One Identity Manager 8.1.1

Authorization and Authentication Guide
Copyright 2019 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 Web site (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. All other trademarks are the property of their respective owners.

Legend

⚠️ **WARNING**: A WARNING icon indicates a potential for property damage, personal injury, or death.

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

ℹ️ **IMPORTANT, NOTE, TIP, MOBILE, or VIDEO**: An information icon indicates supporting information.

One Identity Manager Authorization and Authentication Guide
Updated - August 2019
Version - 8.1.1
Contents

About this guide .................................................................................................................. 7
One Identity Manager Application roles ........................................................................... 8
Application roles overview ............................................................................................... 9
  Application roles for basic functions ............................................................................. 10
  Compliance & Security Officer ....................................................................................... 11
  Auditors ......................................................................................................................... 12
  Application roles for identity audit ............................................................................... 12
  Application roles for company policies ......................................................................... 14
  Application roles for attestation .................................................................................... 15
  Application roles for subscribable reports .................................................................... 16
Management level ............................................................................................................. 17
  Application roles for business roles ............................................................................. 17
  Application roles for organizations ............................................................................. 18
  Application roles for employees .................................................................................. 19
  Application roles for the IT Shop ............................................................................... 20
  Application roles for target systems ........................................................................... 21
  Application roles for Universal Cloud Interface .......................................................... 23
  Application roles for custom tasks .............................................................................. 24
Implementing the application roles .................................................................................. 25
Creating and editing application roles ............................................................................. 25
  Master data of application roles ................................................................................... 26
  Assigning employees to application roles ................................................................... 28
  Customized extension of application role edit permissions ......................................... 28
Additional tasks for managing application roles .............................................................. 29
  Creating dynamic roles for application roles ............................................................... 29
  Specifying mutually exclusive application roles ........................................................ 30
  Assigning subscribable reports to application roles ..................................................... 31
  Assigning extended properties to application roles ..................................................... 32
  Generating extended properties for application roles ................................................. 32
  Reports about application roles ..................................................................................... 32
Granting One Identity Manager schema permissions .................................................... 34
Predefined permissions groups and system users .................................................. 35
Rules for determining the valid permissions for tables and columns ..................... 38
Processing permissions groups .............................................................................. 40
  Permissions groups properties ............................................................................. 41
  Permissions group dependencies ......................................................................... 42
  Editing the dependencies of permissions groups .................................................. 43
Copying permissions groups .................................................................................... 44
Creating permissions groups ................................................................................... 45
Editing system users ................................................................................................ 45
  Creating system users .......................................................................................... 46
System users’ passwords .......................................................................................... 47
System user properties ............................................................................................. 48
Adding system users to permission groups .............................................................. 49
Which employees use the system user? ................................................................... 50
Deleting dynamic system users .............................................................................. 50
Editing table permissions and column permissions ............................................... 51
  Displaying the permissions of a permissions group .............................................. 51
  Displaying permissions for tables ....................................................................... 52
  Editing table properties ....................................................................................... 53
  Editing column permissions ............................................................................... 55
Copying table permissions and column permissions .............................................. 56
  Simulating permissions for system users ............................................................ 57
Displaying permissions for objects ......................................................................... 59
Displaying permissions for the current user ............................................................ 60
Assigning permissions groups to applications ....................................................... 60

Managing permissions to program features ......................................................... 62
  Program functions for starting the One Identity Manager tools ......................... 62
  Displaying the current user’s program functions ............................................... 64
  Assigning program function to permissions groups ............................................. 65
Permissions for executing scripts ........................................................................... 65
Permissions for executing methods ......................................................................... 66
Permissions for triggering processes ..................................................................... 67
Modifying permissions for executing actions in the Launchpad ............................. 68

One Identity Manager Authentication modules .................................................... 69
System users ..............................................................................................................................69
Generic single sign-on (role-based) ..........................................................................................70
Employee ..................................................................................................................................72
Employee (role-based) ..............................................................................................................72
Employee (dynamic) ..................................................................................................................73
User account .............................................................................................................................74
User account (role-based) .........................................................................................................75
Account-based system user .........................................................................................................76
Active Directory user account ....................................................................................................77
Active Directory user account (role-based) ................................................................................78
Active Directory user account (manual input) ...........................................................................79
Active Directory user account (manual input/role-based) ..........................................................80
Active Directory user account (dynamic) ...................................................................................81
LDAP user account (role-based) ...............................................................................................82
LDAP user account (dynamic) .................................................................................................84
HTTP Header .............................................................................................................................86
HTTP Header (role-based) ...........................................................................................................87
OAuth 2.0 / OpenID Connect .......................................................................................................88
OAuth 2.0 / OpenID Connect (role-based) ................................................................................89
Synchronization authentication module .......................................................................................90
Web agent authentication module .............................................................................................91
Component authentication module ...........................................................................................91
Crawler .......................................................................................................................................92
Password reset ...........................................................................................................................92
Password reset (role-based) .......................................................................................................93
Editing authentication modules ..................................................................................................94
Enabling authentication modules .............................................................................................95
Assigning authentication modules to applications ..................................................................95
Disabling or enabling authentication modules for applications .................................................95
Authentication module properties .............................................................................................96
Initial data for authentication modules .......................................................................................97
Configuration data for system user dynamic authentication .....................................................101
Example of a simple system user assignment ........................................................................102
Example of a system user assignment using a selection criterion ..........................................103
Example of a function group assignment .................................................................................104
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabling the validity of a login</td>
<td>105</td>
</tr>
<tr>
<td>OAuth 2.0/OpenID Connect configuration</td>
<td>106</td>
</tr>
<tr>
<td>Expiry of the OAuth 2.0/OpenID Connect authentication</td>
<td>106</td>
</tr>
<tr>
<td>Creating the OAuth 2.0/OpenID Connect configuration</td>
<td>108</td>
</tr>
<tr>
<td>Assigning OAuth 2.0/OpenID Connect configuration to web applications</td>
<td>112</td>
</tr>
<tr>
<td>Displaying the configuration of the identity provider and the OAuth 2.0/OpenID Connect applications</td>
<td>113</td>
</tr>
<tr>
<td>Defining enabling and disabling columns for the determination of user accounts</td>
<td>114</td>
</tr>
<tr>
<td>Multi-factor authentication in One Identity Manager</td>
<td>115</td>
</tr>
<tr>
<td>Editing table properties</td>
<td>117</td>
</tr>
<tr>
<td>Preparing the Starling 2FA token request</td>
<td>117</td>
</tr>
<tr>
<td>Requesting a security code</td>
<td>118</td>
</tr>
<tr>
<td>Allowing approval decisions using the Starling 2FA app</td>
<td>119</td>
</tr>
<tr>
<td>Granulated permissions for the SQL Server and database</td>
<td>120</td>
</tr>
<tr>
<td>Minimum access levels of One Identity Manager tools</td>
<td>120</td>
</tr>
<tr>
<td>Displaying database server logins</td>
<td>122</td>
</tr>
<tr>
<td>Displaying users' access levels</td>
<td>123</td>
</tr>
<tr>
<td>Displaying server and database permissions</td>
<td>123</td>
</tr>
<tr>
<td>About us</td>
<td>124</td>
</tr>
<tr>
<td>Contacting us</td>
<td>124</td>
</tr>
<tr>
<td>Technical support resources</td>
<td>124</td>
</tr>
<tr>
<td>Index</td>
<td>125</td>
</tr>
</tbody>
</table>
About this guide

You can use the One Identity Manager roles and permissions model to control the edit permissions for users of the One Identity Manager. Permissions for accessing tables and columns of the One Identity Manager schema are defined by permissions groups. Permissions groups can be linked to application roles. The users are assigned to application roles and therefore receive the permissions they require. The valid permissions for a user are determined when the user logs into One Identity Manager. One Identity Manager provides different authentication modules for the login.

The One Identity Manager Authorization and Authentication Guide describes the basics and features of the internal One Identity Manager roles and permission model.

You will find an overview of the default application roles, default permissions groups and system users of the One Identity Manager. You will learn how to get the application roles up and running. The guide also explains how you grant permissions for the tables and columns of the One Identity Manager schema. In addition, you will find an overview of the various One Identity Manager authentication modules.

This guide is intended for end users, system administrators, consultants, analysts, and any other IT professionals using the product.

**NOTE:** This guide describes One Identity Manager functionality available to the default user. It is possible that not all the functions described here are available to you. This depends on your system configuration and permissions.

Available documentation

You can access the One Identity Manager documentation in Manager and in Designer by selecting **Help | Search.** The online version of the One Identity Manager documentation is available in the Support-Portal under **Online-Documentation.** You will find videos with additional information at [www.YouTube.com/OneIdentity](http://www.YouTube.com/OneIdentity).
One Identity Manager Application roles

You can use the One Identity Manager role model to control edit permissions for One Identity Manager users. This role model takes into account technical aspects, for example, One Identity Manager tool administrative rights, as well as functional aspects, which result from One Identity Manager user tasks within the company structure (for example, permissions for approving requests). The One Identity Manager makes so-called application roles available.

Application roles have the following aims:

- Program functions, employees, company resources, approval workflows and approval policies are assigned to fixed application roles. Write permissions for these application roles do not need to be defined specifically for the company. This simplifies administration of access permissions.
- Enables audit secure internal administration of One Identity Manager users and their write permissions. Permissions can be granted through assignment, request and approval or by calculation on account of specific properties. Furthermore, issuing permissions with the attestation function is integrated into the attestation process.
- Users are provided with initial permissions, which they required for carrying out their tasks. This is a way, for example, to create initially required user accounts.

Application roles can be limited to permissions groups whose write permissions are predefined by One Identity Manager. Controlling write permissions:

- Navigation configuration in administration tools
- Access to objects and their properties
- Which interface forms and tasks are displayed
- Availability of special program functionality

Users must be role-based to use application roles for logging in to One Identity Manager. Role-based authentications module finds the valid write permissions from all the user’s application roles. This provides the One Identity Manager user with permissions corresponding to their application roles for the One Identity Manager functions when they log onto One Identity Manager tools.
Detailed information about this topic

- Application roles overview on page 9
- Implementing the application roles on page 25
- Creating and editing application roles on page 25

Related topics

- Granting One Identity Manager schema permissions on page 34
- One Identity Manager Authentication modules on page 69

Application roles overview

One Identity Manager supplies default application roles whose permissions are matched to the different task and functions. Assign employees to default applications who take on individual tasks and functions. You can also create your own application roles for custom defined tasks.

| NOTE: | Default application roles are defined in One Identity Manager modules and are not available until the modules are installed. You cannot delete default application roles. |

The following default application roles are defined:

- Application roles for basic functions
- Compliance & Security Officer
- Auditors
- Application roles for identity audit
- Application roles for company policies
- Application roles for attestation
- Application roles for subscribable reports
- Management level
- Application roles for business roles
- Application roles for organizations
- Application roles for employees
- Application roles for the IT Shop
- Application roles for target systems
- Application roles for Universal Cloud Interface
- Application roles for custom tasks
Application roles for basic functions

**NOTE:** This application role is available if the Identity Management Base Module is installed.

The following application roles are available to you for the basic functionality in One Identity Manager.

### Table 1: Application Roles for Basic Functions

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
</table>
| Administrators         | Administrators must be assigned to **Base roles | Administrators**. Users with this application role:  
  - Administer application roles for administrators.  
  - Assign employees to administrator application roles.  
  - Can other employees **Base roles | Administrators** and edit conflicting application roles.  
  - See the master data for the other application roles.  
  - Can use Password Reset Portal to set passwords for selected system users. |
| Everyone (change)      | **Base roles | Everyone (change)** is automatically assigned to every user. Users with this application role:  
  - Can edit certain employee master data in the Web Portal.  
  Should every user be automatically assigned to a custom permissions group when they log in, then this permissions group can be added to the application role.  
  Members of this application role are determined through a dynamic role. |
| Everyone (lookup)      | **Base roles | Everyone (Lookup)** is automatically assigned to every user. Users with this application role:  
  - Obtain read access to objects in the Web Portal.  
  Should every user be automatically assigned to a custom permissions group when they log in, then this permissions group can be added to the application role.  
  Members of this application role are determined through a dynamic role. |
<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Employee managers</strong></td>
<td>**Base roles</td>
</tr>
<tr>
<td></td>
<td>- Can edit master data for the objects they are responsible for and assign company resources to them.</td>
</tr>
<tr>
<td></td>
<td>- Can edit master data for their employees in the Web Portal.</td>
</tr>
<tr>
<td></td>
<td>- Can add their staff members to the IT Shop.</td>
</tr>
<tr>
<td></td>
<td>- Employee and department managers can add new employees in the Web Portal.</td>
</tr>
<tr>
<td></td>
<td>- Can view their staff's compliance rule violations in the Web Portal.</td>
</tr>
<tr>
<td></td>
<td>Members of this application role are determined through a dynamic role.</td>
</tr>
<tr>
<td><strong>Birthright Assignments</strong></td>
<td>**Base roles</td>
</tr>
<tr>
<td><strong>Operations support.</strong></td>
<td>Employees that use the Operations Support Web Portal, must be assigned the application role **Base roles</td>
</tr>
<tr>
<td></td>
<td>Users with this application role:</td>
</tr>
<tr>
<td></td>
<td>- Monitor handling of Job queue processes</td>
</tr>
<tr>
<td></td>
<td>- Monitor handling of the DBQueue</td>
</tr>
<tr>
<td></td>
<td>- Create access codes to enable employees to log on to Password Reset Portal</td>
</tr>
</tbody>
</table>

**Related topics**

- [Customized extension of application role edit permissions](#) on page 28

**Compliance & Security Officer**

**NOTE:** This application role is available if Attestation Module, Compliance Rules Module or Company Policies Module is installed.
Compliance and security officers must be assigned to **Identity & Access Governance | Compliance & Security Officer**.

Users with this application role:

- View all compliance relevant information and other analysis in the Web Portal. This includes attestation policies, company policies and policy violations, compliance rules, and rule violations and risk index functions.
- Edit attestation polices.

**Auditors**

**NOTE:** This application role is available if Attestation Module, Compliance Rules Module or Company Policies Module is installed.

Auditors are assigned to the application role **Identity & Access Governance | Auditors**.

Users with this application role:

- See the Web Portal all the relevant data for an audit.

**Application roles for identity audit**

**NOTE:** This application role is available if the Compliance Rules Module is installed.

The following application roles are available for managing compliance rule:

**Table 2: Application roles for identity audit**

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrators</td>
<td>Administrators must be assigned to the application role **Identity &amp; Access Governance</td>
</tr>
<tr>
<td></td>
<td>- Enter base data for setting up company policies.</td>
</tr>
<tr>
<td></td>
<td>- Create compliance rules and assign rule supervisors to them.</td>
</tr>
<tr>
<td></td>
<td>- Can start rule checking and view rule violations as required.</td>
</tr>
<tr>
<td></td>
<td>- Create reports about rule violations.</td>
</tr>
<tr>
<td></td>
<td>- Enter mitigating controls.</td>
</tr>
<tr>
<td></td>
<td>- Create and edit risk index functions.</td>
</tr>
<tr>
<td></td>
<td>- Monitor Identity Audit functions.</td>
</tr>
<tr>
<td>Application role</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Rule supervisors</td>
<td>Rule supervisors must be assigned to **Identity &amp; Access Governance</td>
</tr>
<tr>
<td>Users with this application role:</td>
<td></td>
</tr>
<tr>
<td>l Are responsible for compliance rule content, for example, an auditor or a auditing department.</td>
<td></td>
</tr>
<tr>
<td>l Edit the compliance rule working copies, which are assigned to the application role.</td>
<td></td>
</tr>
<tr>
<td>l Enable and disable compliance rules.</td>
<td></td>
</tr>
<tr>
<td>l Can start rule checking and view rule violations as required.</td>
<td></td>
</tr>
<tr>
<td>l Assign mitigating controls.</td>
<td></td>
</tr>
</tbody>
</table>

| Exception approvers | Administrators must be assigned to **Identity & Access Governance | Identity Audit | Exception approvers** or to a child role. |
| Users with this application role: | |
| l Edit rule violations in the Web Portal. | |
| l Can grant exception approval or revoke it in the Web Portal. | |

| Attestors | Attestors must be assigned to **Identity & Access Governance | Identity Audit | Attestors**. |
| Users with this application role: | |
| l Attest compliance rules and exception approvals in the Web Portal for which they are responsible. | |
| l Can view master data for these compliance rules but not edit them. | |

| NOTE: This application role is available if the module Attestation Module is installed. |

<p>| Maintain SAP Functions | Administrators must be assigned to <strong>Identity &amp; Access Governance | Identity Audit | Maintain SAP functions</strong> or to a child role. |
| Users with this application role: | |
| l Are responsible for SAP function contents. | |
| l Edit working copies of function definitions for which they are | |</p>
<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>responsible.</td>
</tr>
<tr>
<td></td>
<td>• Define function instances and variables sets for SAP functions.</td>
</tr>
<tr>
<td></td>
<td>• Assign mitigating controls.</td>
</tr>
</tbody>
</table>

**NOTE:** This application role is available if the module SAP R/3 Compliance Add-on Module is installed.

### Application roles for company policies

**NOTE:** This application role is available if the Company Policies Module is installed.

The following application roles are available for managing company policies:

#### Table 3: Application Roles for Company Policies

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
</table>
| Administrators          | Administrators must be assigned to the application role **Identity & Access Governance | Company policies | Administrators**.  
Users with this application role: |
|                         | • Enter base data for for setting up company policies. |
|                         | • Set up policies and assign policy supervisors to them. |
|                         | • Can calculation policies and view policy violations if required. |
|                         | • Set up reports about policy violations. |
|                         | • Enter mitigating controls. |
|                         | • Create and edit risk index functions. |
|                         | • Administer application roles for policy supervisors, exception approvers and attestors. |
|                         | • Set up other application roles as required. |

| Policy supervisors      | Policy supervisors must be assigned to **Identity & Access Governance | Company policies | Policy supervisors** or another child application role.  
Users with this application role: |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• Are responsible for the contents of company policies.</td>
</tr>
<tr>
<td></td>
<td>• Edit working copies of company policies.</td>
</tr>
<tr>
<td></td>
<td>• Enable and disable company policies.</td>
</tr>
</tbody>
</table>
### Application roles for attestation

**NOTE:** This application role is available if the module Attestation Module is installed.

The following application role is available for managing attestation procedures:

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
</table>
| Administrators | Administrators are assigned to **Identity & Access Governance | Attestation | Administrators**.  

