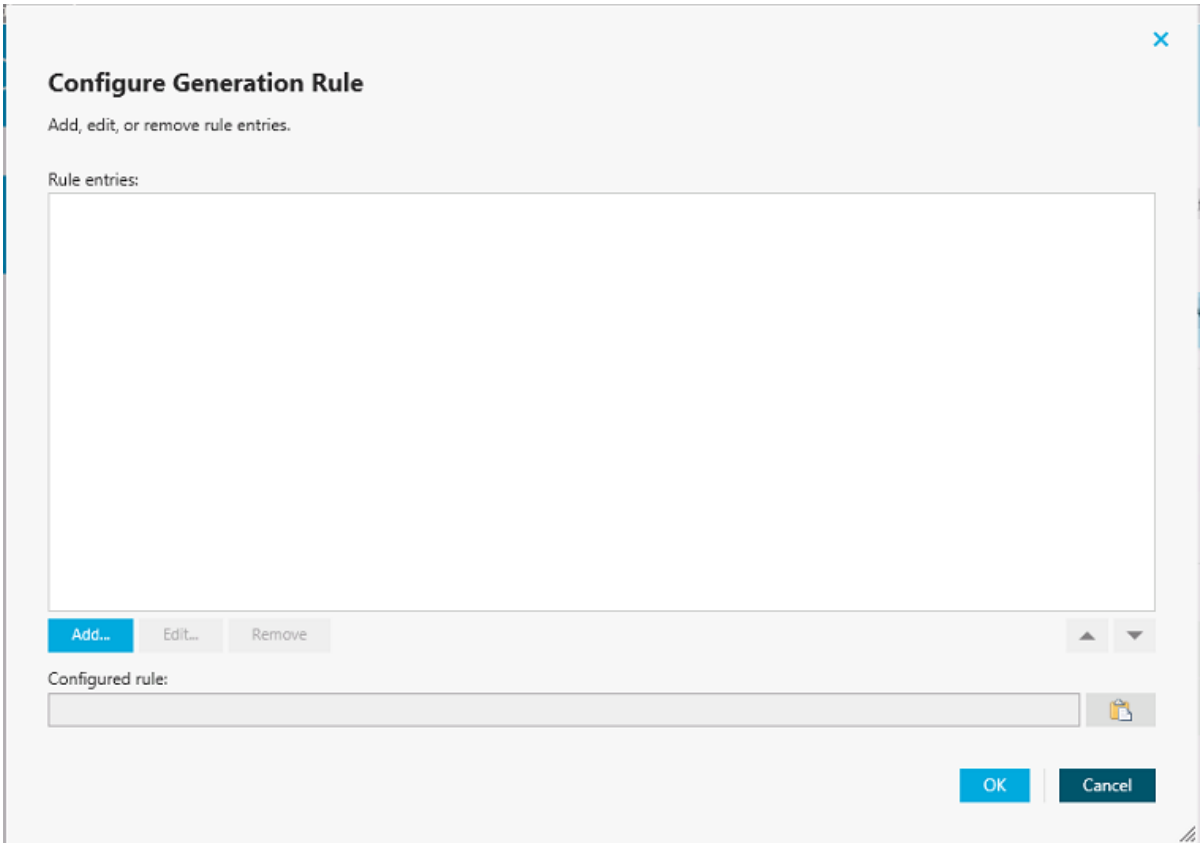
Table 105: Synchronization direction

More recent value	Synchronization direction
Source	Source => Target
Target	Source <= Target
Source and target are equally recent	Source => Target

Using value generation rules

To configure a list of rules for selecting an attribute value or generating a value, you can use the **Configure Generation Rule** dialog.

Figure 10: Configure Generation Rule



To add a new rule entry

1. Click **Add**.
2. Configure the rule entry as appropriate. For more information, see Configuring a rule entry.

To remove an existing rule entry

- Open the **Rule entries** list.
- From the **Rule entries** list, select the entry you want to remove, then click **Remove**.

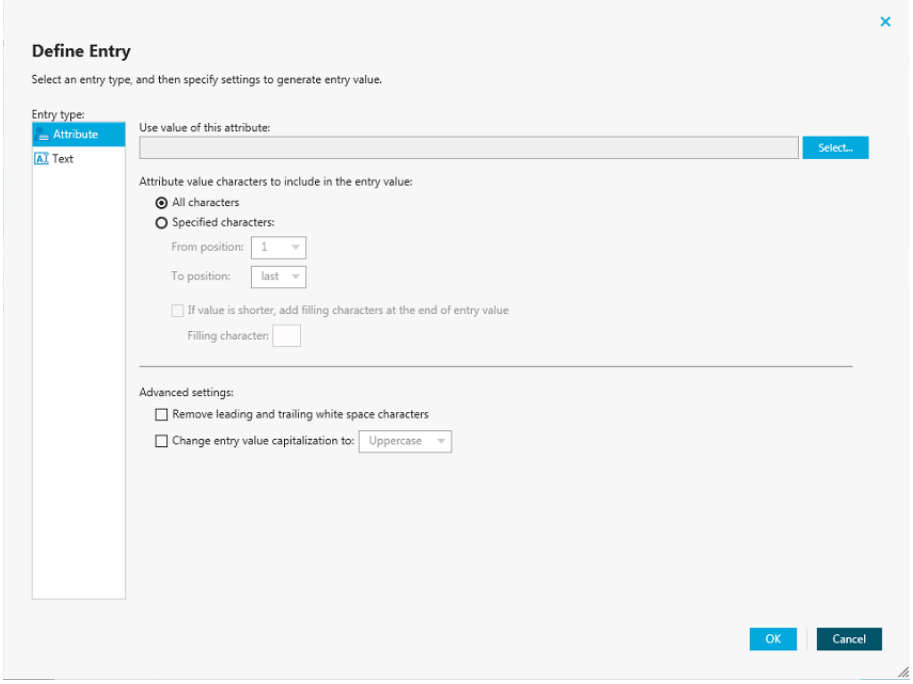
To edit an existing rule entry

1. From the **Rule entries** list, select the entry you want to modify, then click **Edit**.
2. Configure the rule entry as appropriate. For more information, see Configuring a rule entry.

Configuring a rule entry

This section provides instructions on how to configure a rule entry in the **Define Entry** dialog that looks similar to the following:

Figure 11: Define Entry



To configure a text entry

1. Under **Entry type**, select **Text**.
2. In the **Text value** box, type the value.
3. Click **OK**.

To configure an attribute-based entry

1. Under **Entry type**, select **Attribute**.
2. Click **Select** to select the attribute whose value you want to use, and then click **OK**.

3. If you want the entry to include the entire value of the attribute, select **All characters**. Otherwise, click **Specified characters**, then specify the characters to include in the entry.
4. (Optional) To add additional characters to the entry, click **If value is shorter, add filling characters at the end of entry value**.
5. (Optional) Specify **Advanced settings**.
6. When finished, click **OK**.

Using sync workflow step handlers

Sync workflow step handlers allow you to automatically perform custom actions either before running a workflow step or after the workflow step run results have been committed (written) to the data system. Out of the box, Synchronization Service includes a single predefined handler type that can automatically run your custom PowerShell script and thus perform the desired action.

To create, modify, or delete handlers for a sync workflow step, you can use the Step Handlers tab in the sync workflow step properties.

To create a sync workflow step handler

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the appropriate sync workflow.
3. Click the name of the sync workflow step for which you want to create a handler, then click **Step Handlers**.
4. Click **Add handler**, then follow the steps in the wizard to create your handler.

To modify a sync workflow step handler

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the appropriate sync workflow.
3. Click the name of the sync workflow step whose handler you want to modify, then click the **Step Handlers** tab.
4. Click the name of the handler you want to modify.
5. Modify the handler settings as necessary. When you are finished, click **OK**.
6. You can also do the following:
   - **Change the order in which handlers are activated**: Synchronization Service activates handlers in the order in which they appear in the list. To move a handler in the list, use the **Move up** and **Move down** links below the handler.

- **Disable or enable the handler**: You can enable or disable existing handlers. To do so, use the **Enable** or **Disable** link below the handler.

7. When you are finished, click **Save**.

To delete a sync workflow step handler

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the name of the appropriate sync workflow.
3. Click the name of the sync workflow step whose handler you want to delete, and then click **Step Handlers**.
4. Click **Delete** below the handler you want to delete.

Example: Synchronizing group memberships

This example illustrates how to configure a creating step to synchronize group memberships from an Active Directory domain to an AD LDS (ADAM) instance. The example demonstrates how to create rules in the step to synchronize the value of the **member** attribute in the AD domain to the **member** attribute in AD LDS (ADAM).

To synchronize the member attribute

1. Follow the procedure of the Adding a creating step section until you reach the **Specify creation rules** page.
2. In the **Initial Attribute Population Rules** element, click the down arrow on the leftmost button below the list to select **Forward Sync Rule**.
3. In the dialog that opens, add the following pair of attributes:
   - Source item: **member** attribute (Active Directory)
   - Target item: **member** attribute (AD LDS)

   For more information about the options in this dialog, see Configuring a forward sync rule.
4. When you are finished, click **OK**.
5. Follow the steps in the wizard to complete the creating step.

Example: Synchronizing multivalued attributes

This example illustrates how to configure a creating step to synchronize multivalued attributes from an Active Directory domain to an AD LDS (ADAM) instance. The example

demonstrates how to create rules in the step to synchronize the value of the **otherTelephone** attribute in the Active Directory domain to the **otherTelephone** attribute in AD LDS (ADAM).

To synchronize the otherTelephone attribute

1. Follow the procedure of the Adding a creating step section until you reach the **Specify creation rules** page.
2. In the **Initial Attribute Population Rules** element, click the down arrow on the leftmost button below the list to select **Forward Sync Rule**.
3. In the dialog that opens, add the following pair of attributes:
   - Source item: **otherTelephone** attribute (Active Directory)
   - Target item: **otherTelephone** attribute (AD LDS)

   For more information about the options in this dialog, see Configuring a forward sync rule.
4. When you are finished, click **OK**.
5. Follow the steps in the wizard to complete the configuration of the creating step.

Using sync workflow alerts

The Synchronization Service provides an email notification service that allows you to inform recipients about the completion of a sync workflow run.

For each sync workflow that includes at least one synchronization step, you can configure multiple alerts. Then, when a sync workflow run completes, the recipients signed up for the alert receive an email message informing them about the completion of the sync workflow run. For example, you can use sync workflow alerts to inform recipients when a sync workflow run completes with errors.

To manage alerts for a sync workflow, navigate to the **Sync Workflows** tab in the Synchronization Service Console, and then click the **Manage alerts** link below the sync workflow.

To manage outgoing mail profiles for sending sync workflow alerts, in the Synchronization Service Console, click the **Settings** menu in the upper right corner, and then click the **Mail Profiles**.

Creating or editing a sync workflow alert

You can create or edit email-based alerts for existing sync workflows, allowing you to send notifications about key synchronization events, such as completing sync workflow runs or detecting errors.

To create or edit an alert

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the **Manage alerts** link below the sync workflow for which you want to create or edit an alert.

   NOTE: The **Manage alerts** link is only available on sync workflows that include one or more synchronization steps.
3. In the **Manage Sync Workflow Alerts** dialog, do one of the following:
   - If you want to create a new alert, click **Add** under the **Sync workflow alerts** list.
   - If you want to edit an existing alert, select that alert in the **Sync workflow alerts** list, and then click **Edit** under the list.
4. To specify alert settings, use the following options in the dialog that opens, and click **OK**:
   - **When this event occurs**: Select an event that will trigger the alert. You can select one of the following:
     - **Sync workflow run completes (with or without errors)**: Triggers the alert upon the sync workflow run completion regardless of any errors encountered in the run.
     - **Sync workflow run completes with errors**: Triggers the alert only when the sync workflow run completed with errors.
   - **Send email to**: Enter the email addresses of the recipients to which you want to send a notification email message when the selected event occurs. When specifying multiple email addresses, use a semicolon as a separator.
   - **Email message subject**: Enter the text you want to include in the subject of the notification email.
   - **Ignore mapping errors**: Select this check box if you want the alert to skip mapping errors in sync workflow runs. This check box is only available when you select **Sync workflow run completes with errors** in the **When this event occurs** option.
   - **Ignore non-fatal errors in**: Select this check box if you want this alert to skip non-fatal errors in sync workflow runs. A non-fatal error causes a sync workflow run to partially succeed. A fatal error causes a sync workflow run to fail. If you select this check box, you must also select one of the following options:
   - **All sync workflow steps**: Causes the alert to skip non-fatal errors in all steps of the sync workflow.
   - **The specified sync workflow steps**: Causes the alert to skip non-fatal errors only in the sync workflow steps you specify. To specify multiple steps, either enter the step numbers separated by commas (for example, `1, 3, 5`), or specify a range of steps using dash as a separator (for example, `1, 3, 5-8`).

> NOTE: This check box is only available if you select **Sync workflow run completes with errors** in the **When this event occurs** option.

5. Use the **Send email using this outgoing mail profile** list to select the settings to be used for sending notification emails generated by the alerts in the **Sync workflow alerts** list.

   To configure the current outgoing mail profile, click **Properties**. For more information, see Managing outgoing mail profiles.

6. When you are finished, click **OK** to close the **Manage Sync Workflow Alerts** dialog.

Deleting a sync workflow alert

You can delete existing sync workflow alerts in the Synchronization Service Console. This is useful for housekeeping purposes, for example when a sync workflow is modified, and its original alarms are no longer applicable.

To delete an alert

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click the **Manage alerts** link below the sync workflow for which you want to delete an alert.

   > NOTE: The **Manage alerts** link is only available on sync workflows that include one or more synchronization steps.

3. In the **Sync workflow alerts** list, select the alert you want to delete, and then click **Delete** under the list.

Managing outgoing mail profiles

To create, edit, or delete an outgoing mail profile, in the Synchronization Service Console, click **Settings** > **Mail Profiles** in the upper-right corner. Then, follow the appropriate procedure below.

To create a profile

1. Click **Add** below the list of profiles, then specify the settings you want to use. For the descriptions of the settings you can specify, see Outgoing mail profile settings.
2. When you are finished, click **OK**.

To edit a profile

1. In the list, select the outgoing mail profile you want to edit.
2. Click **Edit** below the list of profiles, then specify the settings you want to use. For the description of the settings you can specify, see Outgoing mail profile settings.
3. When you are finished, click **OK**.

To delete a profile

1. In the list, select the outgoing mail profile you want to delete.
2. Click the **Delete** button below the list of profiles.

Outgoing mail profile settings

In each outgoing mail profile, you can use the following settings:

- **Profile name**: Enter a descriptive name with which you want to identify the profile.
- **Outgoing SMTP server**: Enter the fully qualified domain name of the SMTP mail server you want to use for sending notification emails.
- **This server requires an encrypted connection (SSL)**: Select this check box if the specified mail server requires an encrypted connection.
- **This server requires authentication**: Select this check box if the specified mail server requires authentication, then specify the user name and password with which you want to access the server.
- **Sender email address**: Specify the email address you want to use as the originating address in the notification emails.
- **Sender name**: Specify the sender name you want to display in the **From** field to the recipients of the notification emails.

6

Mapping objects

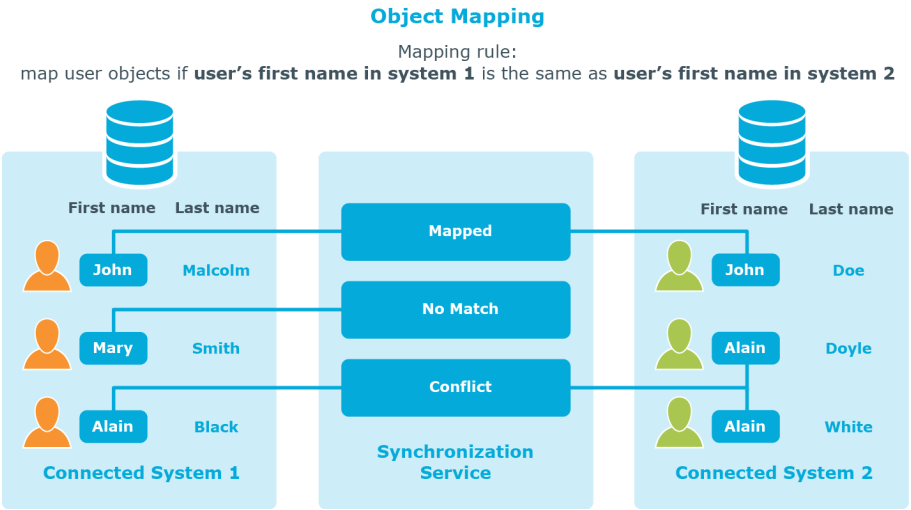
Object mapping allows you to establish one-to-one relationships between objects in two connected data systems. By using object mapping, you can determine what objects will participate in data synchronization operations you run between these two data systems.

Synchronization Service maps objects automatically when running the creating steps of a sync workflow. In this case, a one-to-one relationship is automatically established between source objects and their counterparts created in the target connected system during the creation operation. In some cases, however, you may have to manually map objects. For example, you should configure object mapping before running a sync workflow that includes updating or deprovisioning steps. By doing so, you provide Synchronization Service with the information on which objects need to be updated or deprovisioned in the target data system.

To map objects, you can use mapping pairs and mapping rules. A mapping pair allows you to establish a relationship between a certain object type in one connected system and its counterpart in the other connected system. A mapping rule allows you to define the scope of conditions where the objects belonging to the object types specified in a particular mapping pair will be mapped. You can create multiple mapping rules for a mapping pair, with each mapping rule defining a specific mapping condition. You have to run your mapping rules for them to take effect. After you run a mapping rule, Synchronization Service reads data in the connected data systems for which the rule is configured, and then maps the objects that meet the conditions specified in the mapping rule.

The following example shows how a mapping rule works:

Figure 12: Object mapping



In this example, one-to-one relationship is established between the user object **John Malcolm** in **Connected System 1** and the user object **John Doe** in **Connected System 2**: the first names of these user objects match, and thus the condition specified in the mapping rule is met. Now, if you configure a sync workflow for these systems and populate it with synchronization steps, identity information will be synchronized between these two user objects, since they are mapped. The direction of synchronization depends on which of these two connected data systems acts as the synchronization source and which is the target.

How to map objects

You can map objects in two data systems to which Synchronization Service is connected.

Creating mapping pairs

In this step, you create mapping pairs that specify the types of objects you want to map in two connected systems. You can create as many mapping pairs as required.

To create a mapping pair

1. In the Synchronization Service Console, open **Mapping**.
2. Click the name of the connection for which you want to map objects.
3. Click **Add mapping pair**.
4. On the **Specify source** page, next to **Connected system object type**, click **Select**, and then select the type of object you want to map.

5. Click **Next**.
6. On the **Specify target** page:
   a. Next to **Target connected system**, click **Specify**, and then specify the other connected system where you want to map objects.
   b. Next to **Connected system object type**, click **Select**, and then select the type of object you want to map.
7. To create the mapping pair, click **Finish**.

   Repeat the above steps to create mapping pairs for as many object types as required.

Creating mapping rules

Once you have created a mapping pair, you can configure mapping rules for that pair. Mapping rules define the conditions where the objects that belong to the object types specified in the mapping pair will be mapped. Synchronization Service maps objects only if all mapping rules specified for a mapping pair are met.

To add a new mapping rule

1. In the Synchronization Service Console, open **Mapping**.
2. Click the name of the connection for which you want to create a mapping rule.
3. Click the mapping pair for which you want to create a mapping rule.
4. Click **Add mapping rule**.
5. Use the **Define Mapping Rule** dialog to define the condition where the objects in the connected systems are to be mapped. To do so, click the down arrow on the button next to each of the two provided options and select one of the following:
   - **Attribute**: Allows you to select an attribute in the connected system.
   - **Rule**: Allows you to set up a list of rules to generate a value for the connected system. For more information, see Using value generation rules.
   - **PowerShell Script**: Allows you to type a Windows PowerShell script that generates a value for the connected system.
6. When you are finished, click **OK** to create the mapping rule.

Change scope for mapping rules

Each mapping rule applies to a scope of objects. By default, this scope includes all objects that belong to the object types specified in the mapping rule. If necessary, you can narrow the scope specified for a particular mapping rule or you can revert to the default scope.

To change the scope of a mapping rule

1. In the Synchronization Service Console, open **Mapping**.
2. Click the name of the appropriate connection.
3. Click the appropriate mapping pair entry.
4. Locate the mapping rule whose scope you want to change. Use the following elements provided for each mapping rule entry:
   - **Mapping scope for system 1**: Shows the mapping rule scope applicable to the data system shown on the left part of the mapping pair entry.
   - **Mapping scope for system 2**: Shows the mapping rule scope applicable to the data system shown on the right part of the mapping pair entry.

   These elements can take one of the following values:
   - **Default**: Indicates that the mapping rule applies to all objects of the specified type.
   - **Custom**: Indicates that the mapping rule scope is narrowed down and only applies to some objects of the specified type.
5. Change the mapping rule scope as necessary:
   a. Click the value displayed next to **Mapping scope for system 1** or **Mapping scope for system 2**, and then specify the scope you want to use.
   b. When you are finished, click **OK**.

Running map operation

Once you have created mapping rules for a mapping pair, you have to run the map operation in order to apply these rules and map objects that belong to the mapping pair.

There are two methods to run the map operation:

- Running the map operation once, manually.
- Creating a recurring schedule to automatically run the map operation on a regular basis.

  TIP: This method is recommended when you want to use Synchronization Service to synchronize passwords from an Active Directory domain to other connected systems.

Running the map operation once, manually

This method allows you to run your mapping rules without creating a recurring schedule.

To run the map operation once, manually

1. In the Synchronization Service Console, open **Mapping**.
2. Click the name of the connection for which you want to run the map operation.
3. Click the mapping pair for which you want to run the map operation.
4. Click **Map now.**
5. In the dialog that opens, click one of the following:
   - **Full Map**: With this option, Synchronization Service retrieves the data required to map objects from the connected data systems.
   - **Quick Map**: With this option, Synchronization Service first tries to map objects by using the data that is available in the local cache. If the local cache is missing or cannot be used to map objects, then Synchronization Service retrieves the required data from the connected data systems.

   Wait for the map operation to complete.

   After the map operation is completed, the Synchronization Service Console displays a report that provides information about the objects that participated in the map operation. At this stage, the application does not map the objects. To map the objects, you have to commit the map operation result.

   You can click the number that is provided next to an object category name in the report to view the details of objects that belong to that category.
6. Review the report about the objects that participated in the map operation, then click **Commit** to map the objects.

Creating a recurring schedule to automatically run the map operation on a regular basis

Running mapping rules on a recurring schedule allows you to properly map newly-created Active Directory user objects to their counterparts in the connected systems where you automatically synchronize passwords with the Active Directory domain. If you do not run mapping rules on a regular basis, some passwords may become out of sync due to the changes that inevitably occur to your environment. For example, new user objects are created, some user objects are deleted, but Synchronization Service cannot detect these changes and synchronize passwords for the newly-created users before you apply the mapping rules. In this scenario, the best way to ensure Synchronization Service synchronizes all passwords is creating a recurring schedule for applying your mapping rules on a regular basis.

To create a recurring schedule to automatically run the map operation on a regular basis

1. In the Synchronization Service Console, open **Mapping**.
2. Click the name of the connection for which you want to create a recurring mapping schedule.

3. Click the mapping pair for which you want to run the map operation on a recurring schedule.
4. Click **Schedule mapping**.
5. In the dialog that opens, select the **Schedule the task to run** check box, and then specify a schedule for the map operation.

   One Identity recommends that you schedule the map operation to run once in every 6 hours.
6. If several Synchronization Service instances are installed in your environment, under **Run the task on**, select the computer that hosts the instance you want to use for running the map operation.
7. Click **OK** to activate the schedule.

   The results of a scheduled map operation always apply automatically. As a result, you do not have to commit the changes after the scheduled operation is completed.

   When performing a scheduled map operation, Synchronization Service always retrieves the required data from the connected data systems and never uses the data available in the local cache.

How to unmap objects

You can unmap the objects that were mapped earlier.

To unmap objects

1. In the Synchronization Service Console, open **Mapping**.
2. Click the name of the connection for which you want to unmap objects.
3. Click the mapping pair that specifies the objects types you want to unmap.
4. Click **Unmap now** and wait until the unmap operation completes.

   After the unmap operation is completed, the Synchronization Service Console displays a report that provides information about the objects that participated in the unmap operation. At this stage, the application does not unmap the objects. To unmap them, you need to commit the result of the unmap operation.

   To view the details of objects that belong to a given object category, you can click the number provided next to the object category name in the report.
5. Review the report on the objects that participated in the unmap operation, and then click **Commit** to unmap the objects.

7

Automated password synchronization

If your enterprise environment has multiple data management systems, each having its own password policy and dedicated user authentication mechanism, you may face one or more of the following issues:

- Because users have to remember multiple passwords, they may have difficulty managing them. Some users may even write down their passwords. As a result, passwords can be easily compromised.
- Each time users forget one or several of their numerous access passwords, they have to ask administrators for password resets. This increases operational costs and translates into a loss of productivity.
- There is no way to implement a single password policy for all of the data management systems. This too impacts productivity, as users have to log on to each data management system separately in order to change their passwords.

With Synchronization Service, you can eliminate these issues and significantly simplify password management in an enterprise environment that includes multiple data management systems.

Synchronization Service provides a cost-effective and efficient way to synchronize user passwords from an Active Directory domain to other data systems used in your organization. As a result, users can access other data management systems using their Active Directory domain password. Whenever a user password is changed in the source Active Directory domain, this change is immediately and automatically propagated to other data systems, so each user password remains in sync in the data systems at all times.

You need to connect Synchronization Service to the data systems in which you want to synchronize passwords through special connectors supplied with Synchronization Service.

How to automate password synchronization

To automatically synchronize passwords from an Active Directory domain to another data system, complete these steps:

1. Install Capture Agent on each domain controller in the Active Directory domain you want to be the source for password synchronization operations.

   Capture Agent tracks changes to the user passwords in the source Active Directory domain and provides this information to Synchronization Service, which in turn synchronizes passwords in the target connected systems you specify.

   For more information on how to install Capture Agent, see Managing Capture Agent.

2. Connect the Synchronization Service to the Active Directory domain where you installed Capture Agent.

   Alternatively, you can configure a connection to Active Roles Synchronization Service that manages the source Active Directory domain.

3. Connect the Synchronization Service to the data system where you want to synchronize user object passwords with those in the source Active Directory domain.
   - For some target data systems (such as SQL Server) you must specify the data you want to participate in the password synchronization by configuring an SQL query.
   - If the target data system is an LDAP directory service accessed via the generic LDAP connector, you must specify the target object type for which you want to synchronize passwords and the attribute where you want to store object passwords.

4. Ensure that user objects in the source Active Directory domain are properly mapped to their counterparts in the target connected system.

   For more information about mapping objects, see Mapping objects.

   Synchronization Service automatically maps objects between the source Active Directory domain and the target connected system if you configure sync workflows to manage the creation and deprovision operations between the source Active Directory domain (or Active Roles Synchronization Service that manages that domain) and the target connected system.

   For more information on sync workflows, see Synchronizing identity data.

5. Create a password synchronization rule for the target connected system.

   For more information, see Creating a password sync rule.

After you complete the above steps, the Synchronization Service starts to automatically track user password changes in the source Active Directory domain and synchronize passwords in the target connected system.

If necessary, you can fine-tune the password synchronization settings by completing these optional tasks:

- Modify the default Capture Agent settings.

  For more information, see Configuring Capture Agent.
- Modify the default Synchronization Service settings related to password synchronization.