Users with this application role:  
- Define attestation procedures and attestation policies.  
- Create approval policies and approval workflows.  
- Specify which approval procedure to use to find attestors.

---

NOTE: This application role is available if the module Attestation Module is installed.
<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
</table>
|                  | - Set up attestation case notifications.  
|                  | - Configure attestation schedules.  
|                  | - Enter mitigating controls.  
|                  | - Create and edit risk index functions.  
|                  | - Monitor attestation cases.  
| Chief approval team | The chief approver must be assigned to **Identity & Access Governance| Attestation | Chief approval team**.  
|                  | - Approve using attestation cases.  
|                  | - Assign attestation cases to other attestors.  

**NOTE:** Attestors in charge are determined through approval procedures. Other application roles may be applied here. Application roles for attestors are defined in different module and are available if the Attestation Module is installed.

## Application roles for subscribable reports

**NOTE:** This application role is available if the module Report Subscription Module is installed.

The following application role is available for managing subscribable reports:

**Table 5: Application Roles for Subscribable Reports**

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
</table>
| Administrators   | Administrators must be assigned to **Identity & Access Governance | Company policies | Report Subscriptions**.  
|                  | - Create subscribable reports from existing reports.  
|                  | - Configure report parameters for subscribable reports.  
|                  | - Assign subscribable reports to employees, company structures or IT Shop shelves.  
|                  | - Create custom mail templates for sending subscribed reports by email.  

Management level

**NOTE:** This application role is available if the Identity Management Base Module is installed.

The user must be assigned to the application role *Identity Management | Management level*.

Users with this application role:

- Can view reports and statistics for management levels in the Web Portal.

Application roles for business roles

**NOTE:** This application role is available if the module Business Roles Module is installed.

The following application roles are available for the administration of business roles:

**Table 6: Application Roles for Business Roles**

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
</table>
| Administrators   | Administrators must be assigned to the application role *Identity Management | Business roles | Administrators*. Users with this application role:
|                  |   • Create and edit business roles. 
|                  |   • Assign company resources to business roles. 
|                  |   • Administrate application roles for role approvers, role approvers (IT), and attestors. 
|                  |   • Set up other application roles as required. |
| Attestors        | Attestors must be assigned to *Identity Management | Business roles | Attestors* or a child application role. Users with this application role:
|                  |   • Attest correct assignment of company resource to business roles for which they are responsible. 
|                  |   • Can view master data for these business roles but not edit them. |
| Role approver    | Approvers must be assigned to the application role *Identity Management | Business roles | Role approver*.
<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>**Management</td>
<td>Business roles</td>
</tr>
</tbody>
</table>

Users with this application role:

- Are approvers for the IT Shop.
- Approve requests from business roles for which they are responsible.

| Role approver (IT) | IT role approvers must be assigned to **Identity Management | Business roles | Role approvers (IT)** or a child application role. |

Users with this application role:

- Are IT role approvers for the IT Shop.
- Approve requests from business roles for which they are responsible.

### Application roles for organizations

**NOTE:** This application role is available if the Identity Management Base Module is installed.

The following application roles are available for the administration of departments, cost centers and locations:

#### Table 7: Application Roles for Organizations

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Administrators</strong></td>
<td>Administrators must be assigned to the application role **Identity Management</td>
</tr>
</tbody>
</table>

Users with this application role:

- Set up and edit departments, cost centers and locations.
- Assign company resources to departments, cost centers and locations.
- Administrate application roles for role approvers, role approvers (IT), and attestors.
- Set up other application roles as required.

<p>| Attestors | Attestors must be assigned to the application role <strong>Identity Management | Organizations | Attestors</strong> or a child application role. |</p>
<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attest</td>
<td>Attest correct assignment of company resources to departments, cost centers and locations for which they are responsible.</td>
</tr>
<tr>
<td></td>
<td>Can view master data for departments, cost centers and locations but cannot edit them.</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> This application role is available if the module Attestation Module is installed.</td>
</tr>
<tr>
<td>Role approver</td>
<td>Role approvers must be assigned to the **Identity Management</td>
</tr>
<tr>
<td></td>
<td>Users with this application role:</td>
</tr>
<tr>
<td></td>
<td>• Are approvers for the IT Shop.</td>
</tr>
<tr>
<td></td>
<td>• Approve request from departments, cost centers, and locations for which they are responsible.</td>
</tr>
<tr>
<td>Role approver (IT)</td>
<td>IT role approvers must be assigned to **Identity Management</td>
</tr>
<tr>
<td></td>
<td>Users with this application role:</td>
</tr>
<tr>
<td></td>
<td>• Are IT role approvers for the IT Shop.</td>
</tr>
<tr>
<td></td>
<td>• Approve request from departments, cost centers, and locations for which they are responsible.</td>
</tr>
</tbody>
</table>

## Application roles for employees

**NOTE:** This application role is available if the Identity Management Base Module is installed.

The following application role is available for employee administration:

**Table 8: Application Roles for Employees**

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrators</td>
<td>Employee administrators must be assigned to the application role **Identity Management</td>
</tr>
<tr>
<td></td>
<td>Users with this application role:</td>
</tr>
<tr>
<td></td>
<td>• Can edit master data for all employees</td>
</tr>
</tbody>
</table>
### Application roles for the IT Shop

**NOTE:** This application role is available if the Identity Management Base Module is installed.

The following application roles are available for the IT Shop administration:

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrators</td>
<td>Administrators must be assigned to the **Request &amp; Fulfillment</td>
</tr>
<tr>
<td></td>
<td>- Create the IT Shop structure with shops, shelves, customers, templates and service catalog.</td>
</tr>
<tr>
<td></td>
<td>- Create approval policies and approval workflows.</td>
</tr>
<tr>
<td></td>
<td>- Specify which approval procedure to use to find attestors.</td>
</tr>
<tr>
<td></td>
<td>- Create products and service items.</td>
</tr>
<tr>
<td></td>
<td>- Set up request notifications.</td>
</tr>
<tr>
<td></td>
<td>- Monitor request procedures.</td>
</tr>
<tr>
<td></td>
<td>- Administrate application roles for product owners and attestors.</td>
</tr>
<tr>
<td></td>
<td>- Set up other application roles as required.</td>
</tr>
<tr>
<td></td>
<td>- Create extended properties for company resources of any type.</td>
</tr>
<tr>
<td></td>
<td>- Edit the resources and assign them to IT Shop structures and employees.</td>
</tr>
<tr>
<td></td>
<td>- Assign system entitlements to IT Shop structures.</td>
</tr>
</tbody>
</table>

<p>| Product          | Product owners must be assigned to the <strong>Request &amp; Fulfillment | IT</strong> |</p>
<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
</table>
| owners           | **Shop | Product owner** application role or a child application role. Users with this application role:  
|                  |  
|                  |   - Approve through requests.  
|                  |   - Edit service items and service categories under their management.  
| Attestors        | Attestors must be assigned to the **Request & Fulfillment | IT Shop | Attestors** application role. Users with this application role:  
|                  |  
|                  |   - Attest correct assignment of company resource to IT Shop structures for which they are responsible.  
|                  |   - Can view master data for these IT Shop structures but not edit them.  

**NOTE:** This application role is available if the module Attestation Module is installed.

| Chief approval team | Chief approvers must be assigned to the **Request & Fulfillment | IT Shop | Chief approval team** application role. Users with this application role:  
|---------------------|  
|                     |   - Approve through requests.  
|                     |   - Assign requests to other approvers.  

**NOTE:** Approvers in charge are determined through approval procedures. Other application roles may be applied here. Application roles for approvers are defined in different modules and are available there.

### Application roles for target systems

**NOTE:** Application roles are dependent on the target system and are contained in One Identity Manager modules. Application roles are not available until the modules are installed.

The following application roles are available for target system administration:
Table 10: Application Roles for Target Systems

<table>
<thead>
<tr>
<th>Users</th>
<th>Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Target system administrators</td>
<td>Target system administrators must be assigned to the **Target systems</td>
</tr>
<tr>
<td></td>
<td>• Administrate application roles for individual target systems types.</td>
</tr>
<tr>
<td></td>
<td>• Specify the target system manager.</td>
</tr>
<tr>
<td></td>
<td>• Set up other application roles for target system managers if required.</td>
</tr>
<tr>
<td></td>
<td>• Specify which application roles for target system managers are mutually exclusive.</td>
</tr>
<tr>
<td></td>
<td>• Authorize other employee to be target system administrators.</td>
</tr>
<tr>
<td></td>
<td>• Do not assume any administrative tasks within the target system.</td>
</tr>
<tr>
<td>Target system managers</td>
<td>Target system managers must be assigned to the application role **Target systems</td>
</tr>
<tr>
<td></td>
<td><strong>NOTE:</strong> There is at least one application role per target system for target system managers. This application role is available if the target system module is installed.</td>
</tr>
<tr>
<td></td>
<td>Users with this application role:</td>
</tr>
<tr>
<td></td>
<td>• Assume administrative tasks for the target system.</td>
</tr>
<tr>
<td></td>
<td>• Create, change or delete target system objects, like user accounts or groups.</td>
</tr>
<tr>
<td></td>
<td>• Edit password policies for the target system.</td>
</tr>
<tr>
<td></td>
<td>• Prepare system entitlements for adding to the IT Shop.</td>
</tr>
<tr>
<td></td>
<td>• Can add employees, who have an other identity than the <strong>Primary identity</strong>.</td>
</tr>
<tr>
<td></td>
<td>• Configure synchronization in the Synchronization Editor and defines the mapping for comparing target systems and One Identity Manager.</td>
</tr>
<tr>
<td></td>
<td>• Edit the synchronization's target system types and outstanding objects.</td>
</tr>
<tr>
<td></td>
<td>• Authorize other employees within their area of responsibility as target system managers and create child application roles if required.</td>
</tr>
</tbody>
</table>
Application roles for Universal Cloud Interface

NOTE: Application roles are available if the Universal Cloud Interface Module is installed.

The following application roles are available for managing cloud systems.

Table 11: Application Roles for the Universal Cloud Interface

<table>
<thead>
<tr>
<th>Users</th>
<th>Task</th>
</tr>
</thead>
</table>
| Target system managers for Unified Namespace | Target system managers must be assigned to the **Target systems | Unified Namespace** application role or a child role. Users with this application role:  
  - Obtain view of the objects in the connected target systems across all target systems.  
  - Can create reports across all target systems. If the users are also target system managers of the basic underlying target systems, you can manage these target systems through the Unified Namespace.  

<table>
<thead>
<tr>
<th>Users</th>
<th>Task</th>
</tr>
</thead>
</table>
| Cloud administrators        | Administrators must be assigned to the **Universal Cloud Interface | Administrators** application role or a child application role. Users with this application role:  
  - Manage application roles for the Universal Cloud Interface.  
  - Set up other application roles as required.  
  - Configure synchronization in the Synchronization Editor and define the mapping for comparing tcloud applications and One Identity Manager.  
  - Edit cloud application in the Manager.  
  - Edit pending, manual provisioning processes in the Web Portal and obtain statistics.  
  - Obtain information about the cloud objects in the Web Portal and the Manager.  

<table>
<thead>
<tr>
<th>Users</th>
<th>Task</th>
</tr>
</thead>
</table>
| Cloud operators           | Operators must be assigned to the **Universal Cloud Interface | Operators** application role or a child application role. Users with this application role:
|                           |   - Edit pending, manual provisioning processes in the Web Portal and obtain statistics. |
| Cloud auditors            | Auditors must be assigned to the **Universal Cloud Interface | Auditors** application role or a child application role. Users with this application role:
|                           |   - Can view manual provisioning processes in the Web Portal and obtain statistics. |

**Application roles for custom tasks**

| NOTE: This application role is available if the Identity Management Base Module is installed. |

The following application roles are available for customer features and tasks.

**Table 12: Application Roles for Custom Tasks**

<table>
<thead>
<tr>
<th>Application role</th>
<th>Description</th>
</tr>
</thead>
</table>
| Administrators            | Administrators must be assigned to **Custom | Administrators**. Users with this application role:  
                           |   - Administrate custom application roles.  
                           |   - Set up other application roles for managers if required. |
| Manager/supervisor        | Managers must be assigned to **Custom | Managers or a subordinate role. Users with this application role:  
                           |   - Add custom task in One Identity Manager.  
                           |   - Configure and start synchronization in the Synchronization Editor.  
                           |   - Edit the synchronization’s target system types as well as outstanding objects in the Manager. |

You can use these application roles, for example, to guarantee One Identity Manager users write permissions on custom tables or columns. All application roles that you define here must obtain their write permissions through custom permissions groups.
Implementing the application roles

**IMPORTANT:** To use application roles you must add one employee to the application role **Base roles | Administrators.** This employee is the authorized to assigned administrative One Identity Manager application roles to other employees.

Run this task once.

*To initially add an employee to the application role Base roles | Administrators*

1. Log into the Manager as a non role-based administrative user.
2. Select the **Employees | Employees.**
3. Select the employee to be assigned to the application role **Base role | Administrators.**
4. Select the **Authorize as One Identity Manager administrator** task.

**NOTE:** Once you update the view in Manager, the **Authorize as One Identity Manager administrator** task is no longer displayed in the task view. That means that the task can only be run when there are no other employees assigned to this application role.

After you have been working with One Identity Manager for a while, it is possible that no more employees are assigned to the **Base roles | Administrators** application role. In this case, proceed as described above in order to reassign an employee to this application role.

The One Identity Manager user with the application role **Base roles | Administrators** can now add more employees to application roles and edit the application role master data.

**Related topics**

- Assigning employees to application roles on page 28
- Creating and editing application roles on page 25

Creating and editing application roles

To set up your first application roles you need to add an employee to the application role **Base roles | Administrators.** This employee is authorized to add more employees to different administration application roles. For more information, see Implementing the application roles on page 25.

Administrators can edit child application roles, set up more application roles and assigned employees.

**NOTE:** To edit the application role, log on to the Manager using a role-based authentication module.
To edit an application role

1. In Manager in the One Identity Manager Administration category, select the Application role.
2. Select Change master data.
3. Edit the application role's master data.
4. Save the changes.

To create a new application role

1. In the Manager in the One Identity Manager Administration category, select the application role under which you want to create a new application role.
2. Click in the result list.
3. Enter the application role master data.
4. Save the changes.

NOTE: You cannot delete default application roles.

Related topics

- Master data of application roles on page 26
- Assigning employees to application roles on page 28
- Customized extension of application role edit permissions on page 28
- One Identity Manager Authentication modules on page 69

Master data of application roles

Table 13: Application Role Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application role</td>
<td>Application role name.</td>
</tr>
<tr>
<td>Internal name</td>
<td>Empty text field for a internal company identifier</td>
</tr>
<tr>
<td>Full name</td>
<td>Full name of application role. Is made up automatically from the application role name and the parent application role.</td>
</tr>
<tr>
<td>Parent application role</td>
<td>Application role to which the application role being edited is subordinate.</td>
</tr>
<tr>
<td>Department, location,</td>
<td>Additional information for the application role definition. These input fields are only used for information. They do not indicate for which department,</td>
</tr>
<tr>
<td>Property</td>
<td>Meaning</td>
</tr>
<tr>
<td>----------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>cost center</td>
<td>cost center or location the application roles are responsible.</td>
</tr>
<tr>
<td>Manager</td>
<td>Manager responsible for the application role.</td>
</tr>
<tr>
<td>Deputy manager</td>
<td>Deputy manager for the application role.</td>
</tr>
<tr>
<td>Permissions group</td>
<td>Permissions group for determining write permissions on role-based login. The application role is given access permissions of the associated permissions group. If there is no permissions group assigned, the application role gets write permissions from the parent application role. Administrators can assign the rest of the application roles to custom defined permissions groups. For more information, see Customized extension of application role edit permissions on page 28.</td>
</tr>
<tr>
<td>Note</td>
<td>Permissions groups for default administrator application roles for cannot be edited.</td>
</tr>
<tr>
<td>Description</td>
<td>Spare text box for additional explanation.</td>
</tr>
<tr>
<td>Comment</td>
<td>Spare text box for additional explanation.</td>
</tr>
<tr>
<td>Certification status</td>
<td>Status of the application role’s certification. The following values can be selected.</td>
</tr>
<tr>
<td></td>
<td>- <strong>New</strong>: The application role was newly created in the One Identity Manager database.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Certified</strong>: The master data of the application role is approved by a manager.</td>
</tr>
<tr>
<td></td>
<td>- <strong>Denied</strong>: The application role master data was not approved by a manager.</td>
</tr>
<tr>
<td>Block inheritance</td>
<td>Specifies whether inheritance for this application role can be discontinued. Set this option to prevent company resources being inherited by child application roles.</td>
</tr>
<tr>
<td>Dynamic roles not allowed</td>
<td>Specifies whether a dynamic role can be created for the application role.</td>
</tr>
<tr>
<td>Spare field no. 01 ... Spare field no. 10</td>
<td>Additional company specific information. Use Designer to customize display names, formats and templates for the input fields.</td>
</tr>
</tbody>
</table>
Assigning employees to application roles

Assigned employees obtain all the write permissions of the permission group to which the application role (or a parent application role) is assigned. In addition, employees obtain the company resources assigned to the application role.

Employees of the parent application role are inherited if no employees are directly assigned to an application role.

**NOTE:** The application roles for Base roles | Everyone (Change), Base roles | Everyone (Lookup), Base roles | Employee Managers and Base roles | Birth-right Assignments are automatically assigned to employees. Do not make any manually assignments to these application roles.

To assign employees to an application role

1. In Manager in the One Identity Manager Administration category, select the Application role.
2. Select Assign employees in the task view.
3. Assign employees in Add assignments.
   
   **TIP:** In the Remove assignments area, you can remove the assignment of employees.

   **To remove an assignment**
   
   - Select the employee and double-click.

4. Save the changes.

Related topics

- Creating dynamic roles for application roles on page 29

Customized extension of application role edit permissions

For role-based login, the application roles require a link to a permissions group in which write permissions for One Identity Manager are defined. The application role is given access permissions of the associated permissions group. If there is no permissions group assigned, the application role gets write permissions from the parent application role.

Some of the default application roles are already assigned permissions groups. These permissions groups have the edit permissions for the tables and columns and are equipped with menu items, forms, tasks and program functions, which allow the application data to be edited in Manager and in Web Portal.

You can assign customized permissions groups to application roles so that the write permissions for application roles meet your company requirements. You need to ensure
that your custom permissions groups contain all the write permissions of the default permissions groups for these application roles. This allows users with these application roles to use all default One Identity Manager functionality.

NOTE: You can simplify grouping of permissions by using hierarchical linking of permissions groups. Permissions from hierarchical permissions groups are inherited from top to bottom. That means that a permissions group contains all the permissions belonging parent permissions groups.

Proceed as follows:

1. Create a new permissions group in the Designer.
   
   NOTE: Set the Only use for role-based authentication option for the permissions group.

2. In Designer, make the new permissions group dependent on the default permissions group of the application role. Assign the default permissions group as a parent permissions group. As a result, the newly defined permissions group inherits the properties of the default permissions group.

3. In Designer, grant additional edit permissions for menu items, forms, tables or columns.

4. In Manager, assign the new permissions group to the application role.

A user who logs in to the Manager or to the Web Portal with an application role changed in this way receives – in addition to the default privileges of this application role – the custom edit permissions.

Related topics

- Master data of application roles on page 26
- Granting One Identity Manager schema permissions on page 34

Additional tasks for managing application roles

After you have entered the master data, you can run the following tasks.

Creating dynamic roles for application roles

Use this task to assign employees to an application role through dynamic roles. For detailed information about using dynamic roles, see the One Identity Manager Identity Management Base Module Administration Guide.

NOTE: The task Create dynamic role is only available for application roles, which do not have the option Dynamic roles not allowed set.
To create a dynamic role for the application role

1. In Manager in the One Identity Manager Administration category, select the Application role.
2. Select Create dynamic role in the task view.
3. Enter the required master data. The following applies to dynamic roles for application roles:
   - **Object class**: Select Employee.
   - **Application role**: This data is preset with the selected application role. If these objects fulfill the dynamic role conditions, they become members in the application role.
   - **Dynamic role**: The dynamic role name is made up of the object class and the full name of the application role by default.
4. Save the changes.

To edit a dynamic role

1. In Manager in the One Identity Manager Administration category, select the Application role.
2. Select Application role overview in the task view.
3. In the overview form, click the dynamic role name in the Dynamic roles form element.
4. Select Change master data.
5. Edit the dynamic role.
6. Save the changes.

Related topics

- Master data of application roles on page 26

Specifying mutually exclusive application roles

It is possible that employees cannot own certain system roles at the same time. Thus, for example, exception approvers for rule violations may not be rule supervisors at the same time. To implement this behavior, you can specify mutually exclusive application roles. Then you cannot assign these application roles to the same person anymore.

NOTE: Only system roles, which are defined directly as conflicting application roles cannot be assigned to the same employee. Definitions made on parent or child application roles do not effect the assignment.
To configure inheritance exclusion

- In Designer, enable the QER | Structures | ExcludeStructures configuration parameter and compile the database.

To specify inheritance exclusion for application roles

1. In Manager in the One Identity Manager Administration category, select the application role for which you want to define an inheritance exclusion.
2. Select Edit conflicting application roles in the task view.
3. Assign the application roles that are mutually exclusive to the selected application role in Add assignments.
   - OR -
   Remove the application roles that are no longer mutually exclusive in the Remove assignments area.
4. Save the changes.

Assigning subscribable reports to application roles

Use this task to assign subscribable reports to an application role. All employee in this application role can subscribe to reports in the Web Portal. For detailed information about subscribable reports, see the One Identity Manager Report Subscriptions Administration Guide.

**NOTE:**

- This function is only available if the Report Subscription Module is installed.
- The task is only available if a permissions group is assigned to the application role (or a parent application role).
- Subscribable reports cannot be assigned to the application roles Base roles | Employee Managers, Base roles | Everyone (Lookup) and Base roles | Everyone (Change).

1. In Manager in the One Identity Manager Administration category, select the Application role.
2. Select Assign subscribable reports in the task view.
3. Assign reports in Add assignments.
   - OR -
   In the Remove assignments area, remove the reports.
4. Save the changes.
Assigning extended properties to application roles

Extended properties are meta objects that cannot be mapped directly in One Identity Manager, for example, operating codes, cost codes or cost accounting areas. For detailed information about using extended properties, see the One Identity Manager Identity Management Base Module Administration Guide.

To specify extended properties for an application role

1. In Manager in the One Identity Manager Administration category, select the Application role.
2. Select Assign extended properties in the task view.
3. Assign extended properties in Add assignments.
   
   **TIP:** In the Remove assignments area, you can remove the assignment of extended properties.

   **To remove an assignment**
   
   • Select the extended property and double click .

4. Save the changes.

Generating assignment resources for application roles

It is possible to create assignment resources for individual application roles. This means you can limit assignment resources to individual application roles in the Web Portal. When the assignment resource is requested, it is no longer necessary to select the application role as well. The application role is automatically a part of the assignment request. For detailed information about assignment requests, see the One Identity Manager IT Shop Administration Guide.

To limit an assignment resource to one application role

1. In Manager in the One Identity Manager Administration category, select the Application role.
2. Select the Create assignment resource task.
   
   This starts a wizard, which takes you through adding an assignment resource.

Reports about application roles

One Identity Manager makes various reports available containing information about the selected base object and its relations to other One Identity Manager database objects. The following reports are available for application roles.
### Table 14: Reports about Application Roles

<table>
<thead>
<tr>
<th>Report</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview of all assignments</td>
<td>This report identifies all departments, cost centers, locations, business roles or IT Shop structures in which employees from the selected application role are also members. For detailed information about analyzing role memberships, see the <em>One Identity Manager Identity Management Base Module Administration Guide</em>.</td>
</tr>
<tr>
<td>Show historical memberships</td>
<td>This report lists all members of the selected application role and the length of their membership.</td>
</tr>
</tbody>
</table>
Granting One Identity Manager schema permissions

Permissions for accessing tables and columns of the One Identity Manager schema are themselves mapped in the schema through permissions groups. You can assign permissions groups to system users and to application roles.

The user's effective permissions depend on the authentication module used for logging into One Identity Manager tools.

- The permissions assigned to the system user are found from the permissions groups for logging into One Identity Manager tools with an authentication module that expects a defined system user.
- Dynamic system users are used for logging into One Identity Manager tools with role-based authentication modules. First, the employee memberships in the One Identity Manager application roles are determined during login. Assignments of permissions group to One Identity Manager application roles are used to determine which permissions groups apply to the employee. A dynamic system user is determined from these permissions groups that will be used for the employee's login.

The system user's effective permissions that are found are not saved in the One Identity Manager schema, but are determined when logging into One Identity Manager tools and then they are loaded.

Permissions groups are also used to control access to parts of the user interface, such as, menu items, forms, tasks and program functions. When a user logs into the One Identity Manager tools, menus, forms and methods are loaded depending on the system user's permissions groups, displaying a user interface customized for this system user. For more detailed information about editing the user interface, see the One Identity Manager Configuration Guide.

The One Identity Manager provides permissions groups and system users with a predefined user interface and edit permissions to the One Identity Manager schema's tables and columns. These predefined configurations are maintained by the schema installation and cannot be edited apart from a few properties.
Detailed information about this topic

- Predefined permissions groups and system users on page 35
- Rules for determining the valid permissions for tables and columns on page 38
- Processing permissions groups on page 40
- Editing system users on page 45
- Editing table permissions and column permissions on page 51
- Managing permissions to program features on page 62
- Displaying permissions for objects on page 59
- Displaying permissions for the current user on page 60
- Assigning permissions groups to applications on page 60

Related topics

- One Identity Manager Application roles on page 8
- One Identity Manager Authentication modules on page 69

Predefined permissions groups and system users

The One Identity Manager provides permissions groups and system users with a predefined user interface and special edit permissions to the One Identity Manager schema’s tables and columns. These predefined configurations are maintained by the schema installation and cannot be edited apart from a few properties.

Table 15: Predefined Permissions Groups

<table>
<thead>
<tr>
<th>Permissions group</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>QBM_BaseRights</td>
<td>The QBM_BaseRights permissions group defines the base rights that are required for a system user to log in to the administration tools. This permissions group is always assigned implicitly.</td>
</tr>
<tr>
<td>VID_Features</td>
<td>The VID_Features permissions group covers all program functions required for starting the One Identity Manager tools. The permissions group covers additional program functions for executing special functions in the One Identity Manager</td>
</tr>
<tr>
<td>VI_View</td>
<td>The VI_View permissions group has viewing permissions for all tables and columns of the One Identity Manager application data model.</td>
</tr>
<tr>
<td>Permissions group</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Permissions group VI_Everyone</strong></td>
<td>The <strong>VI_Everyone</strong> permissions group is assigned to form elements of the overview forms that use links to the corresponding menu items. These permissions groups also provide functions for Web Portal users.</td>
</tr>
<tr>
<td><strong>NOTE:</strong> Assign the permissions group to your custom system users such that the overview form is fully displayed to the users.</td>
<td></td>
</tr>
<tr>
<td>Permissions groups for the One Identity Manager system data model</td>
<td>These permissions groups have permissions for the One Identity Manager system data model tables and columns. These permissions groups are equipped with menu items, forms, tasks and program functionality which allows the application data to be edited, for example, with Designer editors.</td>
</tr>
<tr>
<td>Role-based permissions group <strong>VI_4_ALLUSER</strong></td>
<td>The <strong>VI_4_ALLUSER</strong> permissions group provides the base rights as well as menu items, forms, task and program functions to enable the application data to be edited with the Manager and the Web Portal. This permissions group is always assigned implicitly.</td>
</tr>
<tr>
<td>Role-based permissions group <strong>vi_4_ADMIN_LOOKUP</strong></td>
<td>The <strong>vi_4_ADMIN_LOOKUP</strong> permissions group has the viewing permissions for all tables and columns of the One Identity Manager application data model.</td>
</tr>
<tr>
<td><strong>NOTE:</strong> Assign viewing permissions of custom schema extensions to the permissions group. Assign viewing permissions of the module’s own tables and columns to the permissions group.</td>
<td></td>
</tr>
<tr>
<td>Role-based permissions group <strong>QER_OperationsSupport</strong></td>
<td>The <strong>QER_OperationsSupport</strong> permissions group has special permissions for working with the Operations Support Web Portal. The permissions group is assigned to the <strong>OperationsSupportWebPortal</strong> application. The permissions of the permissions group apply only in the Operations Support Web Portal.</td>
</tr>
<tr>
<td>Permissions group</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Role-based permissions groups</td>
<td>Role-based permissions groups have edit permissions for One Identity Manager application data model tables and columns. These permissions groups are equipped with menu items, forms, tasks and program functionality which allow the application data to be edited with the Manager and Web Portal. These permissions groups are linked to the One Identity Manager application roles and simplify administration of access permissions in the One Identity Manager role model.</td>
</tr>
</tbody>
</table>

### Table 16: Predefined system users

<table>
<thead>
<tr>
<th>System user</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dynamic system user</td>
<td>Dynamic system users are used for logging into One Identity Manager tools with role-based authentication modules. First, the employee memberships in the One Identity Manager application roles are determined during login. Assignments of permissions group to One Identity Manager application roles are used to determine which permissions groups apply to the employee. A dynamic system user is determined from these permissions groups that will be used for the employee’s login.</td>
</tr>
<tr>
<td>System user <strong>sa</strong></td>
<td>The <strong>sa</strong> system user is used exclusively by the One Identity Manager Service. This system user is not allocated a permissions groups but has all access permissions, tasks and program functionality.</td>
</tr>
</tbody>
</table>
| System user **viadmin** | The **viadmin** system user is the default system user in One Identity Manager. This system user can be used to compile and initialize the One Identity Manager database and for the first user login to the administration tools.  

| IMPORTANT: Do not use the **viadmin** system user in a live environment. Create your own system user with the appropriate permissions. |
| The system user has all of the specified permissions and the complete user interface. The system user implicitly receives the authorizations and user interface parts of the custom permissions groups. The system user has the permission to set up an employee as a One Identity Manager administrator for the role-based login. The system user is not a member of the application role themselves. |
| System user **Synchronization** | The **Synchronization** system user has the necessary permissions to set up and run target system synchronizations using an application server. |
| System user **viHelpdesk** | The **viHelpdesk** system user has the necessary permissions and the user interface to use the Manager to access the helpdesk resources of the One Identity Manager. |
System user | Description
--- | ---
viITShop | The viITShop system user has the necessary permissions and the user interface to use the Manager to access the IT Shop.

Related topics
- Deleting dynamic system users on page 50
- Permissions group dependencies on page 42
- Processing permissions groups on page 40

Rules for determining the valid permissions for tables and columns

When a system user is used to log into the system, the currently effective permissions for the objects are determined based on the permissions groups. The following rules are used to determine the resulting permissions:

- Permissions from hierarchical permissions groups are inherited from top to bottom. That means that a permissions group contains all the permissions belonging parent permissions groups.

- The number of objects is determined first for hierarchical permissions groups. Column permissions are decided afterwards. In some cases, this results in more permissions than are defined on individual permissions groups.

- A system user receives a permission when at least one of its permissions groups has the permission (directly or inherited).

- The limiting permissions conditions for all the system user's permissions groups are grouped together and used to determine a valid condition for each permission for viewing, editing, inserting and deleting an object.

- Fixed viewing permissions for the database system files are granted by the system, which are sufficient for logging a system user into One Identity Manager tools.

- A system user with read-only permissions only obtains viewing permissions to objects irrespective of any other permissions.

- If permissions are granted on a table for inserting, editing or deleting, viewing permissions are implicit.

- If permissions are granted on a column for inserting, editing or deleting, viewing permissions are implicit.

- If permissions are granted for a table, then viewing permissions are implicitly granted on the primary key column of the table.

- If viewing permissions are granted on a primary key column as a minimum, then viewing permissions are implicitly granted for references table, the primary key
column and the columns that are necessary on the referenced table for viewing according to the defined display pattern.

- Permissions for database views of the **Proxy** type also apply to the underlying tables.
- For database views of the **ReadOnly** type, only the viewing permissions apply irrespective of other permissions.
- If a table or column is disabled due to preprocessor conditions, permissions are not determined for those tables and columns. The table or column is considered not to exist.
- If a permissions group is disabled due to preprocessor conditions, permissions are not taken into account for this permissions group. The permissions group is considered not to exist.

**Example of Permissions Grouping using Permissions Groups**

The following example shows how to group permissions if the user is directly assigned in permissions groups and the permissions groups are not connected hierarchically.

A system user obtains permissions to the table `ADSAccount` through different permissions groups.

<table>
<thead>
<tr>
<th>Permissions group</th>
<th>Viewable</th>
<th>Editable</th>
<th>Insertable</th>
<th>Deletable</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>B</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

In addition, it is granted permissions to the table `LDAPAccount` through these permissions groups.

<table>
<thead>
<tr>
<th>Permissions group</th>
<th>Viewable</th>
<th>Editable</th>
<th>Insertable</th>
<th>Deletable</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>B</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Therefore, the system user has effectively the following permissions:

<table>
<thead>
<tr>
<th>Table</th>
<th>Viewable</th>
<th>Editable</th>
<th>Insertable</th>
<th>Deletable</th>
</tr>
</thead>
<tbody>
<tr>
<td>ADSAccount</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>LDAPAccount</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

**Example of Limiting Conditions**

A system user obtains viewing permissions to the table `Person` through different permissions groups:
This results in the following permissions for the individual employee objects.

<table>
<thead>
<tr>
<th>Permissions group</th>
<th>Viewing Condition</th>
<th>Column Viewing Permissions</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td></td>
<td>Lastname</td>
</tr>
<tr>
<td>B</td>
<td>Lastname like 'B%'</td>
<td>Lastname, Firstname, Entrydate</td>
</tr>
<tr>
<td>C</td>
<td>Lastname like 'Be%'</td>
<td>Lastname, Firstname, Gender</td>
</tr>
<tr>
<td>D</td>
<td>Lastname like 'D%'</td>
<td>Lastname</td>
</tr>
</tbody>
</table>

### Processing permissions groups

The One Identity Manager provides permissions groups with a predefined user interface and special edit permissions for the One Identity Manager schema’s tables and columns. In certain isolated cases, it may be necessary to define custom permissions groups. You need custom permissions groups, for example, if:

- The default permissions groups grant too many permissions,
- Selected default permissions groups are to be summarized to form a new permissions group,
- Additional role-based permissions groups are required for the custom application roles,
- Permissions for custom adjustments such as schema extensions, forms or menu structures.