  For more information, see Configuring Synchronization Service.
- Specify a custom certificate for encrypting the password sync traffic between the Capture Agent and the Synchronization Service. By default, a built-in certificate is used for this purpose.

  For more information, see Specifying a custom certificate for encrypting password sync traffic.
- Configure the Synchronization Service to automatically run your PowerShell script after the password synchronization is completed.

  For more information, see Using PowerShell scripts with password synchronization.

Managing Capture Agent

Capture Agent is required to track changes to the user passwords in the Active Directory domain you want to be the authoritative source for password synchronization operations. To synchronize passwords, you must install Capture Agent on each domain controller in the source Active Directory domain.

Whenever a password changes in the source Active Directory domain, the agent captures that change and provides the changed password to the Synchronization Service. In turn, the Synchronization Service uses the provided information to synchronize passwords in the target connected systems according to your settings.

Installing Capture Agent manually

You can manually deploy Synchronization Service Capture Agent on each domain controller in the source Active Directory domain. Alternatively, you can also perform an unattended installation for the Capture Agent component.

To manually install Capture Agent

1. On the domain controller, open the Active Roles Synchronization Service installation media.
2. In the installation media, navigate to the following directory: `\Tools\Sync Service Capture Agent`
3. Run `SyncServiceCaptureAgent_8.2_x64.msi`.
4. Follow the instructions of the setup wizard.
5. After installing Capture Agent, restart the domain controller.

To perform an unattended installation

1. On the domain controller, open the Windows Command Prompt.
2. Run the following command:

```
msiexec /i "<path-to-SyncServiceCaptureAgent_8.2_x64.msi>" /qb
INSTALLDIR="<path-to-installation-folder>" REBOOT="<reboot-value>"
```

These commands use the following arguments:

- (Optional) **`INSTALLDIR`**: Specifies the Capture Agent installation folder. If this argument is not used, the Capture Agent component is installed to the following default folder:

  `%ProgramFiles%\One Identity\Active Roles\8.2\SyncServiceCaptureAgent`

- **`REBOOT`**: Specifies whether to force or suppress restart after installation with the following available values:
  - **`Force`**: Prompts to restart the system after installation.
  - **`Suppress`**: Suppresses restart prompts after installation.
  - **`ReallySuppress`**: Suppresses all restart prompts and restart attempts during installation.

  NOTE: After installing Capture Agent, restart the domain controller.

  For more information on these values, see REBOOT property in the *Microsoft Windows Installer documentation*.

Using Group Policy to install Capture Agent

You can use this method to automatically deploy Capture Agent on each domain controller in the source Active Directory domain. This method is applicable in the following scenarios only:

Table 106: Prerequisites by scenario

Supported scenario	Prerequisites
Scenario 1: AD domain includes either 32- or 64-bit domain controllers	• All the domain controllers must be held in a single organizational unit (for example, the built-in **Domain Controllers** OU). • At least one group policy object must be linked to the OU holding the domain controllers (for example, the built-in **Default Domain Controllers Policy Group Policy** object).
Scenario 2: AD domain includes both 32-bit and 64-bit domain controllers	• The domain controllers must be held in two separate organizational units, each containing domain controllers of the same bitness.

Supported scenario	Prerequisites
	• At least one group policy object must be linked to each of the two Organizational Units.

To install Capture Agent by using Group Policy

1. Save the `SyncServiceCaptureAgent_8.2_x86.msi` and `SyncServiceCaptureAgent_8.2_x64.msi` files to a network share accessible from each domain controller in the source Active Directory domain.
2. Depending on your scenario, complete the steps in the table:

Table 107: Steps by scenario

Scenario 1: AD domain includes either 32-bit or 64-bit domain controllers	Scenario 2: AD domain includes both 32-bit and 64-bit domain controllers
1. Use Group Policy Editor to open the group policy object linked to the OU holding the domain controllers on which you want to install Capture Agent. 2. In the Group Policy Object Editor console tree, do one of the following: • In Windows Server 2016 or later, expand the **Computer Configuration** node, then expand **Policies**, and select **Software Settings**. 3. In the details pane, click **Software Installation**, on the **Action** menu point to **New**, and then click **Package**. 4. Use the dialog to open one of the following files: • `SyncServiceCaptureAgent_8.2_x86.msi` if all your domain controllers are 32-bit. • `SyncServiceCaptureAgent_8.2_x64.msi` if all your domain controllers are 64-	1. Use Group Policy Object Editor to open the group policy object linked to the OU holding the 32-bit domain controllers. 2. Do one of the following in the Group Policy Object Editor console tree: • In Windows Server 2016 or later, expand the **Computer Configuration** node, then expand **Policies**, and select **Software Settings**. 3. In the details pane, click **Software Installation**, on the **Action** menu point to **New**, and then click **Package**. 4. Use the dialog to open the `SyncServiceCaptureAgent_8.2_x86.msi` file. 5. In the **Deploy Software** dialog, select **Assigned**, and then click **OK**. 6. Repeat every step for the group policy object linked to the OU holding the 64-bit domain controllers. Use the `SyncServiceCaptureAgent_8.2_`

Scenario 1: AD domain includes either 32-bit or 64-bit domain controllers	Scenario 2: AD domain includes both 32-bit and 64-bit domain controllers
bit. 5. In the **Deploy Software** dialog, select **Assigned**, and then click **OK**.	`x64.msi` file to install Capture Agent on these domain controllers.

3. Run the following command at a command prompt to refresh the Group Policy settings:

   ```
   gpupdate /force
   ```

Uninstalling Capture Agent

You can delete the Active Roles Synchronization Service Capture Agent component with the built-in tools of the operating system.

To uninstall Capture Agent

1. On the computer where Capture Agent is installed, open the list of installed programs.
2. In the list of installed programs, select **One Identity Active Roles 8.2 - Synchronization Service Capture Agent x64** or **One Identity Active Roles 8.2 - Synchronization Service Capture Agent x86**.
3. To delete Capture Agent, click **Uninstall**.
4. Follow the on-screen instructions.

Managing password sync rules

To synchronize passwords from an Active Directory domain to other connected systems, you need to create and configure a password synchronization rule for each target connected system where you want to synchronize passwords.

A password synchronization rule allows you to specify the following:

- The Active Directory domain you want to be the source for password synchronization operations.
- The source object type for password synchronization operations (typically, this is the user object type in Active Directory).
- The target connected system in which you want to synchronize passwords with the

source Active Directory domain.

- The target object type for password synchronization operations.

Optionally, you can configure a password synchronization rule to modify attribute values of the target connected system objects whose passwords are being synchronized.

Creating a password sync rule

To create a password sync rule

1. In the Synchronization Service Console, open the **Password Sync** tab.
2. Click **Add password sync rule**.
3. On the **Specify source for password sync** page, do the following:
   a. In the **Source connected system** option, specify the Active Directory domain you want to be the source for password synchronization operations. Alternatively, you can select the Active Roles Synchronization Service instance that manages such an Active Roles domain.
   b. In the **Connected system object type** option, select the object type you want to be the source for password synchronization.
4. Click **Next**.
5. On the **Specify target for password sync** page, do the following:
   a. In the **Target connected system** option, specify the target connected system in which you want to synchronize passwords.
   b. In the **Connected system object type** option, select the object type you want to be the target for password synchronization.
   c. Optionally, you can click **Password Sync Settings** and then use the following tabs to configure more password sync settings:
      - **Password Sync Retry Options**: Use this tab to specify how many times you want Synchronization Service to retry the password synchronization operation in the event of a password synchronization failure. You can select one of the following options:
        - **Unlimited number of times**: Causes Synchronization Service to retry the password synchronization operation until it succeeds.
        - **This maximum number of times**: Specify the maximum number of times you want Synchronization Service to retry the password synchronization operation.
      - **Password Transformation Script**: Use this tab to type a PowerShell script that transforms source Active Directory user passwords into object passwords for the target connected system. Use this item if you want the object passwords in the source and target connected systems to be

different. If you do not want to transform passwords, leave the text box blank.

- **Rules to Modify Object Attributes**: Use this tab to specify rules for modifying attribute values on the target connected system objects. These rules will only apply to the objects on which Synchronization Service modifies passwords in the target connected system.

d. When you are finished, click **OK**.

6. Click **Finish** to create the password sync rule.

Deleting a password sync rule

To delete a password sync rule

1. In the Synchronization Service Console, open the **Password Sync** tab.
2. Locate the rule you want to delete, and then click **Delete this rule** below the rule.

Modifying settings for a password sync rule

You can modify the following settings of an existing password sync rule:

- Specify how many times you want the Synchronization Service to retry the password synchronization operation in the case of a password synchronization failure.
- Specify a PowerShell script to transform a source Active Directory user password into an object password in the target connected system.
- Specify rules to modify the attributes of the target connected system objects on which Synchronization Service changes passwords.

To modify the settings of a password sync rule

1. In the Synchronization Service Console, open the **Password Sync** tab.
2. Click the **Password sync settings** link below the password sync rule you want to modify.
3. In the dialog that opens, use the following tabs:
   - **Password Sync Retry Options**: Use this tab to specify how many times you want Synchronization Service to retry the password synchronization operation in the event of a password synchronization failure. You can select one of the following options:
     - **Unlimited number of times**: Causes Synchronization Service to retry the password synchronization operation until it succeeds.

    - **This maximum number of times**: Specify the maximum number of times you want Synchronization Service to retry the password synchronization operation.
  - **Password Transformation Script**: Use this tab to type a PowerShell script that transforms source Active Directory user passwords into object passwords for the target connected system. Use this item if you want the object passwords in the source and target connected systems to be different. If you do not want to transform passwords, leave the text box blank.
  - **Rules to Modify Object Attributes**: Use this tab to specify rules for modifying attribute values on the target connected system objects. These rules will only apply to the objects on which Synchronization Service modifies passwords in the target connected system.

4. When you are finished, click **OK** to save your changes.

Fine-tuning automated password synchronization

This section provides information about the optional tasks related to configuring the automated password synchronization from an Active Directory domain to connected data systems.

Configuring Capture Agent

Capture Agent has a number of parameters you can modify. After you install the agent, each of these parameters is assigned a default value, as described in the following table:

Table 108: Capture Agent parameters

Parameter	Description	Default value
Maximum connection point validity for Capture Agent Service	Determines the period of time (in hours) during which a connection between Capture Agent and Synchronization Service remains valid.	24 hours
Interval between connection retries	Determines the time interval (in minutes) during which Capture Agent tries to reconnect to Synchronization Service.	10 minutes

Parameter	Description	Default value
Maximum duration of a connection attempt	Determines the period of time (in days) during which Capture Agent tries to connect to Synchronization Service to send the information about changed user passwords. During this period Capture Agent stores the user passwords to be synchronized in an encrypted file.	7 days
Certificate to encrypt Capture Agent traffic	Specifies a certificate for encrypting the password sync data transferred between Capture Agent and Synchronization Service. For more information, see Specifying a custom certificate for encrypting password sync traffic.	By default, a built-in certificate is used.
Connection Point 1 **Connection Point 2** **Connection Point 3** **Connection Point 4** **Connection Point 5** **Connection Point 6** **Connection Point 7**	Define the Synchronization Service instances to which Capture Agent provides information about changed user passwords.	If none of these parameters is set, Capture Agent looks for available instances of the Synchronization Service in the following container: `CN=Active Roles Sync Service,CN=One Identity,CN=System,DC=<domain name>`

You can modify the default values of these parameters by using Group Policy and the Administrative Template supplied with the Synchronization Service. The next steps assume that all the domain controllers where the Capture Agent is installed are held within organizational units.

Creating and linking a Group Policy object

Create a new Group Policy object. Link the object to each organizational unit holding the domain controllers on which the Capture Agent is installed. For more information, see the documentation for your version of the Windows operating system.

Adding an administrative template to Group Policy object

1. Use Group Policy Object Editor to connect to the Group Policy object you created previously.
2. In the Group Policy Object Editor console, expand the Group Policy object, and then do one of the following:
   - In Windows Server 2016 or later, expand **Computer Configuration**, expand **Policies**, and then select **Administrative Templates**.
3. On the **Action** menu, point to **All Tasks**, and click **Add/Remove Templates**.

   The **Add/Remove Templates** dialog opens.
4. In the **Add/Remove Templates** dialog, click **Add**, and then use the **Policy Templates** dialog to open the Administrative Template (`SyncServiceCaptureAgent.adm` file) supplied with the Synchronization Service.

   The `SyncServiceCaptureAgent.adm` file is located in the following folder of the installation media:

   `\Tools\Sync Service Capture Agent.`

Using Group Policy object to modify Capture Agent settings

1. In Windows Server 2016 or later, under **Computer Configuration** > **Policies** > **Administrative Templates** > **Classic Administrative Templates (ADM)** > **Active Roles**, select **Sync Service Capture Agent Settings**.
2. In the details pane, configure the appropriate Group Policy settings.

   The names of Group Policy settings correspond to the names of the Capture Agent parameters provided in the table in Configuring Capture Agent.
3. Run the following command at a command prompt for the changes to take effect:

   ```
   gpupdate /force
   ```

Modifying Synchronization Service parameters

You can modify the default values of the Synchronization Service parameters related to password synchronization. These parameters and their default values are described in the next table.

Table 109: Synchronization Service parameters

Parameter	Description	Default Value
Interval between attempts to reset password	The Capture Agent sends information on changes made to Active Directory user passwords to Synchronization Service. After receiving this information, Synchronization Service tries to reset passwords in the target connected systems you specified. This parameter determines the time interval (in minutes) between attempts to reset passwords in the target connected systems.	10 minutes
Synchronization Service connection point update period	Synchronization Service publishes its connection point in Active Directory. This parameter determines the frequency of updates (in minutes) of the Synchronization Service connection point.	60 minutes
Certificate to encrypt Capture Agent traffic	This parameter specifies the thumbprint of the certificate used to encrypt the password sync traffic between Capture Agent and Synchronization Service. The same certificate must be used for the Capture Agent and the Synchronization Service.	By default, a built-in certificate is used.

You can modify the Synchronization Service parameters using Group Policy and the Administrative Template supplied with Synchronization Service.

To modify Synchronization Service parameters using Group Policy

1. On the computer running the Synchronization Service, start Group Policy Object Editor, and then connect to the **Local Computer Policy** Group Policy object.
2. In the Group Policy Object Editor console, expand the **Local Computer Policy** node, expand the **Computer Configuration** node, and select **Administrative Templates**.
3. On the **Action** menu, point to **All Tasks**, and click **Add/Remove Templates**.
4. In the **Add/Remove Templates** dialog, click **Add**, and then use the **Policy Templates** dialog to open the `SyncService.adm` file that holds the Administrative Template.

   By default, the `SyncService.adm` file is stored in the following subfolder of the Active Roles Synchronization Service installation:

   `\SyncService\Administrative Templates`
5. Under **Computer Configuration** > **Administrative Templates** > **Active Roles**, select **Sync Service Settings**, and then in the details pane, configure the

appropriate group policy settings.

The names of group policy settings correspond to the names of the Synchronization Service parameters provided in the table in Configuring Capture Agent.

6. For the changes to take effect, refresh the Group Policy settings by running the following command at a command prompt:

```
gpupdate /force
```

Specifying a custom certificate for encrypting password sync traffic

By default, Synchronization Service uses a built-in certificate to encrypt password sync traffic between the Capture Agent and the Synchronization Service. If necessary, you can use a custom certificate for this purpose.

NOTE: Consider the following when specifying a custom certificate for encrypting password sync traffic:

- SSL certificates signed with MD5 algorithm are not supported.
- Backward compatibility for Quick Connect v5.5 with Active Roles Synchronization Service Capture Agent can be achieved through custom certificate signed with SHA algorithm.

This section illustrates how to use a custom certificate for encrypting the password synchronization traffic in Windows Server 2012.

Obtaining and installing a certificate

To obtain and install a certificate, you have to make a certificate request. There are two methods to request a certificate in Windows Server 2012:

- **Request certificates using the Certificate Request Wizard**: To request certificates from a Windows Server 2012 enterprise certification authority, you can use the Certificate Request Wizard.
- **Request certificates using the Windows Server 2012 Certificate Services web interface**: Each certification authority that is installed on a computer running Windows Server 2012 has a web interface that allows the users to submit certificate requests. By default, the web interface is accessible at `http://servername/certsrv`, where `servername` refers to the name of the computer running Windows Server 2012.

This section provides steps to request certificates using the Windows Server 2012 Certificate Services web interface. For detailed information about the Certificate Request Wizard, refer to the documentation on Certification Authority.

To request a certificate using the Windows 2012 Certificate Services web interface

1. Use a web browser to open `http://servername/certsrv`, where `servername` refers to the name of the web server running Windows Server 2012 where the certification authority that you want to access is located.
2. On the **Welcome** Web page, click **Request a certificate**.
3. On the **Request a Certificate** page, click **advanced certificate request**.
4. On the **Advanced Certificate Request** page, click **Create and submit a certificate request to this CA**.
5. On the page that opens, do the following:
   - Select the **Store certificate in the local computer certificate store** check box.
   - Under **Additional Options**, select the **PKCS10** option, and in the **Friendly Name** text box, specify a name for your certificate (for example, `My QC Certificate`).

   Keep default values for all other options.
6. Click **Submit**.
7. On the **Certificate Issued** page, click **Install this certificate**.

After you install the certificate, it becomes available in the **Certificates** snap-in, in the **Personal** > **Certificates** store.

Exporting the custom certificate to a file

In this step, you export the issued certificate to a file. You will need the file to install the certificate on each domain controller running Capture Agent and on each computer running Synchronization Service.

To export the certificate

1. On the computer where you installed the certificate in the **Obtaining and installing a certificate** step, open the **Certificates - Local Computers** snap-in.
2. In the Console tree, expand the **Personal** > **Certificates** store.
3. In the details pane, click the issued certificate you want to export.
4. On the **Action** menu, point to **All Tasks**, and then click **Export**.
5. Step through the wizard.
6. On the **Export Private Key** page, select **Yes, export the private key**, and then click **Next**.

   This option is available only if the private key is marked as exportable and you have access to the private key.
7. On the **Export File Format** page, do the following, and then click **Next**:

- To include all certificates in the certification path, select the **Include all certificates in the certification path if possible** check box.
- To enable strong protection, select the **Enable strong protection (requires IE 5.0, NT 4.0 SP4 or above)** check box.

8. On the **Password** page, in the **Password** text box, type a password to encrypt the private key you are exporting. In **Confirm password**, type the same password again, and then click **Next**.
9. On the **File to Export** page, use the **File name** text box to specify the PKCS #12 file to which you want to export the certificate along with the private key, and click **Next**.
10. On the **Completion** page, revise the specified settings and click **Finish** to create the file and close the wizard.

Importing certificate into certificates store

In this step, you import the certificate to the **Personal\Certificates** certificate store by using the **Certificates** snap-in. You must complete this step on each domain controller running Capture Agent and on each computer running Synchronization Service that will participate in the password synchronization.

To import the certificate

1. Open the **Certificates - Local Computers** snap-in.
2. In the Console tree, click the **Personal** > **Certificates** logical store.
3. On the **Action** menu, point to **All Tasks** and then click **Import**.
4. Step through the wizard.
5. On the **File to Import** page, in **File name**, type the file name containing the certificate to be imported or click **Browse** and to locate and select the file. When finished, click **Next**.
6. On the **Password** page, type the password used to encrypt the private key, and then click **Next**.
7. On the **Certificate Store** page, ensure that the **Place all certificates in the following store** option is selected, and the **Certificate store** text box displays **Personal**, and then click **Next**.
8. On the **Completion** page, revise the specified settings and click **Finish** to import the certificate and close the wizard.

Copying the certificate's thumbprint

In this step, you copy the thumbprint of your custom certificate. In the next steps, you will need to provide the thumbprint to Capture Agent and Synchronization Service.

To copy the thumbprint of your custom certificate

1. Open the **Certificates - Local Computer** snap-in.
2. In the Console tree, click the **Personal** store to expand it.
3. Click the **Certificates** store to expand it.
4. In the details pane, double-click the certificate.
5. In the **Certificate** dialog, click the **Details** tab, and scroll through the list of fields to select **Thumbprint**.
6. Copy the hexadecimal value of **Thumbprint** to the clipboard.

You will need the copied thumbprint value to configure the Capture Agent and Synchronization Service.

Providing the certificate's thumbprint to Capture Agent

This step assumes that:

- The same Group Policy object is linked to each OU holding the domain controllers on which the Capture Agent is installed. For more information on how to create and link a Group policy object, see the documentation for your version of Windows.
- The `SyncServiceCaptureAgent.adm` administrative template file is linked to that Group Policy object.

For instructions on how to add an administrative template file to a Group Policy object, see Adding an administrative template to Group Policy object.

To provide the thumbprint to Capture Agent

On any computer joined to the domain where Capture Agent is installed, open Group Policy Object Editor, and connect to the Group Policy object to which you added the Administrative Template in Adding an administrative template to Group Policy object.

1. In the Group Policy Object Editor console, expand the Group Policy object, and then expand the **Computer Configuration** node.
2. Expand the **Administrative Templates** > **Active Roles** node to select **Sync Service Capture Agent Settings**.
3. In the details pane, double-click **Certificate to encrypt Capture Agent traffic**.
4. Select the **Enabled** option, and then paste the certificate's thumbprint (the one you copied in Copying the certificate's thumbprint) in the **Thumbprint** text box. When finished, click **OK**.
5. For the changes to take effect, refresh the Group Policy settings by running the following command at a command prompt:

   ```
   gpupdate /force
   ```

Providing the certificate's thumbprint to Synchronization Service

Perform the next steps on each computer running the Synchronization Service that participates in the password sync operations.

To provide the thumbprint to Synchronization Service

1. On the computer running the Synchronization Service, start Group Policy Object Editor, and then connect to the **Local Computer Policy** Group Policy object.
2. In the Group Policy Object Editor console, expand the **Local Computer Policy** node, expand the **Computer Configuration** node, and select **Administrative Templates**.
3. On the **Action** menu, point to **All Tasks**, and click **Add/Remove Templates**.
4. In the **Add/Remove Templates** dialog, click **Add**, and then use the **Policy Templates** dialog to open the `SyncService.adm` file that holds the Administrative Template.
5. By default, the `SyncService.adm` file is stored in `<Active Roles installation folder>\SyncServiceCaptureAgent\Administrative Templates.`
6. Under **Computer Configuration** > **Administrative Templates** > **Active Roles**, select **Sync Service Settings**.
7. In the details pane, double-click **Certificate to encrypt Capture Agent traffic**.
8. Select the **Enabled** option, and then paste the certificate's thumbprint (the one you copied in Copying the certificate's thumbprint) in the **Thumbprint** text box. When finished, click **OK**.
9. For the changes to take effect, refresh the Group Policy settings by running the following command at a command prompt:

   ```
   gpupdate /force
   ```

Using PowerShell scripts with password synchronization

Optionally, you can configure the Synchronization Service to run your custom PowerShell script before, after, or instead of the password synchronization operation. To do so, create a connection handler. For instructions, see Using connection handlers.

Example of a PowerShell script run after password synchronization

After the password synchronization is complete, the following script sends a notification email message informing the administrator that the specified object password has been modified in the target connected system. The message provides the names of the source Active Directory object and its counterpart in the target connected system.

```
#---- Specify the SMTP Server name in your organization ----
$SmtpServer = "smtpServerName"
$smtp = new-object system.net.mail.smtpClient($SmtpServer)
$mail = new-object System.Net.Mail.MailMessage
# ---- Set the sender mail ----
$mail.From = "yourmail@mydomain.com"
# ---- Set the destination mail ----
$mail.To.Add("Administrator@mydomain.com")
# --- Specify the message subject ----
$mail.Subject = "Password was changed"
# ---- Set the message text ----
$body  = "The passwords were synchronized for the following object pair: "
$body  = $body + $srcObj.Name + "->" + $dstObj.Name
$mail.Body = $body
# ---- Send mail ----
$smtp.Send($mail)
```

8

Synchronization history

Synchronization Service Console provides the Synchronization History option that allows you to view the details of completed synchronization workflow runs, password synchronization rule runs, and map and unmap operations.

The synchronization history also helps you troubleshoot synchronization issues by providing information on the errors that were encountered during sync workflow runs, password sync rule runs, or map and unmap operations.

You can also selectively clean up entries from the synchronization history.

To access the synchronization history, use the **Sync History** tab in the Synchronization Service Console.

Viewing sync workflow history

You can use the **Sync History** tab in the Synchronization Service Console to view a list of completed sync workflow runs.

This list provides information on:

- The names of completed synchronization workflows.
- The dates when each sync workflow run started and completed.
- Which Synchronization Service instance was used to run each synchronization workflow.

You can click a sync workflow run entry in the list to view detailed information about the sync workflow steps that were run, objects that participated in that run, and errors encountered during the run, if any.

To view the details of a completed sync workflow run

1. In the Synchronization Service Console, click the **Sync History** tab.
2. Click **Sync Workflow History**.
3. If you want to filter the list of completed sync workflows, use the following elements:

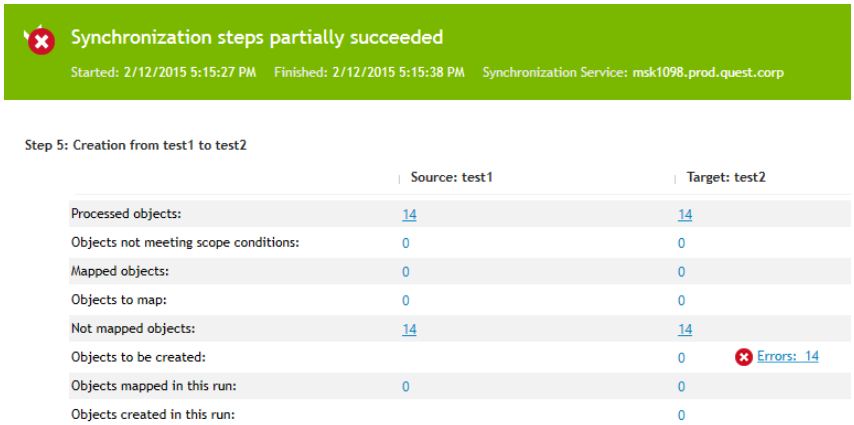
- **Show items completed**: Use this element to specify the time period when the sync workflows you want to view completed.
- **Maximum number of items to show**: Specify the maximum number of completed sync workflows you want to view.

You can sort the list of completed sync workflows by clicking the column titles in the list. Also you can filter the list of completed sync workflows by typing keywords in the text boxes provided below the column titles.

4. To view detailed information about a list item, select the list item and after that click **Details**.

   The details provided for each list entry look similar to the following:

Figure 13: Synchronization Servce details



   To view detailed information about the objects that belong to a certain object category, click the number displayed next to the object category name in the **Source** or **Target** column.

   To view detailed information about encountered errors, click the link displaying the number of errors.

View mapping history

You can use the **Sync History** tab in the Synchronization Service Console to view the detailed information about a particular completed map or unmap operation. By doing so, you can view a list of attributes for each object that participated in the map or unmap operation.

To view the details of a mapped pair of objects

1. In the Synchronization Service Console, click the **Sync History** tab.
2. Click **Mapping History**.

3. If you want to filter the list of completed map and unmap operations, use the following elements:
    - **Show items completed**: Specify a time period when the map and unmap operations you want to view completed.
    - **Maximum number of items to show**: Specify the maximum number of completed map and unmap operations you want to view.

    You can sort the list of map and unmap operations by clicking the column titles. Also you can filter the list of map and unmap operations by typing keywords in the text boxes provided below the column titles.
4. To view detailed information about a list item, select the list item and after that click **Details**.

Searching synchronization history

You can use the **Sync History** tab in the Synchronization Service Console to search for completed creation, deprovision, update, and sync passwords operations in the synchronization history.