When the One Identity Manager database is installed using the Configuration Wizard, custom permissions groups that you can use are already created.

- For non-role-based login, the permission groups `CCCViewPermissions` and `CCCEditPermissions` are created. Administrative system users are automatically added to these permissions groups.
- For role-based login, the permission groups `CCCViewRole` and `CCCEditRole` are created.

Permissions groups are managed in Designer in the Permissions | Permissions groups category. Here you will find an overview of edit permissions and user interface.
components that are assigned to individual permissions groups. In addition, the system users are displayed, which the permissions groups are assigned.

Use User & Permissions Group Editor to create and edit permissions groups in Designer. User & Permissions Group Editor displays the permissions groups in their hierarchy. Each permissions group is represented by a permissions group element. Each permissions group element has a tooltip. The contents of the tooltip is made up of the name and description of the permissions group.

You can run the following tasks:

- Editing the master data of a permissions group
- Defining new permissions group dependencies
- Copying Permissions Groups
- Creating a new permissions group

Related topics

- Predefined permissions groups and system users on page 35
- Permissions groups properties on page 41
- Editing the dependencies of permissions groups on page 43
- Copying permissions groups on page 44
- Creating permissions groups on page 45
- Managing permissions to program features on page 62

Permissions groups properties

Table 17: Permissions Group Properties

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permissions group</td>
<td>Name of the permissions group. Label custom permission groups with the prefix CCC.</td>
</tr>
<tr>
<td>Description</td>
<td>Detailed description of the permissions group’s purpose.</td>
</tr>
<tr>
<td>Remarks</td>
<td>Spare text box for additional explanation.</td>
</tr>
<tr>
<td>Preprocessor condition</td>
<td>You can add a preprocessor condition to permissions groups. This means that the permissions group is only effective when the condition is met.</td>
</tr>
<tr>
<td>Permissions group binary pattern</td>
<td>The permissions group binary pattern is used to calculate effective system user permissions. It is provided by the DBQueue Processor.</td>
</tr>
<tr>
<td>Only use for role-based authentication</td>
<td>This group includes permissions, form assignments, menu items and program functions for role-based authentication. The permissions group</td>
</tr>
</tbody>
</table>
## Permissions group dependencies

You can enable permissions and user interface components to be passed on from one permissions group to other permissions groups by structuring permissions groups hierarchically. This means that inheritance is top down within the hierarchy.

The following applies to permissions group dependencies:

- A role-based permissions group can inherit from role-based permissions groups and non role-based permissions groups.
- A non-role-based permissions group can inherit from non-role-based permissions groups. A non-role-based permissions group must not inherit from role-based permissions groups.

### Example

Two permissions groups are defined with the following permissions and user interface components.

<table>
<thead>
<tr>
<th>Permissions group</th>
<th>Permissions</th>
<th>User interface</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Viewing permissions</td>
<td>Menu structures and forms</td>
</tr>
<tr>
<td>B</td>
<td>Edit permissions</td>
<td>Task definitions</td>
</tr>
</tbody>
</table>

Permissions group B is arranged below permissions group A in the hierarchy and inherits from permissions group A. Consequently, a user of permissions group B has access to the viewing permissions and editing permissions as well as the menu structure, forms and task definitions.
Related topics

- Editing the dependencies of permissions groups on page 43

Editing the dependencies of permissions groups

You edit dependencies between permissions groups in the hierarchical view of the User & Permissions Group Editor. Permissions groups that are higher up in the hierarchy are displayed further to the right in the hierarchical view of User & Permissions Group Editor. When a permissions group is selected in the hierarchical view, the dependencies to other permissions groups are marked in color which also is also used to show the direction of inheritance.

Figure 1: Diagram of Permissions Group Hierarchy (Direction of Inheritance from Right to Left)

<table>
<thead>
<tr>
<th>Color</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Blue</td>
<td>The selected permissions group.</td>
</tr>
<tr>
<td>Purple</td>
<td>This permissions group is a child of the selected permissions group and directly inherits from the selected permissions group.</td>
</tr>
<tr>
<td>Light purple</td>
<td>This permissions group inherits indirectly from the selected permissions group over the hierarchy.</td>
</tr>
<tr>
<td>Red</td>
<td>This permissions group is a parent of the selected permissions group and passes inheritance to the selected permissions group.</td>
</tr>
<tr>
<td>Light red</td>
<td>This permissions group passes inheritance indirectly to the selected permissions group over the hierarchy.</td>
</tr>
<tr>
<td>Green</td>
<td>This permissions group does not inherit or pass inheritance to the selected permissions group.</td>
</tr>
</tbody>
</table>

Table 18: Meaning of Colors in the Hierarchical Representation

To specify dependencies of a permissions group

1. Select the Permissions | Permissions groups category in Designer.
2. Select the permissions group and start User & Permissions Group Editor using the Edit permissions group task.
3. In the hierarchical view of the permissions groups, select the permissions group and run one of the following actions.
• Select the **Inherit permissions from** context menu and select the permissions groups from which the selected permissions group is to inherit.

• Select the **Permissions inherited by** context menu and select the permissions groups to be included in the selected permissions group. Permissions subgroups inherit permissions from the selected permissions group.

## Copying permissions groups

The User & Permissions Group Editor provides a wizard for copying edit permissions and the user interface of an existing permissions group to a new permissions group.

**To copy a permissions group**

1. Select the **Permissions | Permissions groups** category in Designer.
2. Select the permissions group you want to copy and start the User & Permissions Group Editor with the task **Edit permissions group**.
3. Select the **Permissions groups | Copy permissions group** menu.
4. On the home page of the wizard for copying permissions groups, click **Next**.
5. On the **Select permissions group** page, enter the following information:
   - **Select permissions group to copy**: The permissions group is pre-selected.
   - **Copy name**: Name of the new permissions group. A name suggestion is already entered in the field which is made up from the customer prefix and the original permissions group name. You can alter this name but the customer prefix has to remain.
6. On the **Select copy options** page, specify which permissions group relations are to be copied. You can select multiple options. The following copy options are available.

### Table 19: Copy options for permissions groups

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Permissions</td>
<td>Enable this option to copy the table permissions and column permissions of the selected permissions group to the new permissions group.</td>
</tr>
<tr>
<td>User interface</td>
<td>Enable this option to copy the menu items, the forms and the task definitions of the selected permissions group to the new permissions group.</td>
</tr>
<tr>
<td>System user</td>
<td>Select this option if the system user should be copied to the new permissions group.</td>
</tr>
</tbody>
</table>

**NOTE:** Note here that predefined system users are not included in the new permissions group.
7. To start compiling, click **Next**.
   The copying process may take some time.
8. The **Copy permissions group** page shows the individual copy steps and any error messages. If the copy action is complete, click **Next**.
9. To end the wizard, click **Finish** on the last page.

**Related topics**
- Creating permissions groups on page 45
- Permissions groups properties on page 41
- Editing the dependencies of permissions groups on page 43
- Adding system users to permission groups on page 49
- Editing table permissions and column permissions on page 51

**Creating permissions groups**

*To create a permissions group*

1. In the Designer, select the **Permissions** category.
2. Start the User & Permissions Group Editor with the task **Show / edit permissions group**.
3. Add a new permissions group using the **Permissions groups | New** menu.
4. Edit the master data for the permissions group.
5. Save the changes.

**Related topics**
- Copying permissions groups on page 44
- Permissions groups properties on page 41
- Editing the dependencies of permissions groups on page 43
- Adding system users to permission groups on page 49

**Editing system users**

One Identity Manager provides various system users whose permissions are matched to the different tasks. Create your own system users if required. Add the system users to permissions groups, thereby granting the system users permissions for the tables and columns of the One Identity Manager data model, and make the user interface available.
The system user's effective permissions that are found are not saved in the One Identity Manager schema, but are determined when logging into One Identity Manager tools and then they are loaded.

When installing the One Identity Manager database using the Configuration Wizard, create an administrative system user that is added to non-role-based permissions groups and receives all permissions of the viadmin default system user.

System users are shown in Designer in the Permissions | System users category. You will see an overview of the permissions groups that are assigned to each individual system user. Use the User & Permissions Group Editor to create and edit your system user in Designer.

You can run the following tasks:

- Create a new system user, such as an administrative system user or a system user for service accounts
- Configuration of password settings for system users
- Adding a system user to permissions groups
- Determining which employees use a system user

NOTE: You cannot edit dynamic system users.

Related topics

- Predefined permissions groups and system users on page 35
- Creating system users on page 46
- System users’ passwords on page 47
- System user properties on page 48
- Adding system users to permission groups on page 49
- Which employees use the system user? on page 50
- Deleting dynamic system users on page 50

Creating system users

To create a new system user

1. In the Designer, select the Permissions category.
2. Start the User & Permissions Group Editor with the task Show / edit permissions group.
3. Add a new system user using the User | New menu.
4. Edit the system user's master data.
5. Add the system user to permissions groups.
6. Save the changes.
**NOTE:** You can create an administrative system user in User & Permissions Group Editor using the Create administrator menu. Administrative system users are automatically added to all non role-based permissions groups.

### Related topics
- Predefined permissions groups and system users on page 35
- System users’ passwords on page 47
- System user properties on page 48
- Adding system users to permission groups on page 49
- Deleting dynamic system users on page 50
- Which employees use the system user? on page 50
- One Identity Manager Authentication modules on page 69

### System users’ passwords

The **One Identity Manager password policy** is used for logins on the One Identity Manager with a system user. This password policy defined the settings for the system user passwords (DialogUser.Password and Person.DialogUserPassword) as well as the access code for a one off log in on the Web Portal (Person.Passcode).

If necessary, adjust the password policy to your requirements in Designer. For detailed information about editing password policies, see *One Identity Manager Operational Guide*.

**NOTE:** The **One Identity Manager password policy** is marked as the default policy. This password policy is applied if no other password policy can be found for employees, user accounts or system users.

To prevent passwords expiring for service accounts, for example, you can enable the **Password never expires** (DialogUser.PasswordNeverExpires) option in Designer for the respective system users.

### Related topics
- System user properties on page 48
## System user properties

### Table 20: Properties of a system user

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System user</td>
<td>Name of the system user for logging in to the administration tools.</td>
</tr>
<tr>
<td>Password and password confirmation</td>
<td>Password with which the system logs into the administration tools.</td>
</tr>
<tr>
<td>Password last changed</td>
<td>Date of last password change.</td>
</tr>
<tr>
<td>Password never expires</td>
<td>Specifies whether the password expires. Enable the option for service accounts, for example, to prevent the password from expiring. This option overwrites the maximum age of the password.</td>
</tr>
<tr>
<td>Remarks</td>
<td>Spare text box for additional explanation.</td>
</tr>
<tr>
<td>Read-only</td>
<td>Set the option if a system user is a member of all permissions groups, but should only have viewing permissions for the objects. This results in overwriting all other edit permissions that the system user obtains through permissions group memberships.</td>
</tr>
<tr>
<td>Logins</td>
<td>Logins with which the system user can log in to the One Identity Manager tools. Enter the login in the form: Domain\User. This information is required if the Account based system user authentication module is used to log into the One Identity Manager tools.</td>
</tr>
<tr>
<td>Administrative user</td>
<td>Specifies whether this is an administrative system user. Administrative system users are automatically added to all non role-based permissions groups.</td>
</tr>
<tr>
<td>Service account</td>
<td>Specifies whether this is a system user that is used by a service account. This system user is not allocated a permissions groups but has all access permissions, tasks and program functionality.</td>
</tr>
<tr>
<td>External password management</td>
<td>Specifies whether the system user password is determined by an external password management system. You cannot change the password in One Identity Manager. The determination of the system user password must be customized.</td>
</tr>
</tbody>
</table>

**NOTE:** You can create an administrative system user in User & Permissions Group Editor using the Create administrator menu.

### Related topics

- [System users’ passwords](#) on page 47
Adding system users to permission groups

Add the system user to permissions groups, thereby granting the system user permissions for the tables and columns of the One Identity Manager data model and make the user interface available.

NOTE:

- You cannot add system users to role-based permissions groups. Dynamic system users are calculated for role-based login.
- Administrative system users are automatically added to all non role-based permissions groups.
- The QBM_BaseRights permissions group defines the base permissions that are required for a system user to log in to the administration tools. This permissions group is always assigned implicitly.
- The viadmin system user has all of the specified permissions and the complete user interface. The system user implicitly receives the authorizations and user interface parts of the custom permissions groups.

A system user’s memberships of permissions groups are presented in User & Permissions Group Editor. Use the Options | Display permissions group inheritance menu to specify whether to display the direct and inherited memberships of permissions groups for system users.

Figure 2: System User Permissions Group Memberships

Table 21: Meaning of Icons in the Hierarchical Display

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>![ ]</td>
<td>The selected system user is not assigned to this permissions group.</td>
</tr>
<tr>
<td>![ ]</td>
<td>The selected system user is assigned to this permissions group.</td>
</tr>
<tr>
<td>![ ]</td>
<td>The selected system user is indirectly assigned to this permissions group.</td>
</tr>
<tr>
<td>![ ]</td>
<td>The selected system user is directly and indirectly assigned to this permissions group.</td>
</tr>
</tbody>
</table>

To assign a system user to a permissions group

1. In Designer, select the Permissions | System user category.
2. Select a system user and start the User & Permissions Group Editor with the Edit system user task.
3. Select the required permissions group in the hierarchical view. By clicking on the icon you add or delete the selected system user to or from a permissions group.

TIP: To assign a system user to several permissions groups, use the User | Permissions groups menu.

Related topics
- Deleting dynamic system users on page 50

Which employees use the system user?

Employees obtain a system user directly from their master data or dynamically through their One Identity Manager application roles.

To display which employees are assigned to a system user
1. In Designer, select Permissions | System users.
2. Select a system user and start the User & Permissions Group Editor with the Edit system user task.
3. Select View | One Identity Manager employees.

NOTE: You cannot change the assignments in this view.

Deleting dynamic system users

NOTE: If no role-based logins of employees who use dynamic system users are performed for some time, you should delete the dynamic system users for performance reasons. A new dynamic system user is created during the next role-based employee login.

To delete system users
- In Designer, enable the Common | DynamicUserLifetime configuration parameter and enter the maximum retention period in days for dynamic system users.

If the configuration parameter is enabled, system users whose retention period has expired are deleted from the database as part of the daily maintenance tasks.
Editing table permissions and column permissions

You can edit permissions in the Designer using the Permissions Editor. You can also simulate the permissions for the individual system users in Permissions Editor.

With the Permissions Editor you can:

- Assign permissions for custom tables and columns to custom permissions groups
- Assign permissions for predefined tables and columns in the One Identity Manager schema to custom permissions groups
- Assign permissions for custom tables and columns to predefined permissions groups

The permissions for predefined permission groups to predefined tables and columns in the One Identity Manager schema cannot be changed.

For custom schema extensions, use the Schema Extension program, specify a permissions group with read and write permissions as well as a permissions group with read permissions only. This make initial access to the schema extensions possible with One Identity Manager administration tools.

Detailed information about this topic

- Rules for determining the valid permissions for tables and columns on page 38
- Displaying the permissions of a permissions group on page 51
- Displaying permissions for tables on page 52
- Editing table properties on page 53
- Editing column permissions on page 55
- Copying table permissions and column permissions on page 56
- Simulating permissions for system users on page 57

Displaying the permissions of a permissions group

To display all permissions for a permission group

1. In the Designer, select the Permissions category.
2. Start the Permissions Editor using the Edit permissions task.
3. In the Permissions Editor toolbar in the Permissions group menu, select the permissions group for which you want to display permissions.
The tables and columns of the One Identity Manager schema and the permissions of the selected permissions group are displayed in the upper area of Permissions Editor. Use the following Permissions Editor options to adjust the layout.

- To display tables with permissions first, enable the Options | Permissions sort order menu.
- To display disabled tables and columns, enable the Options | Show disabled tables menu.
- To use the display names of the tables and columns, enable the Options | Display name menu.
- To limit the display of the tables, use the Show system tables, Show non-system tables and Show all tables menu items in the Options menu. Alternatively, use the Define filter or Manage filters menu items to define your own user-defined filters for displaying the tables and columns.

For detailed information about creating user-defined filters in Designer, see the One Identity Manager User Guide for One Identity Manager Tools User Interface.

### Displaying permissions for tables

In the Summary of all permissions view in Permissions Editor, the permissions groups that have permissions for a table or column are displayed. The permissions in this view cannot be edited.

**NOTE:** To display the Summary of all permissions view, go to Permissions Editor and enable View | Object permissions. The view is displayed in the lower area of Permissions Editor.

#### To display all permissions for a table and its columns

1. In Designer in the Permissions | By tables category, select the table.
2. Start the Permissions Editor using the Edit permissions for table task.
   - The Summary of all permissions view displays the permissions groups that have permissions for the selected table.
   
   **TIP:** To display a permissions filter completely, click a condition in the view.

3. (Optional) To display all permissions for a column, open the table entry in the upper area of Permissions Editor and select a column.
   - The Summary of all permissions view displays the permissions groups that have permissions for the selected column.
Editing table properties

Use the table permissions to grant the permissions to display, insert, edit and delete the objects. You can define conditions to further limit the permissions for the objects. You can use the conditions, for example, to link the editability of the employees to their last names. For instance, a user can be given read access only to the employees whose last names begin with A-F, whereas he/she can edit employees with last names beginning with G-Z.