You can search by:

- The target connected system on which the operation was run.
- The type of object that participated in the operation.
- The period when the operation completed.

To search the synchronization history for completed operations

1. In the Synchronization Service Console, click the **Sync History** tab.
2. Click **Search**.
3. Use the following options to specify your search criteria:
    - **Target connection**: Select the connected system for which you want to search for completed creation, deprovision, update, and sync passwords operations.
    - **Object type**: Select the object type for which you want to search for completed creation, deprovision, update, and sync passwords operations.
    - **Show items completed**: Specify the time period when the operation you want to search for completed.
    - **Maximum number of items to show**: Specify the maximum number of completed creation, deprovision, update, and sync passwords operations you want to view in the list.

    You can sort the search results by clicking the column titles in the search results list. Also, you can filter the search results by typing keywords in the text boxes provided below the column titles.

4. To view detailed information about a list item, select the list item and after that click **Details**.

Cleaning up synchronization history

You can selectively delete entries from the sync workflow history and object mapping history. To delete entries, you can either run the cleanup operation once or you can define a schedule to run the cleanup operation on a regular basis.

To run the cleanup operation once

1. In the Synchronization Service Console, click the **Sync History** tab.
2. Click **Clean up now**.
3. Specify the entries you want to delete.
4. Click **OK** to delete the entries from the synchronization history.

To create a recurring schedule for the cleanup operation

1. In the Synchronization Service Console, click the **Sync History** tab.
2. Click **Schedule cleanup**.
3. In the dialog that opens, select the **Schedule the task to run** check box, and then specify a schedule for the cleanup operation.
4. If several Synchronization Service instances are deployed in your environment, under **Run the task on**, select the computer that hosts the instance you want to use for running the cleanup operation.
5. To activate the schedule, click **OK**.

To disable a scheduled cleanup operation

1. In the Synchronization Service Console, click the **Sync History** tab.
2. Click **Schedule cleanup**.
3. In the dialog that opens, clear the **Schedule the task to run** check box, and then click **OK**.

9

Scenarios of use

This section provides some use case scenarios that help you familiarize yourself with Synchronization Service. The scenarios illustrate how to create and run sync workflows and their steps to update and create user information from a Human Resources (HR) database represented by a delimited text file to an Active Directory domain.

The scenarios are:

- Scenario: Creating users from a `.csv` file to an Active Directory domain. In this scenario, Synchronization Service creates user accounts from a Comma Separated Values (`.csv`) file that includes a HR database to individual Organizational Units in an Active Directory domain, depending on the city where each user is based.
- Scenario: Using a `.csv` file to update user accounts in an Active Directory domain. In this scenario, Synchronization Service updates user accounts in an Active Directory domain based on the changes made to the HR database saved in a Comma Separated Values (`.csv`) file.
- Scenario: Synchronizing data between One Identity Manager Custom Target Systems and an Active Directory domain. In this scenario, Quick Connect updates data in One Identity Manager based on the changes made in Active Directory domain.
- Scenario: Deprovisioning between One Identity Manager Custom Target Systems and an Active Directory domain. In this scenario, Quick connect deprovisioning synchronized objects in One Identity Manager processed from the Active Directory domain.
- Scenario: Provisioning of Groups between One Identity Manager Custom Target Systems and an Active Directory domain. In this scenario, Quick Connect provisions group objects to be synchronized to One Identity Manager from Active Directory domain.
- Scenario: Enabling Delta Sync mode between One Identity Manager Custom Target Systems and an Active Directory domain. In this scenario, Quick Connect updates data in One Identity Manager based on the changes made in Active Directory domain in the delta sync mode.

Before you proceed with these sample scenarios, perform the following steps:

1. Make sure you have properly configured the connection to the target Active Directory domain in the Synchronization Service Console.

2. Create the **Employees** Organizational Unit (OU) at the root of the target Active Directory domain.
3. In the **Employees** OU, create the following OUs:
    - **New York**
    - **Tokyo**
    - **Amsterdam**
    - **OtherCities**

Scenario: Create users from a .csv file to an Active Directory domain

The following scenario demonstrates how to create user accounts from a Human Resources (HR) database to an Active Directory domain. The HR database is represented by a sample Comma Separated Values (`.csv`) file. Depending on the user city, accounts will be created in one of the following OUs:

- **Employees\New York**
- **Employees\Tokyo**
- **Employees\Amsterdam**
- **Employees\OtherCities**

TIP: You can use the Active Directory Users and Computers tool to ensure that Synchronization Service has created user accounts in the **Employees** OU. The **New York**, **Tokyo**, **Amsterdam**, and **OtherCities** OUs may include some disabled user accounts created by Synchronization Service.

Creating a sync workflow

To create a sync workflow

1. In the Synchronization Service Console, open the **Sync Workflows** tab.
2. Click **Add sync workflow**.
3. In the **Sync workflow name** text box, type a name for the sync workflow being created.
4. Click **OK**.

   The new workflow appears on the **Sync Workflows** tab.

   NOTE: After you created a sync workflow, you must populate it with one or more synchronization steps. For more information, see Synchronizing identity data.

Adding a creating step

This section provides instructions on how to:

- Connect Synchronization Service to the source Comma Separated Values (`.csv`) file and target Active Directory domain.
- Add a new creating step and configure its settings, for example, specify the object attributes to create.
- Develop a Windows PowerShell script that returns the name of an Active Directory container for created user accounts.
- Preview a list of user accounts to be created.

To add a creating step

1. In the Synchronization Service Console, open the **Sync Workflows** tab, and then click the sync workflow you created in the Creating a sync workflow step.
2. Click **Add synchronization step**.
3. On the **Select an action** page, select **Creation**, and then click **Next**.
4. On the **Specify source and criteria** page, click **Specify**, click **Add new connected system**, and then step through the wizard to add the sample Comma Separated Values (`.csv`) file as a connected system:
   a. Use the **Connection name** box to type a descriptive name for the connection being created.
   b. In the **Use the specified connector** list, select **Delimited Text File Connector**. Click **Next**.
   c. Click **Browse** to locate and select the sample Comma Separated Values (`.csv`) file supplied with Synchronization Service. This file is located in the `<Synchronization Service installation folder>\Samples` folder.
   d. Step through the wizard until you are on the **Specify attributes to identify objects** page.
   e. In the **Available attributes** list, select **Employee ID**, click **Add**, and then click **Finish**.
5. Click **Next**.
6. On the **Specify target** page, click **Specify**, and then step through the wizard to add the target Active Directory domain as a connected system:
   a. Use the **Connection name** box to type a descriptive name for the connection being created.
   b. In the **Use the specified connector** list, select **Active Directory Connector**. Click **Next**.
   c. Use the **Domain name** field to type the FQDN name of the target Active

Directory domain. If necessary, adjust other connection settings on this page as appropriate. Click **Finish**.

7. Click the down arrow on the button provided next to the **Target container** option.
8. In the provided list, click **PowerShell Script**.
9. Insert the following script sample into the dialog, and then click **OK**:

```
$userCity = $srcObj["City"]
 switch ($userCity)
{
  "New York" {$container = "OU=New York,OU=Employees,DC=mycompany,DC=com";
break}
  "Amsterdam" {$container =
"OU=Amsterdam,OU=Employees,DC=mycompany,DC=com"; break}
  "Tokyo" {$container = "OU=Tokyo,OU=Employees,DC=mycompany,DC=com";
break}
  default {$container = "OU=OtherCities,OU=Employees,DC=mycompany,DC=com";
break}
}
$container
```

NOTE: Before using the script, change the `DC=mycompany",DC=com` string as appropriate to reflect your environment. For example, if you have created the **Employees** OU in the `testlab.ttt` domain, use the following string: `DC=testlab,DC=ttt`.

10. Click the down arrow on the leftmost button provided below the **Rules to generate unique object name** list.
11. In the provided list, click **Attribute**.
12. Select **Logon Name**, and then click **OK**. Click **Next**.
13. Expand **Initial Attribute Population Rules**, and then create forward sync rules to synchronize the following pairs of attributes:

Table 110: Initial attribute population rules

CSV file attribute	Synchronization direction	Active Directory attribute
Logon Name	=>	Logon Name (Pre-Windows 2000)
First Name	=>	First Name
Last Name	=>	Last Name
City	=>	City

For more information on how to create rules, see Modifying attribute values by using rules.

14. Expand **Initial Password**, click **Text**, and type a password in the **Set Password** dialog. Click **OK**.
15. (Optional) To modify the default options to create new user accounts, expand **User Account Options**.
16. Click **Finish** to close the wizard.

Running the configured creating step

To run the creating step

1. On the **Sync Workflows** tab, click **Run now**.
2. In the **Select sync workflow steps to run** dialog, select the check box next to the step you created, and then to run the step, click **Full Run**.

   After the synchronization step run completes, the Synchronization Service Console displays a report that provides information about the objects that participated in the creating step. At this stage, the application does not commit changes to the target Active Directory domain.

TIP: To view a list of user accounts to be created in the **Employees** OU, click the number next to **Objects to be created**.

Committing changes to Active Directory

To commit changes to the target Active Directory domain

- Click **Commit**.

Scenario: Using a .csv file to update user accounts in an Active Directory domain

This scenario demonstrates how to update user accounts in an Active Directory domain when the information on employees is changed in the Human Resource (HR) database held in a Comma Separated Values (`.csv`) file.

NOTE: This scenario can be used only if the **Employees** OU already contains user accounts created with the creating scenario described earlier in this document. Only accounts for previously created employees will be updated.

Creating an updating step

This section explains how to create a step that updates user accounts from the HR database to the target Active Directory domain.

To add an updating step to your existing sync workflow

1. In the Synchronization Service Console, open the **Sync Workflows** tab, and then click the sync workflow you created in the Creating a sync workflow step.
2. Click **Add synchronization step**.
3. On the **Select an action** page, select **Update**, and then click **Next**.
4. On the **Specify source and criteria** page, do the following:
   a. Click **Specify**, click **Select existing connected system**, and then select the Comma Separated Values (`.csv`) file you connected in Scenario: Create users from a .csv file to an Active Directory domain. Click **Finish**.
   b. Make sure that the object type specified in the **Source object type** box is `csv-Object`.
5. Click **Next**.
6. On the **Specify target** page, do the following:
   a. Click **Specify**, and then select the Active Directory domain you connected in Scenario: Create users from a .csv file to an Active Directory domain.
   b. Make sure that the object type specified in the **Target object type** box is `User (user)`.
7. Click **Next**.
8. Expand **Rules to Modify Object Attributes**, and then create forward sync rules to synchronize the following pairs of attributes:

   Table 111: Rules to modify object attributes

CSV file attribute	Synchronization direction	Active Directory attribute
City	=>	City
Department	=>	Department
First Name	=>	First Name
Last Name	=>	Last Name
Telephone Number	=>	Telephone Number

   For information on how to create rules, see Modifying attribute values by using rules.
9. Click **Finish**.

Running the configured updating step

To run the updating step

1. On the **Sync Workflows** tab, click **Run now**.
2. In the **Select sync workflow steps to run** dialog, select the check box next to the step you created, and then click **OK** to run the step.

   After the synchronization step run completes, the Synchronization Service Console displays a report that provides information about the objects that participated in the updating step. At this stage, the application does not commit changes to the target Active Directory domain.

TIP: To view a list of user accounts to be updated in the **Employees** OU, in the update report, click the number next to **Objects to be updated**.

Committing changes to Active Directory

To commit changes to the target Active Directory domain

- Click **Commit**.

Scenario: Synchronizing data between One Identity Manager Custom Target Systems and an Active Directory domain

Out of the box, Synchronization Service includes the One Identity Manager connector, which allows you to access the One Identity Manager. In this scenario, the basic purpose for the Quick Connect One Identity Manager connector is to use the connector for target systems where there is no existing native One Identity Manager connector.

Administrators can create or configure multiple Custom Target Systems in One Identity Manager. Each Target System has entities such as User Accounts, Groups, Container Structure, and so on.

NOTE: One Identity Manager does not have any specific table space for target systems that do not have a native One Identity Manager connector. The data synchronized is placed in the One Identity Manager tablespace where the tables starts with `UNS..` and end with `B`, referred as `UNS..B` tables.

The following scenario shows how to use the Quick Connect One Identity Manager Connector to synchronize data between One Identity Manager Custom Target Systems and an Active Directory domain.

Creating a connection to One Identity Manager

To create a new connection to One Identity Manager

1. In the Synchronization Service Console, open the **Connections** tab.
2. Click **Add connection**, then use the following options:
   - **Connection name**: Type a descriptive name for the connection.
   - **Use the specified connector**: Select **One Identity Manager Connector**.
3. Click **Next**.
4. On the **Specify connection settings** page, use the following options:
   - **Application Server URL**: Specify the address of the One Identity Manager application server to which you want to connect.
   - **Authentication module**: Identifies the One Identity Manager authentication module that is to be used to verify the connection's user ID and password.
   - **User name**: Specify the user ID for this connection.
   - **Password**: Specify the password of the user ID for this connection.
   - To test the connection with the new parameters, click **Test connection**.
5. Click **Next**.

Configuring One Identity Manager modules, Custom Target System and Container Information

NOTE: The One Identity Manager target systems and One Identity Manager containers are applicable only for the Target System Base module.

To select the One Identity Manager modules, Target Systems, and Containers

1. Select the required One Identity Manager modules.
2. Select **Target System Base module** to synchronize data to One Identity Manager custom target systems (`UNS..B` tables). This enables you to select the target object types such as `UNSAccountB`, `UNSGroupB`, and so on.
3. Select the required One Identity Manager target system, for example, **Azure**.
4. Select the required One Identity Manager container, for example, **Test AD**.
5. Click **Finish** to create a connection to **One Identity Manager**.

Creating a workflow for provisioning

To create a workflow for provisioning data synchronization to One Identity Manager

1. Start the Synchronization Service Console.
2. Open the **Sync Workflows** tab, and then click **Add sync workflow**.
3. Type a descriptive name, for example, `AD to OneIM Sync` for the workflow being created, and then click **OK** to create the workflow.

Creating a provisioning step

To create a provisioning step

1. In the Synchronization Service Console, open the **Sync Workflows** tab, and then click the **AD to OneIM Sync** workflow.
2. Click **Add synchronization step**.
3. On the **Select an action** page, select **Creation**, and then click **Next**.
4. In the **Specify source and criteria** dialog, click **Specify**, click **Add new connected system** or **Select existing connected system**, and then step through the wizard to add the Active Directory **Test AD** as a connected system.
5. Click **Next**.
6. In the **Specify target** dialog, click **Specify.**
7. Click **Add new connected system** or **Select existing connected system**, and then step through the wizard to add the target One Identity Manager domain as a connected system.
8. Click **Select**, to add the required target object type.
9. In the **Select Object Type** dialog, select the **UNSAccountB** object type from the list of object types and click **OK**.

Specifying synchronization rules

To specify the synchronization rules:

1. In the Synchronization Service Console, open the **Sync Workflows** tab, and then click the **AD to OneIM Sync** workflow.
2. Click **Provision from Test AD to One Identity Manager Connection**.
3. Click **Provisioning Rules** and then click **Initial Attribute Population Rules**.

4. From the drop-down, select **Forward Sync Rule**.
5. In the **Forward Sync Rule** dialog, select the source attributes to be mapped to the target attributes, and then click **OK**.

   NOTE: For One Identity Manager workflows, the attribute configuration rule for **CN** is mandatory, else a constraint violation error is displayed and the workflow run fails.
6. Click **Save and Continue**.

Running the workflow

To run the provisioning step:

1. On the **Sync Workflows** tab, click **Run now**.
2. In the **Select sync workflow steps to run** dialog, select the check box next to the step you created, and then to run the step, click **Full Run**.

   After the synchronization step run completes, the Synchronization Service Console displays a report that provides information about the objects that participated in the provisioning step. At this stage, the application does not commit changes to the target One Identity Manager domain.

Committing changes to One Identity Manager

To commit the changes to One Identity Manager

- Click **Commit**.

An `All changes committed` message is displayed. The changes are committed from the source Active Directory **Test AD** to the target One Identity Manager.

Verify on One Identity Manager

To verify if the data is synchronized to One Identity Manager

1. Open the Synchronization Service Console.
2. Verify that all the users from the AD are synchronized with One Identity Manager as per the provisioning rules that were set.

Scenario: Deprovisioning between One Identity Manager Custom Target Systems and an Active Directory domain

The deprovision operation in data synchronization using Synchronization Service allows you to modify or remove objects in the target data system after their counterparts have been disconnected from the source data system. You can configure Synchronization Service to remove target objects permanently or change them to a specific state. To specify the objects that will participate in the deprovision operation you can use object mapping rules. This scenario describes how to create a deprovisioning step for a workflow to modify or delete the synchronized objects in the target system based on the deprovisioning criteria that is set.

To create a deprovisioning step

1. In the Synchronization Service Console, open the **Sync Workflows** tab, and then click the **AD to OneIM Sync** workflow.
2. Click **Add synchronization step**.
3. In the **Select an action** dialog, select **Deprovision**, and then click **Next**.
4. In the **Specify source and criteria** dialog, click **Specify**, click **Add new connected system** or **Select existing connected system**, and then step through the wizard to add the Active Directory **Test AD** as a connected system.
5. Specify a deprovisioning criteria by selecting one of the following:
   - **Source object is deleted or out of synchronization scope**
   - **Source object deprovisioning is initiated in connected system**
   - **Source object meets these criteria - Add the criteria for the source objects to be deprovisioned in the target system**
6. Click **Next**.
7. In the **Specify target** dialog, click **Specify**.
8. Click **Add new connected system** or **Select existing connected system**, and then step through the wizard to add the target One Identity Manager domain as a connected system.
9. Click **Select**, to add the required target object type.
10. In the **Select Object Type** dialog, select the **UNSAccountB** object type from the list of object types and click **OK**.
11. In the **Specify deprovisioning action** dialog, select the one of the following action to deprovision:

- **Delete target objects**
- **Initiate the object deprovisioning in <target system>**
- **Modify target objects - Click Forward Synch rule and select the attributes to modify the object attributes**

12. Click **Next**.

    Active Roles Synchronization Service then creates the deprovisioning step with the rules for the specified deprovisioning action.

Scenario: Provisioning of groups between One Identity Manager Custom Target Systems and an Active Directory domain

Synchronization Service allows you to ensure that group membership information is in sync in all connected data systems. For example, when provisioning a group object from an Active Directory domain to One Identity Manager domain, you can configure rules to synchronize the **Member** attribute from the source to the target domain.

This scenario describes how to create a provisioning step for a workflow to synchronize group objects between the source and target systems.

To create a group provisioning step:

1. In the Synchronization Service Console, open the **Sync Workflows** tab, and then click the **AD to OneIM Sync** workflow.
2. Click **Add synchronization step**.
3. In the **Select an action** dialog, select **Creation**, then click **Next**.
4. In the **Specify source and criteria** dialog, click **Specify**, click **Add new connected system** or **Select existing connected system**, then progress through the wizard to add the Active Directory **Test AD** as a connected system.
5. In **Specify object type** field, click **Select** and from the **Select Object type** list, select **Group**, then click **OK**.
6. In the **Provisioning Criteria** section, click **Add**.
7. In the **Select Container** dialog, from the containers list, select the required container and click **OK**.
8. Click **Next**.
9. In the **Specify target** dialog, click **Specify**.

10. Click **Add new connected system** or **Select existing connected system**, and then step through the wizard to add the target One Identity Manager domain as a connected system.
11. Click **Select**, to add the required target object type.
12. In the **Select Object Type** dialog, select the `UNSGroupB` object type from the list of object types.
13. Click **OK**.

    Active Roles Synchronization Service then creates the group provisioning step.

Scenario: Enabling Delta Sync mode between One Identity Manager Custom Target Systems and an Active Directory domain

The Delta processing mode of the Synchronization Service allows you to synchronize identities between the source and the target systems for only the data that has changed in the source and target connected systems since their last synchronization.

This scenario describes how to enable the delta processing mode between the source (Active Directory domain) and target (One Identity Manager) systems.

To enable the delta processing mode

1. Create a sync workflow for provisioning data synchronization between the source (Active Directory) and target (One Identity Manager) system.
2. Add a creating step for the workflow to provision users from the source system to target system.
3. Click on the synchronization step for provision of users.
4. On the **General Options** tab, specify the delta process mode:
   a. Under **Source Connected System**, select **Process delta from last run**.
   b. Under **Target Connected System** select **Process delta from last run**.
5. Click **Save and continue**.

   NOTE: Before any data has been processed from the source to the target system, the initial synchronization of data is always performed in the **Process all delta** mode.
6. Run the configured creating step.

   The data for the users added or updated to the source since the previous run, is displayed under **Processed Objects**.

Example of using the Generic SCIM Connector for data synchronization

Once you configured a connection with the **Generic SCIM Connector** as described in Configuring the Generic SCIM Connector for Starling Connect connections, you can configure import-based data synchronization tasks to import data from the SCIM-based SuccessFactors HR and ServiceNow connectors of Starling Connect to another target system supported by Active Roles Synchronization Service.

Creating such a SCIM-based sync workflow has two main steps:

1. Mapping objects by configuring one or more **mapping pairs** and **mapping rules**. By mapping objects, you can specify logic checks by which Active Roles Synchronization Service can identify if two data entries stored in two separate databases are the same or not.
   - With mapping pairs, you can establish a relationship between object types in two connected systems.
   - With mapping rules, you can define the conditions on how the objects specified in the mapping pair will be mapped during synchronization.

   Example: Mapping objects by user ID

   You can use object mapping, for example, to identify the same data entries between a SuccessFactors HR database (connected to Active Roles Synchronization Service via a **Generic SCIM Connector** connection) and an SQL server (connected to Active Roles Synchronization Service via a **Microsoft SQL Server Connector**).

   To do so, you can set up a mapping that compares the User ID value of the data entries in the two systems. If the data entries in the two systems share the same User ID, Active Roles Synchronization Service will consider them the same.

   For more information on object mapping, see Mapping objects. For an example mapping procedure using the **Generic SCIM Connector**, see Creating object mapping between a SCIM connection and an SQL connection.

2. Setting up a sync workflow based on the configured object mapping, so that you can automate creating, removing or deprovisioning specific data entries between the connected systems.

   For more information on sync workflows, see Synchronizing identity data. For an example workflow configuration procedure using the **Generic SCIM Connector**, see Creating a sync workflow for synchronizing data from a SCIM-based Starling Connect connector.

The following chapters will provide an example for setting up a sync workflow that will import data from a SuccessFactors HR database via a **Generic SCIM Connector** connection, and synchronizing that data to an SQL database.

Creating object mapping between a SCIM connection and an SQL connection

Once you configured a connection with the **Generic SCIM Connector** as described in Configuring the Generic SCIM Connector for Starling Connect connections, you can configure import-based data synchronization tasks to import data from the SCIM-based SuccessFactors HR and ServiceNow connectors of Starling Connect to another target system supported by Active Roles Synchronization Service.

The first step of creating this synchronization is mapping objects between the SCIM-based source system and a target system, so that Active Roles Synchronization Service can detect identical data entries between the two system for proper data synchronization.

By mapping objects, you can specify logic checks by which Active Roles Synchronization Service can identify if two data entries stored in two separate databases are the same or not.

- With mapping pairs, you can establish a relationship between object types in two connected systems.
- With mapping rules, you can define the conditions on how the objects specified in the mapping pair will be mapped during synchronization.

The following example procedures show how to create a mapping pair and a mapping rule between:

- A SuccessFactors HR database connected to Active Roles Synchronization Service with the **Generic SCIM Connector**. The SuccessFactors HR database will be the source system from which Active Roles Synchronization Service imports the data.
- An SQL database connected to Active Roles Synchronization Service with the **Microsoft SQL Server Connector**. The SQL database will act as the target system to which Active Roles Synchronization Service will synchronize the SuccessFactors HR data.

Prerequisites

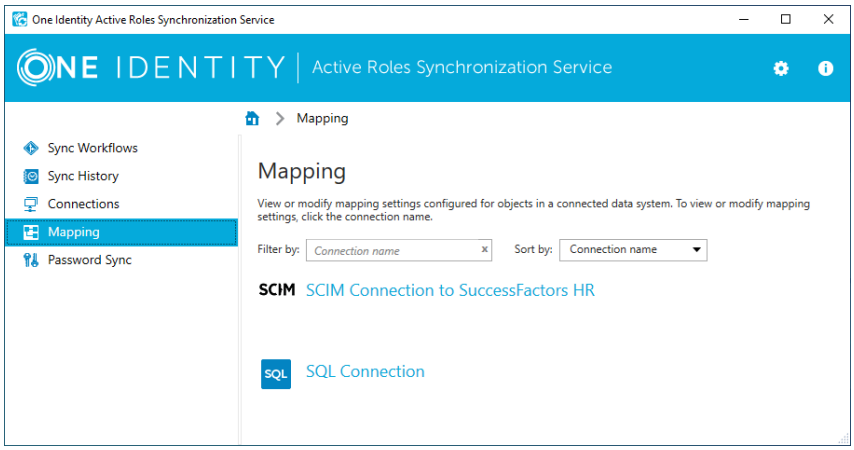
You can perform the following procedures only if Active Roles Synchronization Service already contains the following working connectors:

- A **Generic SCIM Connector** connecting Active Roles Synchronization Service to the Starling Connect SuccessFactors HR connector. To configure such a connection, see Configuring the Generic SCIM Connector for Starling Connect connections. In this example procedure, this connection is called **SCIM Connection to SuccessFactors HR**.
- A **Microsoft SQL Server Connector** providing connection to the SQL Server used in this example. To configure such a connection, see Creating a Microsoft SQL Server connection. In this example, this connection is called **SQL Connection**.

To configure a mapping pair between a SuccessFactors HR database and an SQL database

1. In Active Roles Synchronization Service, navigate to **Mapping**, then click the **SCIM Connection to SuccessFactors HR** connection.

   Figure 14: Active Roles Synchronization Service – Selecting a connector for mapping objects

2. To start configuring a new object mapping with the **Add mapping pair** dialog, click **Add mapping pair**.
3. In the **Specify source** step, under **Connected system object type**, select the resource object type you want the object mapping to check. In this example, we are using the `Employees` data entry of the SuccessFactors HR database, so click **Select**, then in the **Select Object Type** step, select **Employees**.

   TIP: If the data entry is hard to find due to the length of the list, use the **Filter by name** field to find it quicker.

   To apply your selection, click **OK**, then **Next**.

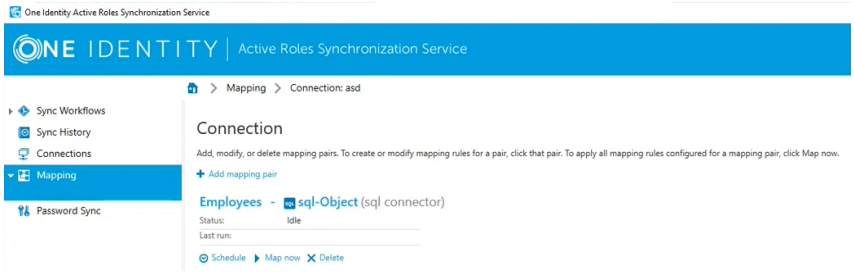
4. In the **Specify target** step, under **Target connected system**, configure the target system where the other resource object type is located. To do so, click **Specify**, and in the **Add Connected System Wizard**, select the **Select existing connected system** option, then the connector of the SQL server (in this example, **SQL Connection**). To apply your selection, click **Finish**.
5. Under **Connected system object type**, select **sql-Object**.
6. To create the mapping pair, click **Finish**.
7. (Optional) If needed, you can configure additional mapping pairs as well for your sync workflow. To do so, click **Add mapping pair** again, and repeat the procedure. This example procedure uses only one mapping pair.