NOTE: The permissions are always edited in Permissions Editor for the permissions group that you selected in the Permissions Editor toolbar in the Permissions group menu. If you wish to grant permissions for another permissions group, first select this permissions group in the menu and then edit the permissions.

To edit the permissions for a table for a permissions group

1. In the Designer, select the Permissions category.
2. Start the Permissions Editor using the Edit permissions task.
3. In the Permissions Editor toolbar in the Permissions group menu, select the permissions group for which you want to grant the permissions.
4. Select the table at the top of the Permissions Editor.
   TIP: Use Shift + select or Ctrl + select to select multiple tables.
5. Edit the permissions for the permissions group in the Permissions area.
   • To insert new permissions, select the New context menu and enable the associated check boxes. You can grant the following permissions.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewable</td>
<td>The table data is displayed.</td>
</tr>
<tr>
<td>Insertable</td>
<td>New data can be added to the table.</td>
</tr>
<tr>
<td>Editable</td>
<td>Table data can be edited.</td>
</tr>
<tr>
<td>Deletable</td>
<td>Table data can be deleted.</td>
</tr>
</tbody>
</table>

NOTE: If you grant the Insertable, Editable or Deletable permissions, the Viewable permission is also granted.

• To withdraw a permission, disable the associated checkbox.
• To withdraw all permissions for a table, use the Delete.
6. (Optional) To specify other conditions for table permissions, go to the lower part of the Permissions Editor and change to the Group permissions for table and view and select the Permissions filter.
NOTE: You can only define permissions filters for the tables of the application data model.

- Enter the conditions as valid WHERE clauses for database queries. You can enter the following permissions filters.

<table>
<thead>
<tr>
<th>Permissions filter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewing Condition</td>
<td>Limiting condition for displaying data sets.</td>
</tr>
<tr>
<td>Edit condition</td>
<td>Limiting condition for editing data sets.</td>
</tr>
<tr>
<td>Insert condition</td>
<td>Limiting condition for inserting data sets.</td>
</tr>
<tr>
<td>Deletion condition</td>
<td>Limiting condition for deleting data sets.</td>
</tr>
</tbody>
</table>

**Example for permissions filters**

A user should be able to see all employees, but only edit the employees whose last names begin with B. Specify the limiting edit condition as follows, for example:

```
Lastname like 'B%'
```

**TIP:** Use the SQL check button to test the condition. This checks the syntax. The number of objects that match the condition is returned.

**Related topics**

- Editing column permissions on page 55
- Copying table permissions and column permissions on page 56
Editing column permissions

**IMPORTANT:**

- If you grant permissions to columns, you must also grant the permissions to the tables. For example, a column is only viewable if the table is also viewable.
- To insert objects into a table, the **Insert** permissions is required for at least the required fields in the table.
- **NOTE:** If you grant the **Insert** or **Edit** permissions, the **View** permission is also granted.
- Use the column definition to conditionally remove viewing permissions from scripts or create edit permissions for a column. If the return value is **False**, the permissions are removed. For more information about editing column definitions, see the *One Identity Manager Configuration Guide*.

**NOTE:** The permissions are always edited in Permissions Editor for the permissions group that you selected in the Permissions Editor toolbar in the **Permissions group** menu. If you wish to grant permissions for another permissions group, first select this permissions group in the menu and then edit the permissions.

To modify the permissions for a column for a permissions group

1. In the Designer, select the **Permissions** category.
2. Start the Permissions Editor using the **Edit permissions** task.
3. In the Permissions Editor toolbar in the **Permissions group** menu, select the permissions group for which you want to grant the permissions.
4. Select the table at the top of the Permissions Editor and select the column.

   **TIP:** Use **Shift + select** or **Ctrl + select** to select multiple columns.
5. Edit the permissions for the permissions group in the **Permissions** area.

   - To insert new permissions, select the **New** context menu and enable the associated check boxes. You can grant the following permissions.

<table>
<thead>
<tr>
<th>Permissions</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Viewable</td>
<td>The column is displayed.</td>
</tr>
<tr>
<td>Editable</td>
<td>The values in the columns can be changed</td>
</tr>
<tr>
<td>Insertable</td>
<td>The value in the column can be edited when a new data record. Once the data record has been saved it can no longer be edited.</td>
</tr>
<tr>
<td>Permissions</td>
<td>Meaning</td>
</tr>
<tr>
<td>-------------</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td>For example, when an Active Directory User is created, an Active Directory Container is defined. Because this is a key field the Active Directory Container cannot be changed after the object has been saved.</td>
</tr>
</tbody>
</table>

- To withdraw a permission, disable the associated checkbox.
- To withdraw all permissions for a column, use the **Delete** context menu.

**Related topics**
- Editing table properties on page 53
- Copying table permissions and column permissions on page 56

**Copying table permissions and column permissions**

To transfer the permissions of a permissions group quickly from one table to another table, you can copy the table permissions and column permissions. Two methods are provided in the Permissions Editor to do this:

- **Copy** and **Insert**: This method copies the permissions of the source table (source column) to a permissions group. The permissions are copied for the permissions group that you selected in the Permissions Editor toolbar in the **Permissions group** menu.
  
  All copied permissions are inserted for the target table (target column). Any existing rights for the target table (target column) remain unaffected.

- **Copy all permissions** and **Paste all permissions**: This method copies all source table (source column) permissions. The initial selection of the permissions group in Permissions Editor makes no difference here. All permissions from all permissions groups for the source table (source column) are applied.
  
  All copied permissions are inserted for the target table (target column). Existing permissions for target table (target column) that do not exist for the source table (source column) are removed from the target table (target column).

**To copy the permissions of a permissions group**

1. In the Designer, select the **Permissions** category.
2. Start the Permissions Editor using the **Edit permissions** task.
3. In the Permissions Editor toolbar in the **Permissions group** menu, select the permissions group for which you want to grant the permissions.
4. To transfer the table permissions.
a. Select the table at the top of the Permissions Editor from which you want to transfer the permissions.
b. Use the **Copy** context menu to copy the permissions to the buffer.
c. Select the table at the top of the Permissions Editor for which you want to transfer the permissions.
d. Use the **Insert** context menu to insert the permissions.
e. If necessary, repeat step c) and d) for other tables.

5. To transfer the column permissions
a. Select the table at the top of the Permissions Editor and select the column from which you want to transfer permissions.
b. Use the **Copy** context menu to copy the permissions.
c. Select the table at the top of the Permissions Editor and select the column for which you want to copy permissions.
d. Use the **Insert** context menu to insert the permissions.
e. If necessary, repeat step c) and d) for other columns.

**Related topics**
- Editing table properties on page 53
- Editing column permissions on page 55

**Simulating permissions for system users**

By simulating the permissions in Permissions Editor, you can see which permissions a system user has based on his or her permissions group. You can specify which permissions groups of a system user to include in the simulation. The result displayed shows which of the selected permissions groups has which table permissions and column permissions. Effective permissions for the system user are also displayed.

**NOTE:** Simulation mode remains active until you end it. In simulation mode, you can edit permissions group permissions and update simulation data.

**To run a permissions simulation**

1. In the Designer, select the **Permissions** category.
2. Start the Permissions Editor using the **Edit permissions** task.
3. From the **Simulation | Start simulation** menu, start the simulation wizard.
4. On the start page of the wizard, click **Next**.
5. On the **Simulation base configuration** page, select the following data.
   - **User**: Select the system user whose permissions you want to simulate.
   - **Direct groups**: Use this button to select all permissions groups that are directly assigned to the system user.
   - **All groups**: Use this button to select all permissions groups that are directly assigned to the system user as well as all permissions groups that the system user inherits indirectly.
   - **Permissions groups**: Select individual permissions groups directly. Use **Ctrl + select** to select multiple tables.

6. On the **Simulation configuration** page, specify the tables for which the permissions are simulated.
   - In the **Selected tables** area, all tables of the One Identity Manager schema are selected. If necessary, limit the selection to individual tables. Click **None** to undo the selection. Use **Shift + select** to select individual tables.
   - Using the **Context table** menu, you can specify a table from whose view implicit permissions to the display values of the foreign key columns are assigned.
     
     Example:
     
     For the **Employee** table, viewing permissions to the **UID_Org** column have been assigned. As a result, viewing permissions are implicitly assigned for columns of the **Org** table, which are used as a display template, for example, **Org.Ident_Org**.
     
     To simulate this example, select the **Employee table** under **Context table** and the **Org table** under **Selected tables**.

7. The processing progress of the simulation is displayed on the **Simulation** page. The simulation process can take some time.

8. To end the wizard, click **Finish** on the last page.

   After you complete simulation wizard, the system user's effective table permissions and column permissions are displayed in the upper area of the Permissions Editor in the **Simulation** area.

9. To determine which table permission or column permission results from which of the system user's permissions groups, select the table or column in the upper area of the Permissions Editor.

   The permissions and permissions groups are displayed in the **Permissions simulation** view in the lower area of Permissions Editor.

10. To end the simulation mode, select the **Simulation | End simulation** menu.

    The simulation data is deleted and the **Permissions simulation** view is closed.
Displaying permissions for objects

You can display object properties and permissions in the One Identity Manager tools.

**To show extended object properties**

- Select the object and open the **Properties** context menu.

On **General**, you can see the object's general properties, for example, ID, status, or primary key.

All the object columns are displayed in a grid on **Properties** with their values. You can choose between a simple column view and the advanced view with additional data for column definitions.

**Table 25: Icon used for Column Properties**

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>![icon]</td>
<td>Required field.</td>
</tr>
<tr>
<td>![icon]</td>
<td>No viewing permissions.</td>
</tr>
<tr>
<td>![icon]</td>
<td>No edit permissions.</td>
</tr>
</tbody>
</table>

On **Access permissions**, you can see which permissions are valid for an object based on permissions groups. The first entry shows the basic permissions for the table. The permissions for this particular object are displayed beneath that. The other entries show the column permissions.

**TIP:** Double-click the table entry, the object entry, or a column entry to display the permissions group from which the permissions were determined.

**Table 26: Icon used for permissions**

<table>
<thead>
<tr>
<th>Icon</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>![icon]</td>
<td>Permissions exist.</td>
</tr>
<tr>
<td>![icon]</td>
<td>Permissions have been removed by the object layer.</td>
</tr>
<tr>
<td>![icon]</td>
<td>Permissions limited by conditions.</td>
</tr>
</tbody>
</table>
Displaying permissions for the current user

To get more information about the current user
- To display additional user information, double-click the ✉ icon in the status bar.

Table 27: Extra Information about the Current User

<table>
<thead>
<tr>
<th>Property</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>System user</td>
<td>Name of system user</td>
</tr>
<tr>
<td>Authenticated by</td>
<td>Name of the authentication module used for logging in.</td>
</tr>
<tr>
<td>Employee UID (UserUID)</td>
<td>Unique ID for the current user’s employee if an employee related authen-</td>
</tr>
<tr>
<td></td>
<td>tication module is used to log in.</td>
</tr>
<tr>
<td>Read-only</td>
<td>The system user has only has read permissions. Modification to data are</td>
</tr>
<tr>
<td></td>
<td>not possible.</td>
</tr>
<tr>
<td>Dynamic user</td>
<td>The current user uses a dynamic system user. Dynamic system users are</td>
</tr>
<tr>
<td></td>
<td>applied when a role-based authentication module is used.</td>
</tr>
<tr>
<td>Remarks</td>
<td>More details about the system user in use.</td>
</tr>
<tr>
<td>Permissions groups</td>
<td>Permissions groups that are assigned to the system user. Which user</td>
</tr>
<tr>
<td></td>
<td>interface and editing permissions apply depend on the permissions</td>
</tr>
<tr>
<td></td>
<td>groups.</td>
</tr>
<tr>
<td>Program functions</td>
<td>Program functions assigned to the system user The menu items and</td>
</tr>
<tr>
<td></td>
<td>functions available depend on the program functions.</td>
</tr>
</tbody>
</table>

Assigning permissions groups to applications

If you assign a permissions group to an application, the permissions of the group apply only to this application. When a user logs on to the application, they receive the permissions of the permissions group in addition to their own permissions.

To assign a permissions group to an application
1. Select the Permissions | Permissions groups | Role based permissions groups category in Designer.
2. Select the View | Select table relations menu item and enable the
DialogGroupInProductLimited table.

3. In the List Editor, select the permissions group.

4. Assign the application in the Applications edit view.

For detailed information about applications in One Identity Manager, see the One Identity Manager Configuration Guide.
Managing permissions to program features

Program functions are part of the permission model in One Identity Manager. They allow you to enable and disable features. Program functions are not assigned to single users but to permissions groups. The set of program functions defined for a user is determined by his or her permissions groups and their program functions.

The One Identity Manager tools can only be started if the user has the relevant program function permissions. Furthermore, some functions in the One Identity Manager tools are available only if the program functions are assigned to the current user. This includes data export from the Manager, calling the SQL Editor in the Designer or showing DBQueue Processor information in all programs, as examples.

Detailed information about this topic

- Program functions for starting the One Identity Manager tools on page 62
- Displaying permissions for the current user on page 60
- Assigning program function to permissions groups on page 65
- Permissions for executing scripts on page 65
- Permissions for executing methods on page 66
- Permissions for triggering processes on page 67
- Modifying permissions for executing actions in the Launchpad on page 68

Program functions for starting the One Identity Manager tools

The One Identity Manager tools can only be started if the user has the relevant program function permissions. The following program functions allow the One Identity Manager tools to be started.
To make the program function available to users

- In the Designer under Permissions | Program functions, check which permissions group contains the required program function and assign the program functions to other permissions groups as necessary.
- For non-role-based logins: Add the system user to the permissions group in the Designer under Permissions | System users.
- For role-based logins: Ensure that the user is assigned to the application role that contains the program function.

Table 28: Program functions for starting the One Identity Manager tools

<table>
<thead>
<tr>
<th>Program function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApplicationStart_Analyzer</td>
<td>Allows the program Analyzer (Analyzer.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_ApiDesigner</td>
<td>Allows the program API Designer (ApiDesigner.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_ConfigWizard</td>
<td>Allows the program Configuration Wizard (ConfigWizard.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_CryptoConfig</td>
<td>Allows the program Crypto Configuration (CryptoConfig.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_DataImporter</td>
<td>Allows the program Data Import (DataImporter.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_DBClone</td>
<td>Allows the program (DBClone.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_DBComparer</td>
<td>Allows the program (DBComparer.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_DBCompiler</td>
<td>Allows the program Database Compiler (DBCompiler.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_Designer</td>
<td>Allows the program Designer (Designer.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_JobQueueInfo</td>
<td>Allows the program Job Queue Info (JobQueueInfo.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_LaunchPad</td>
<td>Allows the program Launchpad (LaunchPad.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_LicenseMeter</td>
<td>Allows the program License Meter (LicenseMeter.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_Manager</td>
<td>Allows the program Manager (Manager.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_ObjectBrowser</td>
<td>Allows the program Object Browser (ObjectBrowser.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_Report</td>
<td>Allows the program Report Editor (ReportEdit2.exe) to be started.</td>
</tr>
</tbody>
</table>
### Program function

<table>
<thead>
<tr>
<th>Program function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReportEdit</td>
<td>Started.</td>
</tr>
<tr>
<td>ApplicationStart_SchemaExtension</td>
<td>Allows the program Schema Extension (SchemaExtension.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_ServerInstaller</td>
<td>Allows the program Server Installer (ServerInstaller.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_SoftwareLoader</td>
<td>Allows the program Software Loader (SoftwareLoader.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_SynchronizationEditor</td>
<td>Allows the program Synchronization Editor (SynchronizationEditor.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_SystemDebugger</td>
<td>Allows the program System Debugging (SystemDebugger.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_Transporter</td>
<td>Allows the program Database Transporter (Transporter.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_WebDesignerCompiler</td>
<td>Allows the program (VI.WebDesigner.CompilerCmd.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_WebConfig</td>
<td>Allows the program Web Designer Configuration Editor (WebConfigEditor.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_WebDesigner</td>
<td>Allows the program Web Designer (WebDesigner.exe) to be started.</td>
</tr>
<tr>
<td>ApplicationStart_WebDesignerInstall</td>
<td>Allows the program Web Installer (WebDesigner.Installer.exe) to be started.</td>
</tr>
</tbody>
</table>

### Related topics
- Assigning program function to permissions groups on page 65
- Adding system users to permission groups on page 49
- Assigning employees to application roles on page 28

### Displaying the current user's program functions

To identify the program functions available to the current user:

- To display user information, double-click the icon in the program status bar 🔄
  The Program functions tab shows the program functions that are available.
Assigning program function to permissions groups

To assign a program function to a permissions group

1. Select Permissions | Program functions in Designer.
2. Select the View | Select table relations menu item and enable the DialogGroupHasFeature table.
3. Select the program function in the List Editor.
4. Assign the Permissions groups in the Permissions groups edit view.
5. Save the changes.

Related topics

- Adding system users to permission groups on page 49

Permissions for executing scripts

The basic permissions for executing scripts are granted to the logged in user via the program feature Allow the starting of arbitrary scripts from the frontend (Common_StartScripts).

If a script also uses a program function (table QBMScriptHasFeature), the user can only run this script if the necessary program function is granted to him. An error occurs if the user does not own this program function and tries to run it.