Once the mapping pair is created, you can configure its associated mapping rule.

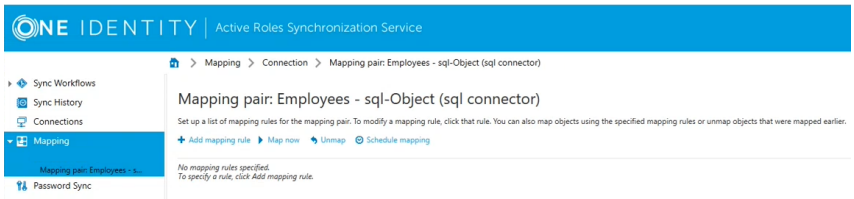
To configure a mapping rule between a SuccessFactors HR database and an SQL database

1. In Active Roles Synchronization Service, navigate to **Mapping**, then click the **SCIM Connection to SuccessFactors HR** connection.
2. The previously configured mapping pair appears. To open the available mapping pair settings, click the **Employees** object type in the mapping pair.

Figure 15: Active Roles Synchronization Service – Mapping pair in a configured SCIM connection



3. To start configuring a new mapping rule, in the **Mapping pair** window, click **Add mapping rule**.



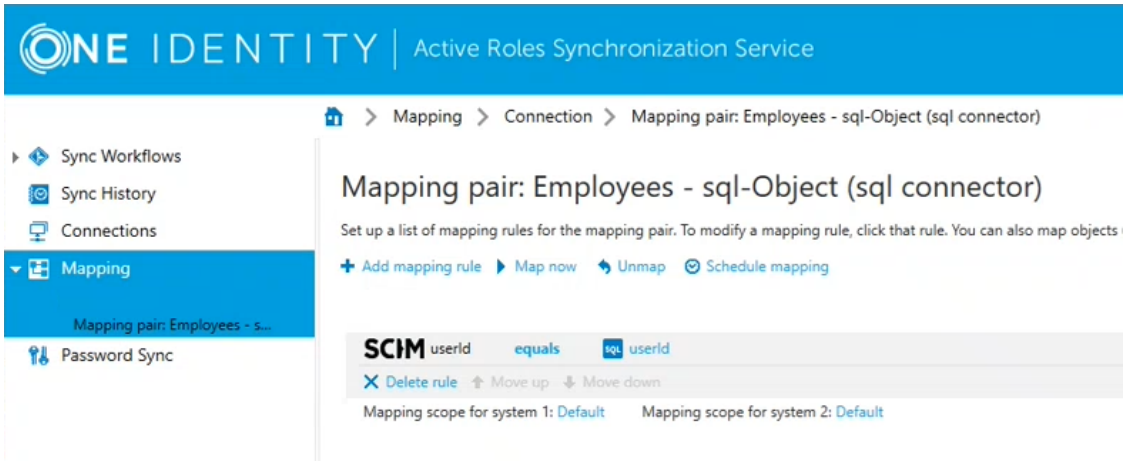
4. In the **Define Mapping Rule** window, specify the source and target resource object types that must be equal so that Active Roles Synchronization Service can map the data pairs. In this example, we are using the `UserID` attribute for this purpose both in the SuccessFactors HR database and in the SQL database as well.

Therefore, at the **Value generated for SCIM Connection to SuccessFactors HR by using** field, click **Attribute**, then in the **Select attribute** window, select **userId**. This adds the `userId` object value to both the source and target fields.

> TIP: If the data entry is hard to find due to the length of the list, use the **Filter by name** field to find it quicker.

5. To finish adding the mapping rule, click **OK**.

Figure 16: Active Roles Synchronization Service – Mapping rule in a configured SCIM mapping pair



6. To start the mapping synchronization based on the configured value pair of the mapping rule, click **Map now**. Active Roles Synchronization Service offers two mapping types:
   - **Quick Map**, using local cached data to speed up the mapping process.
   - **Full Map**, retrieving data from the source and target data system for accuracy.

   As this is the first time of running this mapping, perform a **Full Map**.

Once the mapping rule finishes running successfully, it will indicate the unmapped, changed and mapped objects, along with the objects that do not meet the scope conditions of the configured mapping rule.

Creating a sync workflow for synchronizing data from a SCIM-based Starling Connect connector

Once you configured a connection with the **Generic SCIM Connector** as described in Configuring the Generic SCIM Connector for Starling Connect connections, you can configure import-based data synchronization tasks to import data from the SCIM-based SuccessFactors HR and ServiceNow connectors of Starling Connect to another target system supported by Active Roles Synchronization Service.

The second step of creating this synchronization task is setting up a sync workflow based on the object mapping configured in Creating object mapping between a SCIM connection and an SQL connection. By configuring a workflow, you can automate **creating**, **removing** or **deprovisioning** specific data entries between the connected systems.

The following example procedure shows how to create a workflow that creates and updates data synchronization between:

- A SuccessFactors HR database connected to Active Roles Synchronization Service with the **Generic SCIM Connector**. The SuccessFactors HR database will be the source system from which Active Roles Synchronization Service imports the data.
- An SQL database connected to Active Roles Synchronization Service with the **Microsoft SQL Server Connector**. The SQL database will act as the target system to which Active Roles Synchronization Service will synchronize the SuccessFactors HR data.

Prerequisites

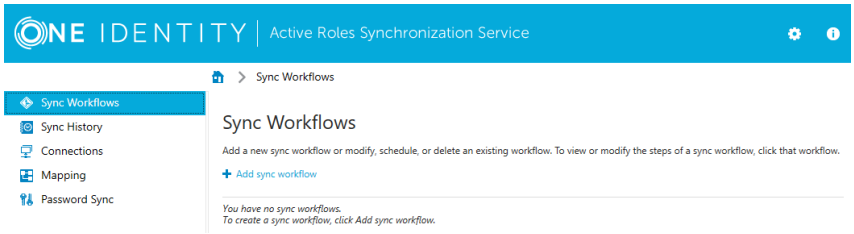
Before performing the procedure, make sure that the following conditions are met:

- Active Roles Synchronization Service must already contain the following working connectors:
  - A **Generic SCIM Connector** connecting Active Roles Synchronization Service to the Starling Connect SuccessFactors HR connector. To configure such a connection, see Configuring the Generic SCIM Connector for Starling Connect connections. In this example procedure, this connection is called **SCIM Connection to SuccessFactors HR**.
  - A **Microsoft SQL Server Connector** providing connection to the SQL Server used in this example. To configure such a connection, see Creating a Microsoft SQL Server connection. In this example, this connection is called **SQL Connection**.
- The mapping pair and mapping rule configured in Creating object mapping between a SCIM connection and an SQL connection are active and working.

To configure a data sync workflow between a SuccessFactors HR database and an SQL database

1. In Active Roles Synchronization Service, click **Sync Workflows** > **Add sync workflow**.

Figure 17: Active Roles Active Roles Synchronization Service – Adding a new sync workflow

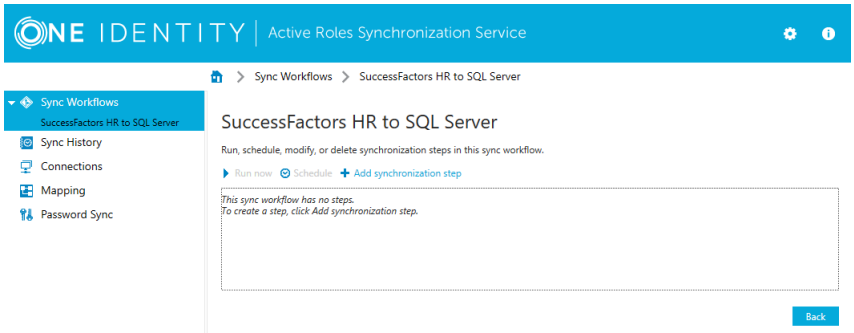


2. In the **Sync workflow name** step, name the workflow (for example, **SuccessFactors HR to SQL Server**), then click **OK**.

   The new workflow then appears in the **Sync Workflows** tab.

3. Configure a data synchronization creation step for the workflow. To do so, in **Sync Workflows**, click the name of the workflow (in this example, **SuccessFactors HR to SQL Server**), then click **Add synchronization step**.

Figure 18: Active Roles Synchronization Service – Adding a new synchronization step



4. In the **Select an action** step, select **Creation**, then click **Next**.

   The **Creation** step of the workflow will be used to create the synchronized data entries of the SuccessFactors HR database in the target SQL database. The **Creation** step performs data synchronization only for data entries that do not exist in the target system. Because of this, you typically run this step only once.

5. In the **Specify source and criteria** step, configure the following settings:

   - **Source connected system**: Specify the SuccessFactors HR database connection here, created with the **Generic SCIM Connector**. To do so, click **Specify** > **Select existing connected system**, then select the SCIM-based connection (in this example, **SCIM Connection to SuccessFactors HR**).
   - **Source object type**: Specify the source object type here (in this example, the `Employees` object type). To do so, click **Select**, then in the **Select Object Type** window, select **Employees**, and click **OK**.

> TIP: If the data entry is hard to find due to the length of the list, use the **Filter by name** field to find it quicker.

- (Optional) **Creation Criteria**: Specify additional conditions that the specified source object(s) must meet for synchronization in this workflow step. This setting is not used in this example.

6. In the **Specify target** step, configure the following settings:
   - **Target connected system**: Specify the SQL Server connection here, created with the **Microsoft SQL Server Connector**. To do so, click **Specify** > **Select existing connected system**, then select the SQL Server connection (in this example, **SQL Connection**).
   - **Target object type**: Specify the target object type here. By default, when selecting an SQL Server connection in **Target connected system**, Active Roles Synchronization Service sets this setting to **sql-Object**, the object type used in this example.
7. In the **Specify creation rules** step, configure the logic (called forward synchronization rules) that Active Roles Synchronization Service will use to perform first-time synchronization and copy data entries from the SuccessFactors HR database over to the target SQL database.

   To do so, specify one or more unique attributes that Active Roles Synchronization Service can use to link the corresponding data entries in the connected SuccessFactors HR and SQL data systems. In this example, four such SuccessFactors HR attributes are specified: **userName**, **userId**, **emails.value** and **name.familyName**.

   To specify these creation rules:

   a. Click **Forward Sync Rule**.

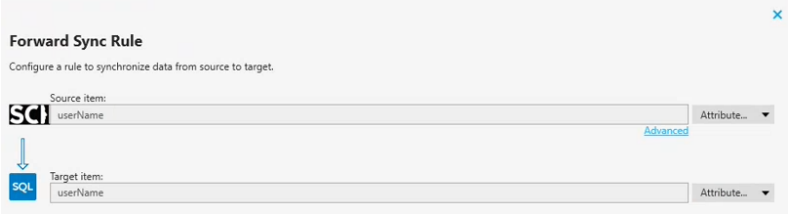
   b. Click **Source item** > **Attribute**, and in the **Select Object Attribute** window, search for the user name attribute in the SuccessFactors HR database (for example, **userName**), then click **OK**.

   > TIP: If the data entry is hard to find due to the length of the list, use the **Filter by name** field to find it quicker.

   c. Click **Target item** > **Attribute**, and search for the applicable user name attribute pair in the SQL database (for example, **userName**), then click **OK**.

   > TIP: If the data entry is hard to find due to the length of the list, use the **Filter by name** field to find it quicker.

Figure 19: Active Roles Synchronization Service – Mapping attributes for a forward synchronization rule

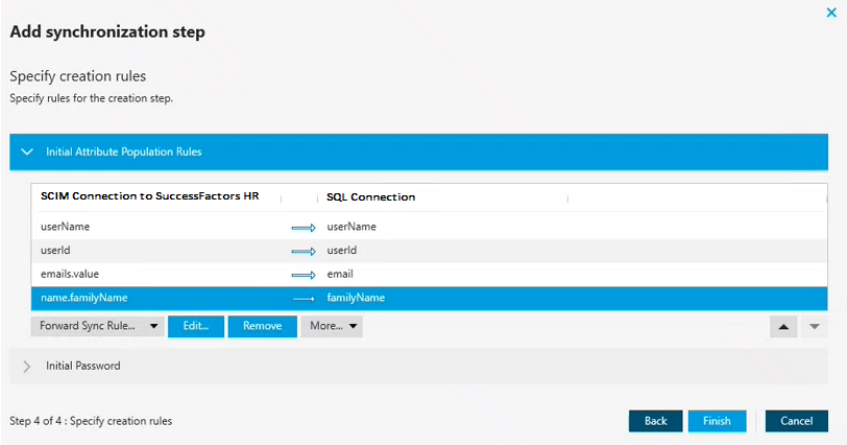


d. To apply the forward synchronization rule created for the specified user name attributes, click **OK**.

e. To configure synchronization rules for the **userId**, **emails.value** and **name.familyName** SuccessFactors HR data entries too, click **Forward Sync Rule** again, and repeat the previous sub-steps by selecting the source and target attributes applicable to these data entries.

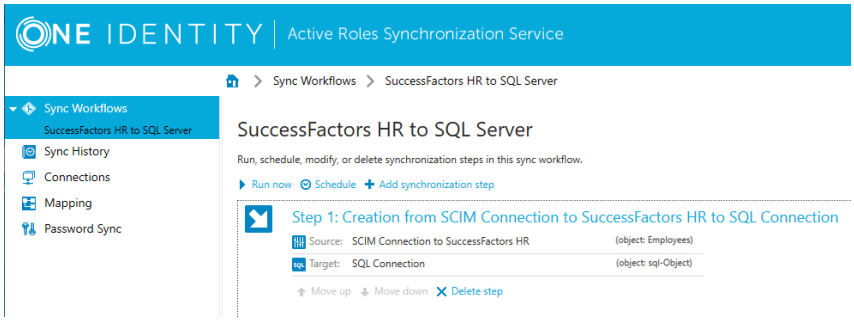
8. Once all forward synchronization rules are configured, to finish configuring the **Creation** step, click **Finish**.