To control execution of a script using a program function

1. Create a new program function.
   a. Select Permissions | Program functions in Designer.
   b. Select Object | New.
   c. Enter the following information:
      - Program function: Name of the program function.
      - Description: Short description of the program function.
      - Function group: Property for grouping program functions.
2. Connect the program function with the scripts that the user will trigger.
   a. Select Permissions | Program functions in Designer.
   b. Select the View | Select table relations menu item and enable the QBMScriptHasFeature table.
c. Select the newly created program function in List Editor.
d. Assign the scripts in the Scripts edit view.

3. Assign the required program functions to the custom permissions group whose systems users will trigger these scripts.
   a. Select Permissions | Program functions in Designer.
   b. Select the View | Select table relations menu item and enable the DialogGroupHasFeature table.
   c. Select your newly created program function in List Editor.
   d. In the List Editor, use Ctrl+Selection to select your newly created program function and the function **Allow the starting of arbitrary scripts from the frontend** (Common_StartScripts).
   e. Assign the permissions group in the Permissions groups edit view.

Related topics
- Processing permissions groups on page 40

Permissions for executing methods

If a task definition is assigned a program function (QBMMethodHasFeature), users can only execute this task if they are also assigned the necessary program function. An error occurs if the user does not own this program function and tries to run it. Program functions are not assigned to single users but to permissions groups. All users that are assigned to these groups can use the program function.

To make a task definition available to users using a program function

1. Create a new program function.
   a. Select Permissions | Program functions in Designer.
   b. Select Object | New.
   c. Enter the following information:
      - **Program function**: Name of the program function.
      - **Description**: Short description of the program function.
      - **Function group**: Property for grouping program functions.

2. Connect the program function with the task definition events that the user will trigger.
   a. Select Permissions | Program functions in Designer.
   b. Select the View | Select table relations menu item and enable the QBMMethodHasFeature table.
c. Select the newly created program function in List Editor.
d. In the Tasks edit view, assign the task definitions.

3. Assign the required program functions to the custom permissions group whose systems users will trigger these tasks.
   a. Select Permissions | Program functions in Designer.
   b. Select the View | Select table relations menu item and enable the DialogGroupHasFeature table.
   c. Select your newly created program function in List Editor.
   d. Assign the permissions group in the Permissions groups edit view.

Related topics
- Processing permissions groups on page 40

Permissions for triggering processes

The basic permissions for triggering processes are granted to the logged in user via the programm feature **Allow to trigger any events from the frontend** (Common_TriggerEvents).

In One Identity Manager, the triggering of events on stored processes is linked to the permissions concept. Users can only trigger events on objects like this if they own edit permissions for them. This can lead to table users who only have viewing permissions not being able to trigger additional events for processes.

In this case, it is possible to connect the object events (QBMEvent table) with a program function (QBMFeature table). An event (JobEventGen table), which is defined for a process, is linked with an object event (JobEventGen.UID_QBMEvent column). If the object event is assigned a program function (QBMEventHasFeature table), users that own this program function can trigger the associated object event and therefore the process, depending on their permissions.

**To control triggering a process through a program function**

1. Create a new program function.
   a. Select Permissions | Program functions in Designer.
   b. Select Object | New.
   c. Enter the following information:
      - **Program function**: Name of the program function.
      - **Description**: Short description of the program function.
      - **Function group**: Property for grouping program functions.
2. Connect the program function with object events that the user will trigger.
a. Select **Permissions | Program functions** in Designer.
b. Select the **View | Select table relations** menu item and enable the **QBMEventHasFeature** table.
c. Select the newly created program function in List Editor.
d. Assign the object events in the **Object events** edit view.

3. Assign the required program functions to the custom permissions group whose systems users will trigger these events.
   a. Select **Permissions | Program functions** in Designer.
   b. Select the **View | Select table relations** menu item and enable the **DialogGroupHasFeature** table.
   c. In List Editor, use **Ctrl+Selection** to select your newly created program function and the function **Allow to trigger any events from the frontend** (Common_TriggerEvents).
   d. Assign the permissions group in the **Permissions groups** edit view.

Related topics

- Processing permissions groups on page 40

**Modifying permissions for executing actions in the Launchpad**

One Identity Manager supplies a number of Launchpad actions that you can use to start applications via the Launchpad. You can also start your own applications over the Launchpad.

If some actions in the Launchpad should not be made available to all users, you can manage the permissions by assigning Launchpad to program functions (table **QBMLaunchActionHasFeature**). Only tasks containing actions that the user’s program function permissions permit him to run are shown in the Launchpad.

**To assign a program function to Launchpad actions**

1. Select **Permissions | Program functions** in Designer.
2. Select the **View | Select table relations** menu item and enable the **QBMLaunchActionHasFeature** table.
3. Select the program function in the List Editor.
4. Assign the actions in the **Actions** edit view.
5. Save the changes.
One Identity Manager Authentication modules

One Identity Manager uses different authentication modules for logging in to administration tools. Authentication modules identify the system users to be used and load the user interface and database resource editing permissions depending on their permission group memberships.

Before you can use an authentication module for logging on, the following prerequisites must be fulfilled:

1. The authentication module must be enabled.
2. The authentication module must be assigned to the application.
3. The assignment of the authentication module to the application must be enabled.

This allows you to log in to the assigned application using this authentication module. Ensure that users found through the authentication module also have the required program function to use the program.

**NOTE:** After the initial schema installation, only the **System user** and **Component authenticator** authentication modules and the role-based authentication modules are enabled in One Identity Manager.

**NOTE:** Authentication modules are defined in the One Identity Manager modules and are not available until the modules are installed.

System users

<table>
<thead>
<tr>
<th>Credentials</th>
<th>The system user’s identifier and password.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prerequisites</td>
<td>• The system user with permissions exists in the One Identity Manager database.</td>
</tr>
<tr>
<td>Set as default</td>
<td>Yes</td>
</tr>
<tr>
<td>Single sign-on</td>
<td>No</td>
</tr>
<tr>
<td>---------------</td>
<td>----</td>
</tr>
<tr>
<td>Front-end login allowed</td>
<td>Yes</td>
</tr>
<tr>
<td>Web Portal login allowed</td>
<td>No</td>
</tr>
<tr>
<td>Remarks</td>
<td>The user interface and the write permissions are loaded through the system user. Data modifications are attributed to the system user.</td>
</tr>
</tbody>
</table>

**IMPORTANT:** The **viadmin** system user is available by default. The system user has the predefined user interface and access permissions to database resources. The user interface and the permissions structure for the system user must not be used or changed in the production system because this system user is overwritten as a template user with each schema update.

**TIP:** Create your own system user with the appropriate permissions. This can be done on initial installation of the One Identity Manager database. This system user can compile an initial One Identity Manager database and can be used to log into the administration tools for the first time.

### Generic single sign-on (role-based)

**NOTE:** This authentication module is available if the Identity Management Base Module is installed.

<table>
<thead>
<tr>
<th>Credentials</th>
<th>The authentication module uses the login data of the user currently logged in on the workstation.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prerequisites</td>
<td></td>
</tr>
</tbody>
</table>
  - The employee exists in the One Identity Manager database.  
  - The employee is assigned at least one application role.  
  - The user account exists in the One Identity Manager database and the employee is entered in the user account’s master data. |
| Set as default | No |
| Single sign-on | Yes |
| Front-end login allowed | Yes |
| Web Portal login allowed | Yes |
Remarks

One Identity Manager searches for the user account according to the configuration and finds the employee assigned to the user account.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

A dynamic system user determined from the employee’s application roles. The user interface and the write permissions are loaded through this system user.

Changes to the data are assigned to the logged in employee.

Modify the following configuration parameters in the Designer to implement the authentication module.

### Table 29: Configuration Parameters for the Authentication Module

<table>
<thead>
<tr>
<th>Configuration parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
</tbody>
</table>
### Employee

**NOTE:** This authentication module is available if the Identity Management Base Module is installed.

**Credentials**
- Employee's central user account and password.

**Prerequisites**
- The system user with permissions exists in the One Identity Manager database.
- The employee exists in the One Identity Manager database.
- The central user account is entered in the employee's master data.
- The system user is entered in the employee's master data.
- The system user password is entered in the employee's master data.

**Set as default**
- Yes

**Single sign-on**
- No

**Front-end login allowed**
- Yes

**Web Portal login allowed**
- Yes

**Remarks**
If an employee has more than one identity, the **QER | Person | MasterIdentity | UseMasterForAuthentication** configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

The user interface and the write permissions are loaded through the system user that is directly assigned to the logged in employee. Changes to the data are assigned to the logged in employee.

### Employee (role-based)

**NOTE:** This authentication module is available if the Identity Management Base Module is installed.
Credentials  Employee's central user account and password.

Prerequisites
- The employee exists in the One Identity Manager database.
- The central user account is entered in the employee's master data.
- The system user password is entered in the employee's master data.
- The employee is assigned at least one application role.

Set as default  Yes
Single sign-on  No
Front-end login allowed  Yes
Web Portal login allowed  Yes

Remarks  If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.
- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

A dynamic system user determined from the employee's application roles. The user interface and the write permissions are loaded through this system user.
Changes to the data are assigned to the logged in employee.

Employee (dynamic)

NOTE: This authentication module is available if the Identity Management Base Module is installed.

Credentials  Employee's central user account and password.

Prerequisites
- The employee exists in the One Identity Manager database.
- The central user account is entered in the employee's master data.
- The system user password is entered in the employee's master data.
- The configuration data for dynamically determining the system user is defined in the application. Thus, an employee can, for example, be...
assigned a system user dynamically depending on their department membership.

Set as default | Yes
Single sign-on | No
Front-end login allowed | Yes
Web Portal login allowed | Yes

Remarks
If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

The application configuration data is used to determine a system user, which is automatically assigned to the employee. The user interface and write permissions are loaded through the system user that is dynamically assigned to the logged in employee.

Changes to the data are assigned to the logged in employee.

Related topics
- Configuration data for system user dynamic authentication on page 101

User account

NOTE: This authentication module is available if the Identity Management Base Module is installed.

Credentials
The authentication module uses the Active Directory login data of the user currently logged in on the workstation.

Prerequisites
- The system user with permissions exists in the One Identity Manager database.
- The employee exists in the One Identity Manager database.
- Permitted logins are entered in the employee's master data. The logins
are expected in the form: domain\user.

- The system user is entered in the employee's master data.

<table>
<thead>
<tr>
<th>Set as default</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single sign-on</td>
<td>Yes</td>
</tr>
<tr>
<td>Front-end login allowed</td>
<td>Yes</td>
</tr>
<tr>
<td>Web Portal login allowed</td>
<td>Yes</td>
</tr>
</tbody>
</table>
| Remarks | All employee logins saved in the One Identity Manager database are found. The employee whose login data matches that of the current user is used for logging in.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

The user interface and access permissions are loaded through the system user that is directly assigned to the employee found. Data modifications are attributed to the current user account.

### User account (role-based)

**NOTE:** This authentication module is available if the Identity Management Base Module is installed.

**Credentials**

- The authentication module uses the Active Directory login data of the user currently logged in on the workstation.

**Prerequisites**

- The employee exists in the One Identity Manager database.
- Permitted logins are entered in the employee's master data. The logins are expected in the form: domain\user.
- The employee is assigned at least one application role.

| Set as default | No |
Single sign-on: Yes
Front-end login allowed: Yes
Web Portal login allowed: Yes
Remarks: All employee logins saved in the One Identity Manager database are found. The employee whose login data matches that of the current user is used for logging in.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

A dynamic system user determined from the employee's application roles. The user interface and the write permissions are loaded through this system user. Data modifications are attributed to the current user account.

**Account-based system user**

**Credentials:** The authentication module uses the Active Directory login data of the user currently logged in on the workstation.

**Prerequisites:**

- The system user with permissions exists in the One Identity Manager database.
- Permitted logins are entered in the system user’s master data. The logins are expected in the form: domain\user.

**Set as default:** No
**Single sign-on:** Yes
**Front-end login allowed:** Yes
**Web Portal login allowed:** No
**Remarks:** All system user logins saved in the One Identity Manager database are
found. The system user whose login data matches that of the current user is used for logging in.
The user interface and the write permissions are loaded through the system user.
Data modifications are attributed to the current user account.

**Active Directory user account**

> **NOTE:** This authentication module is available if the module Active Directory Module is installed.

**Credentials**

The authentication module uses the Active Directory login data of the user currently logged in on the workstation.

**Prerequisites**

- The system user with permissions exists in the One Identity Manager database.
- The employee exists in the One Identity Manager database.
- The system user is entered in the employee’s master data.
- The Active Directory user account exists in the One Identity Manager database and the employee is entered in the user account’s master data.

**Set as default**

Yes

**Single sign-on**

Yes

**Front-end login allowed**

Yes

**Web Portal login allowed**

Yes

**Remarks**

The appropriate user account is found in the One Identity Manager database through the user’s SID and the domain given at login. One Identity Manager determines which employee is assigned to the user account.

If an employee has more than one identity, the **QER | Person | MasterIdentity | UseMasterForAuthentication** configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for
The user interface and access permissions are loaded through the system user that is directly assigned to the employee found. If a system user is not assigned to the employee, the system user from the **SysConfig | Logon | DefaultUser** configuration parameter is used.

Data modifications are attributed to the current user account.

**NOTE:** If the option **Connect automatically** is set, authentication is no longer necessary for subsequent logins.

### Active Directory user account (role-based)

**NOTE:** This authentication module is available if the module Active Directory Module is installed.

**Credentials**

The authentication module uses the Active Directory login data of the user currently logged in on the workstation.

**Prerequisites**

- The employee exists in the One Identity Manager database.
- The employee is assigned at least one application role.
- The Active Directory user account exists in the One Identity Manager database and the employee is entered in the user account’s master data.

**Set as default**

Yes

**Single sign-on**

Yes

**Front-end login allowed**

Yes

**Web Portal login allowed**

Yes

**Remarks**

The appropriate user account is found in the One Identity Manager database through the user’s SID and the domain given at login. One Identity Manager determines which employee is assigned to the user account.

If an employee has more than one identity, the **QER | Person | MasterIdentity | UseMasterForAuthentication** configuration parameter controls which employee identity is used for authentication.
• If this configuration parameter is set, the employee’s main identity is used for authentication.
• If the parameter is not set, the employee’s subidentity is used for authentication.

A dynamic system user determined from the employee’s application roles. The user interface and the write permissions are loaded through this system user.

Data modifications are attributed to the current user account.

**NOTE:** If the option **Connect automatically** is set, authentication is no longer necessary for subsequent logins.

**Active Directory user account (manual input)**

**NOTE:** This authentication module is available if the module Active Directory Module is installed.

**Credentials**
- Login name and password for registering with Active Directory. You do not have to enter the domain.

**Prerequisites**
- The employee exists in the One Identity Manager database.
- The Active Directory user account exists in the One Identity Manager database and the employee is entered in the user account’s master data.
- The permitted domains for login are entered in the **TargetSystem | ADS | AuthenticationDomains** configuration parameter.
- The configuration data for dynamically determining the system user is defined in the application. Thus, an employee can, for example, be assigned a system user dynamically depending on their department membership.

| Set as default | No |
| Single sign-on | No |
| Front-end login allowed | Yes |
| Web Portal login allowed | Yes |
Remarks

The user’s identity is determined from a predefined list of permitted Active Directory domains. The corresponding user account and employee are determined in the One Identity Manager database, which the user account is assigned to.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

The application configuration data is used to determine a system user, which is automatically assigned to the employee. The user interface and write permissions are loaded through the system user that is dynamically assigned to the logged in employee.

Data modifications are attributed to the current user account.

Active Directory user account (manual input/role-based)

**NOTE:** This authentication module is available if the module Active Directory Module is installed.

**Credentials**
Login name and password for registering with Active Directory. You do not have to enter the domain.

**Prerequisites**
- The employee exists in the One Identity Manager database.
- The employee is assigned at least one application role.
- The Active Directory user account exists in the One Identity Manager database and the employee is entered in the user account's master data.
- The permitted domains for login are entered in the TargetSystem | ADS | AuthenticationDomains configuration parameter.

| Set as default | Yes |
| Single sign-on | No |
| Front-end login allowed | Yes |
Web Portal login allowed | Yes

Remarks | The user’s identity is determined from a predefined list of permitted Active Directory domains. The corresponding user account and employee are determined in the One Identity Manager database, which the user account is assigned to.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

A dynamic system user determined from the employee's application roles. The user interface and the write permissions are loaded through this system user.

Data modifications are attributed to the current user account.

<table>
<thead>
<tr>
<th>Active Directory user account (dynamic)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>NOTE:</strong> This authentication module is available if the module Active Directory Module is installed.</td>
</tr>
</tbody>
</table>

Credentials | The authentication module uses the Active Directory login data of the user currently logged in on the workstation.

Prerequisites | The employee exists in the One Identity Manager database.
- The Active Directory user account exists in the One Identity Manager database and the employee is entered in the user account's master data.
- The configuration data for dynamically determining the system user is defined in the application. Thus, an employee can, for example, be assigned a system user dynamically depending on their department membership.

| Set as default | No |
| Single sign-on | Yes |
| Front-end login allowed | Yes |
Web Portal login allowed: Yes

Remarks: The appropriate user account is found in the One Identity Manager database through the user’s SID and the domain given at login. One Identity Manager determines which employee is assigned to the user account.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

The application configuration data is used to determine a system user, which is automatically assigned to the employee. The user interface and write permissions are loaded through the system user that is dynamically assigned to the logged in employee.

Data modifications are attributed to the current user account.

**NOTE:** If the option Connect automatically is set, authentication is no longer necessary for subsequent logins.

Related topics

- Configuration data for system user dynamic authentication on page 101

**LDAP user account (role-based)**

**NOTE:** This authentication module is available if the module LDAP Module is installed.

Credentials

- Login name, identifier, distinguished name or user ID of an LDAP user account.
- LDAP user account's password.

Prerequisites

- The employee exists in the One Identity Manager database.
- The employee is assigned at least one application role.
- The LDAP user account exists in the One Identity Manager database and the employee is entered in the user account's master data.
- The configuration data for dynamically determining the system user is defined in the application. Thus, an employee can, for example, be...
assigned a system user dynamically depending on their department membership.

Set as default  No
Single sign-on  No
Front-end login allowed  Yes
Web Portal login allowed  Yes
Remarks  If you log in using a login name, identifier or user ID, the corresponding user account is determined in the One Identity Manager database through the container’s domain. Logging in with a distinguished name is done directly. One Identity Manager determines which employee is assigned to the LDAP user account.

If an employee has more than one identity, the **QER | Person | MasterIdentity | UseMasterForAuthentication** configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

A dynamic system user determined from the employee’s application roles. The user interface and the write permissions are loaded through this system user.

Data modifications are attributed to the current user account.

Modify the following configuration parameters in the Designer to implement the authentication module.

**Table 30: Configuration Parameters for the Authentication Module**

<table>
<thead>
<tr>
<th>Configuration parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>TargetSystem</td>
<td>The configuration parameter allows configuration of the LDAP authentication module.</td>
</tr>
<tr>
<td>LDAP</td>
<td>Authentication</td>
</tr>
<tr>
<td>Authentic-ation</td>
<td>The configuration parameter specified the authentication mechanism. Permitted values are <strong>Secure, Encryption, SecureSocketsLayer, ReadOnlyServer, Anonymous, FastBind, Signing, Sealing, Delegation</strong> and <strong>ServerBind</strong>. The value can be combined with commas (,). For more information about authentication types, see the <a href="https://docs.microsoft.com">MSDN Library</a>.</td>
</tr>
</tbody>
</table>
### Configuration parameter | Meaning
---|---
The default is **ServerBind**.

<table>
<thead>
<tr>
<th>TargetSystem</th>
<th>LDAP</th>
<th>Authentication</th>
<th>Port</th>
</tr>
</thead>
</table>
| | LDAP server's port. The default is port **389**.

<table>
<thead>
<tr>
<th>TargetSystem</th>
<th>LDAP</th>
<th>Authentication</th>
<th>RootDN</th>
</tr>
</thead>
</table>
| | The configuration parameter contains the root domain's distinguished name. Syntax: dc=MyDomain

<table>
<thead>
<tr>
<th>TargetSystem</th>
<th>LDAP</th>
<th>Authentication</th>
<th>Server</th>
</tr>
</thead>
</table>
| | The configuration parameter contains the name of the LDAP server.

### LDAP user account (dynamic)

**NOTE:** This authentication module is available if the LDAP Module is installed.

- **Credentials**
  - Login name, identifier, distinguished name or user ID of an LDAP user account.
  - LDAP user account's password.

- **Prerequisites**
  - The employee exists in the One Identity Manager database.
  - The LDAP user account exists in the One Identity Manager database and the employee is entered in the user account's master data.
  - The configuration data for dynamically determining the system user is defined in the application. Thus, an employee can, for example, be assigned a system user dynamically depending on their department membership.

- **Set as default**
  - No

- **Single sign-on**
  - No

- **Front-end login allowed**
  - Yes

- **Web Portal**
  - Yes
If you log in using a login name, identifier or user ID, the corresponding user account is determined in the One Identity Manager database through the container's domain. Logging in with a distinguished name is done directly. One Identity Manager determines which employee is assigned to the LDAP user account.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.

The application configuration data is used to determine a system user, which is automatically assigned to the employee. The user interface and write permissions are loaded through the system user that is dynamically assigned to the logged in employee.

Data modifications are attributed to the current user account.

Modify the following configuration parameters in the Designer to implement the authentication module.

### Table 31: Configuration Parameters for the Authentication Module

<table>
<thead>
<tr>
<th>Configuration parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>TargetSystem</td>
<td>LDAP</td>
</tr>
<tr>
<td>TargetSystem</td>
<td>LDAP</td>
</tr>
<tr>
<td>TargetSystem</td>
<td>LDAP</td>
</tr>
</tbody>
</table>

The default is **ServerBind**. For more information about authentication types, see the MSDN Library.
**Configuration parameter**  
**Meaning**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>dc=MyDomain</td>
<td>The configuration parameter contains the name of the LDAP server.</td>
</tr>
<tr>
<td>TargetSystem</td>
<td>The configuration parameter contains the name of the LDAP server.</td>
</tr>
<tr>
<td></td>
<td>LDAP</td>
</tr>
</tbody>
</table>

**Related topics**

- Configuration data for system user dynamic authentication on page 101

**HTTP Header**

The authentication module supports authentication via web single sign-on solutions that work with a proxy--based architecture.

**Credentials**

Employee's central user account or personnel number.

**Prerequisites**

- The system user with permissions exists in the One Identity Manager database.
- The employee exists in the One Identity Manager database.
- The central user account or personnel number is entered in the employee's master data.
- The system user is entered in the employee's master data.

**Set as default**

No

**Single sign-on**

Yes

**Front-end login allowed**

No

**Web Portal login allowed**

Yes

**Remarks**

You must pass the user (in the form: UserName =<user name of authenticated user>) in the HTTP header. The employee is found in the One Identity Manager database whose central user account or personnel number matches the user name passed down.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.
The user interface and the write permissions are loaded through the system user that is directly assigned to the logged in employee. If a system user is not assigned to the employee, the system user from the SysConfig | Logon | DefaultUser configuration parameter is used. Changes to the data are assigned to the logged in employee.

### HTTP Header (role-based)

The authentication module supports authentication via web single sign-on solutions that work with a proxy-based architecture.

- **Credentials**  Employee's central user account or personnel number.
- **Prerequisites**
  - The employee exists in the One Identity Manager database.
  - The central user account or personnel number is entered in the employee's master data.
  - The employee is assigned at least one application role.
- **Set as default**  Yes
- **Single sign-on**  Yes
- **Front-end login allowed**  No
- **Web Portal login allowed**  Yes
- **Remarks**
  You must pass the user (in the form: `UserName=<user name of authenticated user>`) in the HTTP header. The employee is found in the One Identity Manager database whose central user account or personnel number matches the user name passed down.

If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.

- If this configuration parameter is set, the employee’s main identity is used for authentication.
- If the parameter is not set, the employee’s subidentity is used for authentication.
OAuth 2.0 / OpenID Connect

NOTE: This authentication module is available if the Identity Management Base Module is installed.

The authorization module supports the authorization code for OAuth 2.0 and OpenID Connect. For more detailed information about the authorization code flow, see, for example, the OAuth Specification or the OpenID Connect Specification.

This authentication module uses a Secure Token Service for logging in. This login procedure can be used with every Secure Token Service that can return an OAuth 2.0 token.

Credentials

Prerequisites
- The system user with permissions exists in the One Identity Manager database.
- The employee exists in the One Identity Manager database.
- The system user is entered in the employee's master data.
- The user account exists in the One Identity Manager database and the employee is entered in the user account's master data.

Set as default
No

Single sign-on
No

Front-end login allowed
Yes

Web Portal login allowed
Yes

Remarks
One Identity Manager determines which employee is assigned to the user account.
If an employee has more than one identity, the QER | Person | MasterIdentity | UseMasterForAuthentication configuration parameter controls which employee identity is used for authentication.
If this configuration parameter is set, the employee’s main identity is used for authentication.

If the parameter is not set, the employee’s subidentity is used for authentication.

The user interface and access permissions are loaded through the system user that is directly assigned to the employee found.

Data modifications are attributed to the current user account. To do this, the claim type whose value is used for labeling data changes must be declared.

Related topics

- OAuth 2.0/OpenID Connect configuration on page 106
- Expiry of the OAuth 2.0/OpenID Connect authentication on page 106

## OAuth 2.0 / OpenID Connect (role-based)

[NOTE: This authentication module is available if the Identity Management Base Module is installed.]

The authorization module supports the authorization code for OAuth 2.0 and OpenID Connect. For more detailed information about the authorization code flow, see, for example, the OAuth Specification or the OpenID Connect Specification.

This authentication module uses a Secure Token Service for logging in. This login procedure can be used with every Secure Token Service that can return an OAuth 2.0 token.

**Credentials**

- Dependent on the authentication method of the secure token service.

**Prerequisites**

- The employee exists in the One Identity Manager database.
- The employee is assigned at least one application role.
- The user account exists in the One Identity Manager database and the employee is entered in the user account’s master data.

| Set as default | No |
| Single sign-on | No |
| Front-end login allowed | Yes |
Data modifications are attributed to the current user account. To do this, the claim type whose value is used for labeling data changes must be declared.

Related topics

- OAuth 2.0/OpenID Connect configuration on page 106
- Expiry of the OAuth 2.0/OpenID Connect authentication on page 106

Synchronization authentication module

**NOTE:** This authentication module is available if the module Target System Synchronization Module is installed.

This authentication module integrates the default method for Synchronization Editor login.

**Credentials**

Login uses the **sa** system user.

**Prerequisites**

- **Set as default**: Yes
- **Single sign-on**: No
- **Front-end login allowed**: No
- **Web Portal login allowed**: No

**Remarks**

You should not change the system user **sa**. The system user is overwritten with each schema update.
Web agent authentication module

The authentication module integrates the default method for Web Designer login, to access the database before the first user login.

Credentials
Login uses the sa system user.

Prerequisites
Set as default Yes
Single sign-on No
Front-end login allowed No
Web Portal login allowed No

Remarks You should not change the system user sa. The system user is overwritten with each schema update.

Component authentication module

This authentication module integrates the default method for registering process components.

Credentials
Login uses the sa system user.

Prerequisites
Set as default Yes
Single sign-on No
Front-end login allowed No
Web Portal login allowed No

Remarks You should not change the system user sa. The system user is overwritten with each schema update.
Crawler

The authentication module is used by the application server to compile search indexes for full text search over the database.

Credentials
Login uses the **sa** system user.

Prerequisites

<table>
<thead>
<tr>
<th>Set as default</th>
<th>Yes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single sign-on</td>
<td>No</td>
</tr>
<tr>
<td>Front-end login allowed</td>
<td>No</td>
</tr>
<tr>
<td>Web Portal login allowed</td>
<td>No</td>
</tr>
</tbody>
</table>

Remarks
You should not change the system user **sa**. The system user is overwritten with each schema update.

Password reset

**NOTE:** This authentication module is available if the Identity Management Base Module is installed.

The authentication module is used for login to Password Reset Portal. The authentication module checks the access code or the employee’s answers to the password questions. In the case of login with an access code, this information is deleted after a successful login.

Credentials
Central user account and access code.
- OR -
Central user account and answers to the password questions.

Prerequisites
- The employee exists in the One Identity Manager database.
- The central user account is entered in the employee’s master data.
- The employee is not deactivated or has the certification status **New**.
- The employee has an access code or the questions and answers for the password prompt have been specified.

<table>
<thead>
<tr>
<th>Set as default</th>
<th>No</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single sign-on</td>
<td>No</td>
</tr>
</tbody>
</table>
The application token for Password Reset Portal must be specified. You set the application token when installing Password Reset Portal. The application token is saved as a hash value in the database in the QER | Person | PasswordResetAuthenticator | ApplicationToken parameter and stored encrypted in the web.config file. For detailed information about configuring Password Reset Portal, see the One Identity Manager Web Application Configuration Guide.

Password reset (role-based)

NOTE: This authentication module is available if the Identity Management Base Module is installed.

The authentication module is used for login to Password Reset Portal. The authentication module checks the access code or the employee’s answers to the password questions. In the case of login with an access code, this information is deleted after a successful login.

Credentials
- Central user account and access code.
  - OR -
  Central user account and answers to the password questions.

Prerequisites
- The employee exists in the One Identity Manager database.
- The central user account is entered in the employee's master data.
- The employee is not deactivated or has the certification status New.
- The employee has an access code or the questions and answers for the password prompt have been specified.
- The employee is assigned at least one application role.

Set as default
- Yes

Single sign-on
- No

Front-end login allowed
- No

Web Portal login allowed
- No
The application token for Password Reset Portal must be specified. You set the application token when installing Password Reset Portal. The application token is saved as a hash value in the database in the QER | Person | PasswordResetAuthenticator | ApplicationToken parameter and stored encrypted in the web.config file. For detailed information about configuring Password Reset Portal, see the One Identity Manager Web Application Configuration Guide.

A dynamic system user determined from the employee's application roles. The user interface and the write permissions are loaded through this system user.

Editing authentication modules

Before you can use an authentication module for logging on, the following prerequisites must be fulfilled:

1. The authentication module must be enabled.
2. The authentication module must be assigned to the application.
3. The assignment of the authentication module to the application must be enabled.

This allows you to log in to the assigned application using this authentication module. Ensure that users found through the authentication module also have the required program function to use the program.

Detailed information about this topic

- Enabling authentication modules on page 95
- Assigning authentication modules to applications on page 95
- Disabling or enabling authentication modules for applications on page 95
- Authentication module properties on page 96
- Initial data for authentication modules on page 97
- Configuration data for system user dynamic authentication on page 101
- One Identity Manager Authentication modules on page 69
- Managing permissions to program features on page 62
Enabling authentication modules

To enable an authentication module
1. In Designer, select Base data | Security settings | Authentication modules.
2. In List Editor, select the authentication module.
3. In the Properties view, set the Activated property to True.
4. Save the changes.

Related topics
- Disabling or enabling authentication modules for applications on page 95

Assigning authentication modules to applications

If create custom authentication modules, assign them to the existing programs. In general, you do not need to change assignments of predefined authentication modules.

To assign an authentication module to applications
1. In Designer, select Base data | Security settings | Authentication modules.
2. Select View | Select table relations and enable the DialogProductHasAuthentifier table.
3. In List Editor, select the authentication module.
4. Assign the application in the Application edit view.
5. Save the changes.

Related topics
- Disabling or enabling authentication modules for applications on page 95

Disabling or enabling authentication modules for applications

To disable an authentication module for an application
1. Select Base data | Security settings | Programs in Designer.
2. In List Editor, select the application and click on Usage overview.
3. In the **Effective authenticators** form element, select the authentication module.
4. Start the Object Editor using **Edit object**.
5. In the **Disabled** property, set the value to **True**.
6. Save the changes.

**To enable an authentication module for an application**

1. Select **Base data | Security settings | Programs** in Designer.
2. In List Editor, select the application and click on **Usage overview**.
3. In the **Disabled authenticators** form element, select the authentication module.
4. Start the Object Editor using **Edit object**.
5. In the **Disabled** property, set the value to **False**.
6. Save the changes.

**Related topics**

- Assigning authentication modules to applications on page 95
- Enabling authentication modules on page 95

### Authentication module properties

**Table 32: Authentication Module Properties**

<table>
<thead>
<tr>
<th>Property</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enabled</td>
<td>Specifies whether the authentication module can be used.</td>
</tr>
<tr>
<td>Display name</td>
<td>This name is used to identify the authentication module in the admin-</td>
</tr>
<tr>
<td></td>
<td>istration tool’s login window.</td>
</tr>
<tr>
<td>Authentication module</td>
<td>Internal name of the authentication module.</td>
</tr>
<tr>
<td>Authentication type</td>
<td>Specifies the type of authentication module. You can choose <strong>Dynamic</strong></td>
</tr>
<tr>
<td></td>
<td>and <strong>Role based</strong>.</td>
</tr>
<tr>
<td>Processing status</td>
<td>The processing status is used for creating custom configuration</td>
</tr>
<tr>
<td></td>
<td>packages.</td>
</tr>
<tr>
<td>Initial data</td>
<td>Initial data for logging in with this authentication module.</td>
</tr>
<tr>
<td>Class</td>
<td>Authentication module class.</td>
</tr>
<tr>
<td>Assembly name</td>
<td>Name of the assembly file.</td>
</tr>
<tr>
<td>Sort order</td>
<td>Specify the order in which the modules are displayed in the login</td>
</tr>
<tr>
<td></td>
<td>window.</td>
</tr>
</tbody>
</table>
### Property Meaning

<table>
<thead>
<tr>
<th>Property</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single sign-on</td>
<td>Specifies whether the authentication module may be authenticated without a password.</td>
</tr>
<tr>
<td>Select in front-end</td>
<td>Specifies whether the authentication module can be selected in the login window.</td>
</tr>
</tbody>
</table>

## Initial data for authentication modules

The initial data is one part of the authentication string (parameter-value pair without module ID). Initial data from the authentication string is preallocated by default for each authentication instance.

The authentication string is formatted as follows:

```
Module=<name>;<property1>=<value1>;<property2>=<value2>,...
```

Example:

```
Module=DialogUser;User=<username>;Password=<password>
```

### To specify initial data

1. In Designer, select **Base data | Security settings | Authentication modules**.
2. Select the authentication module and enter the data in **Initial data**.
   - Syntax:
     ```
     property1=value1;property2=value2
     ```
   - Example:
     ```
     User=<username>;Password=<password>
     ```

You can use different initial data depending on the authentication module.