Figure 20: Active Roles Synchronization Service – Finalizing all forward synchronization rules



This creates the **Creation** step as the first step of the sync workflow.

Figure 21: Active Roles Synchronization Service – Step 1 created for the SuccessFactors HR / SQL server workflow



9. Now that the **Creation** step of the workflow is configured, configure the **Update** step. To do so, click **Add synchronization step** again.

   The **Update** step of the workflow will be used to update existing data entries mapped between the SuccessFactors HR database and the target SQL database. The **Update** step performs data synchronization only for existing data entries: it does not create new ones. Because of this, you typically run this step after running the **Creation** step, and run only the **Update** step later once the data entries have been created with the **Creation** step.

10. In the **Select an action** step, select **Update**, then click **Next**.

11. In the **Specify source and criteria** step, configure the following settings:

    - **Source connected system**: Specify the SuccessFactors HR database connection here, created with the **Generic SCIM Connector**. To do so, click **Specify** > **Select existing connected system**, then select the SCIM-based connection (in this example, **SCIM Connection to SuccessFactors HR**).
    - **Source object type**: Specify the source object type here (in this example, the `Employees` object type). To do so, click **Select**, then in the **Select Object Type** window, select **Employees**, and click **OK**.

      TIP: If the data entry is hard to find due to the length of the list, use the **Filter by name** field to find it quicker.

    - (Optional) **Creation Criteria**: Specify additional conditions that the specified source object(s) must meet for synchronization in this workflow step. This setting is not used in this example.

12. In the **Specify target** step, configure the following settings:

    - **Target connected system**: Specify the SQL Server connection here, created with the **Microsoft SQL Server Connector**. To do so, click **Specify** > **Select existing connected system**, then select the SQL Server connection (in this example, **SQL Connection**).
    - **Target object type**: Specify the target object type here. By default, when selecting an SQL Server connection in **Target connected system**, Active Roles Synchronization Service sets this setting to **sql-Object**, the object type used in this example.

13. In the **Specify update rules** step, configure the forward synchronization rules that Active Roles Synchronization Service will use to update existing data entries in the target SQL database from the SuccessFactors HR database. In this example, four such attributes are specified: **userName**, **userId**, SuccessFactors HR ID (displayed as **sfid**) and metadata information (displayed as **meta**).

    To specify these creation rules:

    a. Click **Forward Sync Rule**.

    b. Click **Source item** > **Attribute**, and in the **Select Object Attribute** window, search for the user name attribute in the SuccessFactors HR database (for example, **userName**), then click **OK**.

    > TIP: If the data entry is hard to find due to the length of the list, use the **Filter by name** field to find it quicker.

    c. Click **Target item** > **Attribute**, and search for the applicable user name attribute pair in the SQL database (for example, **userName**), then click **OK**.

    > TIP: If the data entry is hard to find due to the length of the list, use the **Filter by name** field to find it quicker.

    d. To apply the forward synchronization rule created for the specified user name attributes, click **OK**.

    e. To configure synchronization rules for the **user ID**, **sfid** and **meta** data entries too, click **Forward Sync Rule** again, and repeat the previous sub-steps by selecting the source and target attributes applicable to these data entries.

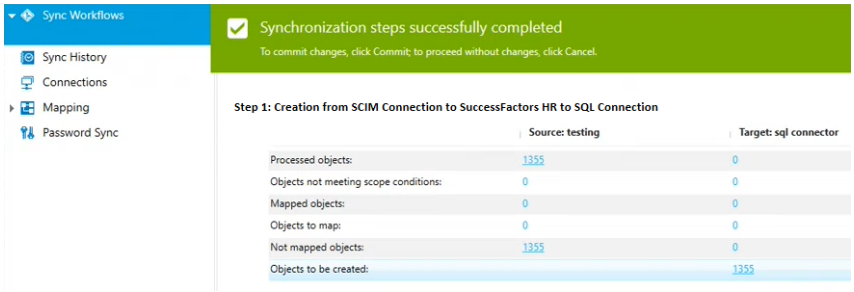
14. Once all forward synchronization rules are configured, to finish configuring the **Update** step, click **Finish**. The configured workflow will appear, containing both steps.

15. Start the workflow by clicking **Run workflow**. For the first-time run, select only **Step 1 (Creation from SCIM Connection to SuccessFactors HR to SQL Connection)**, then select the running method:

    - **Full Run** fetches all data entries specified in the workflow steps directly from the source system. As such, One Identity recommends using this method when running the workflow the first time, even if the process takes longer than a **Quick Run**.
    - **Quick Run** uses cached data whenever possible, and is normally faster.

    The run may take several minutes to complete.

Figure 22: Active Roles Synchronization Service – Running a configured sync workflow for the first time



16. Once Active Roles Synchronization Service found all mapped objects, apply the synchronization changes by clicking **Commit**.

    Alternatively, to check detailed information about the processed objects, click the **Processed objects** number. The **Objects processed in** window then opens, listing all new data objects that Active Roles Synchronization Service will synchronize to the target SQL database.

Synchronizing complex multi-value objects from a SCIM source system

Data sync workflows that import data with a connection based on the **Generic SCIM Connector** can import all three types of SCIM 2.0-based data entries:

- **Simple attributes**, that is, data entries with a single simple value. For example, a user ID specified in a single string is a simple attribute.
- **Complex single-value attributes**, that is, data entries specified with several sub-attributes. For example, the following `name` attribute is a complex single-value attribute, specifying the name of an employee with three simple sub-attributes:

  ```
  "name": {
         "givenName": "Sam",
         "familyName": "Smith",
         "formatted": "Sam Smith"
           },
  ```

  The value of complex single-value attributes is the sum of the sub-attribute values.
- **Complex multi-value attributes**, that is, data entries with multiple complex values, each of them specified with several simple sub-attributes. For example, the following `addresses` attribute is a complex multi-value attribute, specifying several addresses, each of them being a complex value containing several simple sub-attributes:

```
"addresses": [
        {
                "type": "work",
                "streetAddress": "22 Example Street",
                "region": "Springfield",
                "postalCode": "51487",
                "country": "United States",
                "primary": true
        },
        {
                "type": "home",
                "streetAddress": "12 Rue Exemple",
                "region": "Montreal",
                "postalCode": "46179",
                "country": "Canada"
        }
 ],
```

However, even though sync workflows using connections set with the **Generic SCIM Connector** can import all three of these value types, Active Roles Synchronization Service does not recognize complex single-value attributes and complex multi-value attributes, as they contain more values than what Active Roles Synchronization Service can identify for a single data entry by default.

To import complex single-value and multi-value attributes successfully, you can use the following methods:

- For **complex single-value attributes**, you can map each individual sub-attribute of the complex single-value attribute to separate attributes in the target system. For example, in case of the `name` complex single-value attribute, you can map the `givenName`, `familyName` and `formatted` sub-attributes to separate `name.givenName`, `name.familyName`, and `name.formatted` attributes in the target system, respectively.
- For **complex multi-value attributes**, you can use two methods:
  - When importing complex multi-value attributes, Active Roles Synchronization Service can take a single value (and its sub-attributes), map the sub-attributes to a set of target values (similarly to complex single-value attributes), then discard the rest of the complex values of the attribute.

    By default, Active Roles Synchronization Service takes the primary value of the complex multi-value attribute (marked with a specific `primary` sub-attribute). If no primary value is specified within the complex multi-value attribute, Active Roles Synchronization Service imports the first value (and its sub-attributes) only.

    > NOTE: This method imports only the primary value (or the first value, if no primary value is specified). Active Roles Synchronization Service will discard all other values (and their sub-attributes).
  - If you map a complex multi-value attribute (such as the `addresses` attribute shown in the above example) when configuring a mapping rule for a workflow,

you can configure an Active Roles Synchronization Service workflow to process and extract every value (and their sub-attributes) of the complex multi-value attribute with script-based attribute mapping.

The following procedure will provide an example on how to apply such a PowerShell script to properly process the `addresses` complex multi-value attribute shown in this chapter.

To configure a custom PowerShell script for a workflow to import complex multi-value attributes

1. In the Active Roles Synchronization Service, click **Sync Workflow**, then click the sync workflow that imports data from a SCIM-based source system (for example, the **SuccessFactors HR to SQL Server** workflow used in Creating a sync workflow for synchronizing data from a SCIM-based Starling Connect connector).
2. Click the first step of the workflow (in the example **SuccessFactors HR to SQL Server** workflow, this is named **Step 1 (Creation from SCIM Connection to SuccessFactors HR to SQL Connection)**.
3. Under **Creation Rules**, to open the initial population rules, click **Forward Sync Rule**.
4. In the **Forward Sync Rule** window, at the **Source item** setting, open the **Attribute** drop-down, and click **PowerShell Script**.
5. In the **PowerShell Script Editor**, paste the following script example, and click **OK**:

```
$addressesJsonArray = $srcObj["addresses"] | ConvertFrom-Json

 if ($addressesJsonArray) {
    for ($i = 0; $i -lt $addressesJsonArray.Length; $i++) {
       if ($addressesJsonArray[$i].type -eq "work") {
          return $addressesJsonArray[$i].streetAddress + ", " +
$addressesJsonArray[$i].region + ", " + $addressesJsonArray[$i].locality
        }
    }
 }
```

The example script contains the following key parts:

- `$srcObj` refers to the source object that the script will act on.
- `$srcObj["addresses"]` extracts the raw value of the `addresses` attribute. In this example, this attribute is a complex multi-value SCIM attribute, so the attribute value will be a JSON array.
- `$addressesJsonArray` is a .NET array object containing the values of the complex multi-value attribute.

The rest of the script performs the following steps:

   a. It checks that the array is valid.
   b. It traverses the elements of the array, and looks for the first element with a `type` sub-attribute with a `work` value.
   c. Once it finds an element with a `work` value `type`, it constructs a formatted string from the `streetAddress`, `region` and `locality` sub-attributes.
   d. It returns the results.
6. Use the output to parse and extract the data into other target values in the target system.

Appendix A

Developing PowerShell scripts for attribute synchronization rules

You can configure synchronization rules for such steps as creating, deprovisioning, or update. Synchronization Service provides a user interface (Synchronization Service Console) that allows you to set up a direct or rules-based synchronization rule without any coding.

However, to set up a script-based synchronization rule, you must develop a Windows PowerShell script that will build values of the target object attributes using values of the source object attributes.

This section provides some reference materials on using the Windows PowerShell Script Host feature and provides the sample script.

Accessing source and target objects using built-in hash tables

Synchronization Service synchronizes data between the source and target objects using the pre-configured synchronization rules.

In the PowerShell scripts used to set up the script-based synchronization rules, you can employ the `$srcObj` and `$dstObj` built-in associative arrays (hash tables) that allow the scripts to access the current values of attributes of the source and target objects, respectively. The array keys names are names of the object attributes.

For more information about the use of the associative arrays, see the *Microsoft PowerShell Documentation*.

In addition to `$srcObj` and `$dstObj`, Synchronization Service defines the `$Request` built-in hash table. The `$Request` key names are also names of the object attributes. The `$Request` hash table contains new values of the target object attributes to which the target object attributes must be set after completing the synchronization process.

To clarify the use of built-in hash tables, let us consider the following scenario: you synchronize between the `mail` attributes of user objects in an LDAP directory (source connected system) and Active Roles Synchronization Service (target connected system)

using the following synchronization rule: the value of the `mail` attribute in the target connected system must be equal to that in the source connected system concatenated with current date.

For example, before the synchronization process started, the source object had the `mail` attribute: `JDoe@mail1.mycompany.com`, the target object had the `mail` attribute: `JDoe@mail2009.mycompany.com`. After the synchronization process completes, the target user will have the following mail: `JDoe@mail1.mycompany.com (5 December, 2012)` (if you performed the synchronization process on 5 December, 2012.

The following code snippet illustrates the use of built-in hash tables:

```
#Returns "JDoe@mail1.mycompany.com
$strSourceMail=$srcObj["mail"]
#Returns JDoe@mail2009.mycompany.com
$strTargetMail=$DstObj["mail"]
#Returns JDoe@mail1.mycompany.com (5 January, 2010)
$strNewMail=$Request["mail"]
```

Example script

The following script illustrates the use of `$srcObj`.

A creating task (creating step of a sync workflow as applied to Synchronization Service) causes Synchronization Service to create user identity information from a delimited text file to Active Directory using the following creating rule: the `co` attribute in all created users must be set to the name of country where the user lives. The script-based creating rule calculates the `co` attribute value basing on the user's city (the `City` attribute in the connected data source).

The following script implements the described scenario:

```
# --- Retrieve the City attribute of the user object in connected data
source.
$userCity = $srcObj["City"]
# --- Determine the user's country
switch ($UserCity)
{
"New York" {$country = "United States"; break}
"Paris" {$country = "France"; break}
"Tokyo" {$country = "Japan"; break}
default {$country = "Unknown"}
}
# --- Return the user country. The script-based creating rule
# --- assigns this value to the "co" attribute in the created user object.
$country
# End of the script
```

Appendix B

Using PowerShell script to transform passwords

You can use a Windows PowerShell script in a password sync rule to transform passwords. This section provides some reference materials on how to write a Windows PowerShell script for password transformation.

Accessing source object password

To synchronize passwords between the source Active Directory domain and the target connected data system, Synchronization Service uses the password sync rules you configure. In a password rule settings, you can type a PowerShell script that transforms source Active Directory user passwords into object passwords for the target connected system. For example, you can use such a script if you want the object passwords in the source and target connected systems to be different.

When developing a PowerShell script to transform passwords, you can employ the `$srcPwd` built-in associative array (hash table) that allows the scripts to access the source object password. The `$srcPwd` returns a string that contains the object password.

Example script

To clarify the use of `$srcPwd`, consider a scenario where the target object password in the target connected data system must include only 8 first characters of the source object password in the source Active Directory domain.

The following scripts implements the described scenario:

```
if($srcPwd.length -gt 8)
{
$srcPwd.substring(0,8)
}
else
{
$srcPwd
}
# End of the script
```

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