### Table 33: Initial data for authentication modules

<table>
<thead>
<tr>
<th>Module Display Name</th>
<th>Authentication module</th>
<th>Parameter</th>
<th>Meaning/Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>System user</td>
<td>DialogUser</td>
<td>User</td>
<td>User name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Password</td>
<td>User password.</td>
</tr>
<tr>
<td>Active Directory</td>
<td>ADSAccount</td>
<td>Product</td>
<td>Use case. The system user is determined through the use</td>
</tr>
<tr>
<td>Module Display Name</td>
<td>Authentication module</td>
<td>Parameter</td>
<td>Meaning/Comment</td>
</tr>
<tr>
<td>---------------------</td>
<td>-----------------------------</td>
<td>-----------</td>
<td>---------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>user account (dynamic)</td>
<td>DynamicManualADS</td>
<td>Product</td>
<td>Use case. The system user is determined through the use case configuration data.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>User</td>
<td>User name. The user’s identity is determined from a predefined list of permitted Active Directory domains. Enter the permitted Active Directory domains in the TargetSystem</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Password</td>
<td>User password.</td>
</tr>
<tr>
<td>Active Directory user account (manual input)</td>
<td>RoleBasedADSAccount</td>
<td>No parameters required</td>
<td></td>
</tr>
<tr>
<td></td>
<td>RoleBasedManualADS</td>
<td>User</td>
<td>User name. The user’s identity is determined from a predefined list of permitted Active Directory domains. Enter the permitted Active Directory domains in the TargetSystem</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Password</td>
<td>User password.</td>
</tr>
<tr>
<td>Employee</td>
<td>Employee</td>
<td>User</td>
<td>Employee’s central user account.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Password</td>
<td>User password.</td>
</tr>
<tr>
<td>Employee (dynamic)</td>
<td>DynamicPerson</td>
<td>Product</td>
<td>Use case. The system user is determined through the use case configuration data.</td>
</tr>
</tbody>
</table>

One Identity Manager 8.1.1 Authorization and Authentication Guide
One Identity Manager Authentication modules
<table>
<thead>
<tr>
<th>Module Display Name</th>
<th>Authentication module</th>
<th>Parameter</th>
<th>Meaning/Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>User</td>
<td>User name.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Employee (role-based)</td>
<td>RoleBasedPerson</td>
<td>User</td>
<td>User name.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Password</td>
<td>User password.</td>
</tr>
<tr>
<td>HTTP header</td>
<td>HTTPHeader</td>
<td>Header</td>
<td>HTTP Header to use.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KeyColumn</td>
<td>Comma delimited list of key columns in the table Person to be searched for user names. Default: CentralAccount, PersonnelNumber</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTTP header (role-based)</td>
<td>RoleBasedHTTPHeader</td>
<td>Header</td>
<td>HTTP Header to use.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>KeyColumn</td>
<td>Comma delimited list of key columns in the table Person to be searched for user names. Default: CentralAccount, PersonnelNumber</td>
</tr>
<tr>
<td>LDAP user account (dynamic)</td>
<td>DynamicLdap</td>
<td>User</td>
<td>User name. Default: CN, DistinguishedName, UserID, UIDLDAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Password</td>
<td>User password.</td>
</tr>
<tr>
<td>LDAP user account (role-based)</td>
<td>RoleBasedLdap</td>
<td>User</td>
<td>User name. Default: CN, DistinguishedName, UserID, UIDLDAP</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Password</td>
<td>User password.</td>
</tr>
<tr>
<td>Generic single sign-on (role-based)</td>
<td>RoleBasedGeneric</td>
<td>SearchTable</td>
<td>Table in which to search for the user name of the logged in user. This table must contain a FK named UID_Person which points to the table Person.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>SearchColumn</td>
<td>Column from the SearchTable in which to search for the user</td>
</tr>
<tr>
<td>Module Display Name</td>
<td>Authentication module</td>
<td>Parameter</td>
<td>Meaning/Comment</td>
</tr>
<tr>
<td>---------------------</td>
<td>--------------------------------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>name of the logged-in user.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>DisabledBy</td>
<td>Pipe (</td>
</tr>
<tr>
<td></td>
<td></td>
<td>EnabledBy</td>
<td>Pipe (</td>
</tr>
<tr>
<td>OAuth 2.0/OpenID Connect</td>
<td>OAuth</td>
<td></td>
<td>Dependent on the authentication method of the secure token service.</td>
</tr>
<tr>
<td>OAuth 2.0/OpenID Connect (role-based)</td>
<td>OAuthRoleBased</td>
<td></td>
<td>Dependent on the authentication method of the secure token service.</td>
</tr>
<tr>
<td>Account based system user</td>
<td>DialogUserAccountBased</td>
<td></td>
<td>No parameters required</td>
</tr>
<tr>
<td>User account</td>
<td>QERAccount</td>
<td></td>
<td>No parameters required</td>
</tr>
<tr>
<td>User account (role-based)</td>
<td>RoleBasedQERAccount</td>
<td></td>
<td>No parameters required</td>
</tr>
<tr>
<td>Password reset</td>
<td>PasswordReset</td>
<td></td>
<td>No parameters required</td>
</tr>
<tr>
<td>Password reset (role-based)</td>
<td>RoleBasedPasswordReset</td>
<td></td>
<td>No parameters required</td>
</tr>
</tbody>
</table>

**Related topics**

- Configuration data for system user dynamic authentication on page 101
- One Identity Manager Authentication modules on page 69
Configuration data for system user dynamic authentication

In the case of dynamic authentication modules, the system user assigned to the employee is not used for the log in. The system user which is configured using the user interface special configuration data is taken instead.

To specify configuration data

1. Select Base data | Security settings | Programs in Designer.
2. Select the application and adjust the Configuration data.

Use XML syntax for entering the configuration data:

```xml
<DialogUserDetect>
  <Usermappings>
    <Usermapping>
      DialogUser = "System user name"
      Selection = "Selection criterion"
    </Usermapping>
    <Usermapping>
      DialogUser = "System user name"
    </Usermapping>
    ...
  </Usermappings>
</DialogUserDetect>
```

Enter the system user (DialogUser) in Usermappings section. Specify which employee the given system user should use with the selection criterion (Selection). You are not obliged to enter a selection criterion for the assignment. The first system user that has the required assignment is used for the log in.

You can assign function groups to permissions groups on order to deal with complex rights and user interface structures. The function groups allow you to map the functions an employee has in the company, for example, IT controller or branch manager. Assign the function groups to the permissions groups. A function group can refer to several permissions groups and several function groups can refer to one permissions group.

If the section FunctionGroupMapping is in the configuration data, this is evaluated first and the system user that is found is used. The authentication module uses the system user that is the exact member of the permissions group found for the login. If none is found the section Usermapping is evaluated.

```xml
<DialogUserDetect>
  <FunctionGroupMapping>
    PersonToFunction = "View mapping employee to function group"
  </FunctionGroupMapping>
</DialogUserDetect>
```
FunctionToGroup = "View mapping function group to permissions group"

<Usermappings>
  <Usermapping
    DialogUser = "System user name"
    Selection = "Selection criterion"
  />
  ...
</Usermappings>
</DialogUserDetect>

Related topics
- Example of a simple system user assignment on page 102
- Example of a system user assignment using a selection criterion on page 103
- Example of a function group assignment on page 104
- Granting One Identity Manager schema permissions on page 34

Example of a simple system user assignment

All employees should be able to see the user interface for an IT Shop in a web front-end, without taking table and column permissions into account.

To do this, set up a new application, for example WebShop_Customer, and adapt the configuration data as follows:

<Usermappings>
  <Usermapping
    DialogUser = "dlg_all"
  />
</Usermappings>

Create a new WebShop_Customer permissions group, which receives the user interface for the application comprising the menu items, interface forms and task definitions. The user interface could consist of the following menu items:
- Employee contact data
- Requesting a product
- Unsubscribing a product
Define a new `dlg_all` system user and include it in the `vi_DE-CentralPwd`, `vi_DE-ITShopOrder` and `WebShop_Customer` permissions groups.

Related topics

- Configuration data for system user dynamic authentication on page 101
- Example of a system user assignment using a selection criterion on page 103
- Example of a function group assignment on page 104
- Granting One Identity Manager schema permissions on page 34

Example of a system user assignment using a selection criterion

The scenario described in the previous example is extended such that only the cost center manager can see an employee’s leaving date. You need to add the input field `LeavingDate` to the contact data form to do this.

Permissions are used for controlling viewing and editing. Set up a new `dlg_kst` system user and include the system user in the `vi_DE-CentralPwd`, `vi_DE-ITShopOrder` and `WebShop_Customer` permissions groups. You should also give the system user read and write access to the column `Person.Exitdate`.

Extend the application configuration data in such a way that the cost center managers use the `dlg_kst` system user to log in. All other employees use the `dlg_all` system user to log in.

Change the configuration data as follows:

```xml
<DialogUserDetect>
  <Usermappings>
    <Usermapping>
      DialogUser = "dlg_kst"
      Selection = "select 1 where %uid% in (select uid_personhead from profitcenter)"
    </Usermapping>
    <Usermapping>
      DialogUser = "dlg_all"
    </Usermapping>
  </Usermappings>
</DialogUserDetect>
```
Example of a function group assignment

To assign function groups to permissions groups you have to define two database views. The first database view shows the assignment of employees to function groups. The database view contains two columns UID_Person and FunctionGroup.

Example:
```sql
create view custom_Person2Fu as
    select uid_personHead as UID_Person, 'Cost center manager' as FunctionGroup
    from Profitcenter
    where isnull(uid_personHead, '') > ''
    union all
    select uid_personHead, 'Department manager' as FunctionGroup
    from Department
    where isnull(uid_personHead, '') > ''
```

The second database view assigns function groups to permissions groups. This database view contains two columns FunctionGroup and DialogGroup.

Example:
```sql
create view custom_Fu2D as
    select 'Cost center manager' as FunctionGroup, '<UID_Custom_Dialoggroup_ChefP>' as DialogGroup
    union all select 'Department manager', '<UID_Custom_Dialoggroup_ChefD>' as DialogGroup
```

Set up role-based permissions groups with the required permissions.

**TIP:** A role-based permissions group can inherit from a non role-based permissions group. This allows you to build up an inheritance hierarchy to making it easier to grant permissions.

Change the configuration data for assigning function groups to permissions groups as follows:
```xml
<DialogUserDetect>
    <FunctionGroupMapping
        PersonToFunction = "custom_Person2Fu"
```
FunctionToGroup = "custom_Fu2D"

</DialogUserDetect>

Related topics

- Configuration data for system user dynamic authentication on page 101
- Example of a simple system user assignment on page 102
- Example of a system user assignment using a selection criterion on page 103
- Granting One Identity Manager schema permissions on page 34

Enabling the validity of a login

The system runs additional validity checks to prevent users from working with established connections, if they were deactivated after they logged in. The check takes place when the next permissions based action on the connection at a fixed interval of 20 minutes.

You can adjust the interval in the configuration parameter Common | Authentication | CheckInterval. Edit the configuration parameter in the Designer.
OAuth 2.0/OpenID Connect configuration

The OAuth2.0/OpenID Connect and OAuth2.0/OpenID Connect (role-based) authentication modules support the authorization code flow for OAuth 2.0 and OpenID Connect. For more detailed information about the authorization code flow, see, for example, the OAuth Specification or the OpenID Connect Specification.

To use the OAuth2.0/OpenID Connect authentication:

- In the Designer, create the identity provider and the OAuth2.0/OpenID Connect applications for the identity provider. A wizard is available in the Designer to assist in this process.
- Assign the OAuth2.0/OpenID Connect application to the web applications.

Related topics

- Expiry of the OAuth 2.0/OpenID Connect authentication on page 106
- Creating the OAuth 2.0/OpenID Connect configuration on page 108
- Assigning OAuth 2.0/OpenID Connect configuration to web applications on page 112
- Defining enabling and disabling columns for the determination of user accounts on page 114
- OAuth 2.0 / OpenID Connect on page 88
- OAuth 2.0 / OpenID Connect (role-based) on page 89

Expiry of the OAuth 2.0/OpenID Connect authentication

At the endpoint of the authorization, the web application (or native application) requests the authorization code. The login endpoint is used to call an advanced login window, which serves to determine the authorization code. The authentication module requires an access token from the token endpoint and the certificate is required to check the security token.
In the process, an attempt is made to find the certificate from the web application configuration. If this is not possible, the settings of the identity provider are used. To find the certificate for testing the token, the certificate stores are queries in the following order:

1. Configuration of the OAuth 2.0/OpenID Connect application (table QBMIdentityClient)
   b. Subject or fingerprint from the local memory (QBMIdentityClient.CertificateSubject and QBMIdentityClient.CertificateThumbPrint).

   In addition, the subject or fingerprint is used to check certificates from the server if they are specified and do not exist locally on the server.

2. Configuration of the identity provider (table QBMIdentityProvider)
   b. Subject or fingerprint from the local memory (QBMIdentityProvider.CertificateSubject and QBMIdentityProvider.CertificateThumbPrint).

   In addition, the subject or fingerprint is used to check certificates from the server if they are specified and do not exist locally on the server.
   d. JSON-Web-Key endpoint (QBMIdentityProvider.JsonWebKeyEndpoint).

To identify the user account, the system determines which claim type is used to find the user information and which information from the One Identity Manager schema is used to find the user account.

Authentication through OpenID is built on OAuth 2.0. The OpenID Connect authentication uses the same mechanisms, but makes the claims available either in an ID token or via a UserInfo endpoint. Other configuration settings are required for using OpenID Connect. If the Scope contains the openid value, the authentication module uses OpenID Connect for authentication.

Related topics

- Creating the OAuth 2.0/OpenID Connect configuration on page 108
- OAuth 2.0 / OpenID Connect on page 88
- OAuth 2.0 / OpenID Connect (role-based) on page 89
Creating the OAuth 2.0/OpenID Connect configuration

To create an OAuth 2.0/OpenID Connect configuration

1. In the Designer, select Base data | Security settings | OAuth 2.0/OpenID Connect configuration.
2. Select Create a new identity provider.
3. On the start page of the wizard, click Next.
4. On the New identity provider page, enter the display name for the configuration and a description.
5. Click Next.
6. On the Automatic configuration discovery page, you define how you want to enter the information about the identity provider.
   - If the configuration data can be determined automatically via OpenID Connect Discovery:
     a. Select Automatic configuration data discovery.
     b. Enter the address (URL) for automatic determination of the configuration data in the input field, or select an example address via the selection menu.
     c. Click on Execute.
     d. The configuration data is determined and a dialog window is displayed. To accept the configuration data, click OK.
   - If you do not want to determined the configuration data automatically, select Manual data input.
     Enter the configuration data on the next page of the wizard.
7. Click Next.
8. On the Configuration data page, enter the general information for the database user.
   
   **NOTE:** If you selected automatic determination of configuration data, some of the information is already completed.

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login endpoint</td>
<td>Uniform Resource Locator (URL) of the Secure Token Service login page.</td>
</tr>
<tr>
<td></td>
<td>Example: <a href="http://localhost/rsts/login">http://localhost/rsts/login</a></td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Logout endpoint</td>
<td>URL of the log-out endpoint</td>
</tr>
<tr>
<td></td>
<td>Example: <a href="http://localhost/rsts/login?wa=wsignin1.0">http://localhost/rsts/login?wa=wsignin1.0</a></td>
</tr>
<tr>
<td>Token endpoint</td>
<td>Uniform Resource Identifier (URL) of the token endpoint of the authorization server for returning the access token to the client for logging in.</td>
</tr>
<tr>
<td></td>
<td>Example: <a href="https://localhost/rsts/oauth2/token">https://localhost/rsts/oauth2/token</a></td>
</tr>
<tr>
<td>UserInfo endpoint</td>
<td>URL of the OpenID Connect UserInfo endpoint.</td>
</tr>
<tr>
<td>Self-signed certificates allowed</td>
<td>Specifies whether self-signed certificates are allowed for connecting to the token endpoint and UserInfo endpoint.</td>
</tr>
<tr>
<td>Issuer</td>
<td>Uniform Resource Identifier (URI) of the certificate issuer for verifying the security token.</td>
</tr>
<tr>
<td></td>
<td>Example: urn:STS/identity</td>
</tr>
<tr>
<td>Scope</td>
<td>Protocol for authentication. If the value is openid, OpenID Connect is used for authentication, otherwise OAuth 2.0 is used.</td>
</tr>
<tr>
<td>Shared Secret</td>
<td>Shared-Secret value used for authentication at the token endpoint. If all applications of the identity provider use the same Shared Secret, enter the value here. If the applications use different Shared Secrets, enter the Shared Secret values when creating the applications.</td>
</tr>
</tbody>
</table>

9. Click Next.

10. On the Configure certificates page, enter the information for the identity provider's certificate. If all applications use the same certificate, enter the information here. NOTE: If you selected automatic determination of configuration data, some of the information is already completed.

    **NOTE:** If you selected automatic determination of configuration data, some of the information is already completed.

**Table 35: Information about the identity provider certificate**

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate endpoint</td>
<td>Uniform Resource Locator (URL) of the certificate end point on the authorization server.</td>
</tr>
<tr>
<td></td>
<td>Example: <a href="https://localhost/RSTS/SigningCertificate">https://localhost/RSTS/SigningCertificate</a></td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Subject of the certificate</td>
<td>Subject of the certificate used for verification. The subject or fingerprint must be set.</td>
</tr>
<tr>
<td>Fingerprint</td>
<td>Fingerprint of the certificate used to verify the security token.</td>
</tr>
<tr>
<td>JSON-Web-Key endpoint</td>
<td>URL of the JSON web key endpoint providing the token signing keys.</td>
</tr>
<tr>
<td>Certificate</td>
<td>Character string of the certificate content. It is used if no certificate is configured.</td>
</tr>
</tbody>
</table>

11. Click **Next**.

12. On the **Search rule for user information** page, you define how the login information is determined between the identity provider and the One Identity Manager database.

### Table 36: Determining the login information

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
</table>
| Value for the search | Full name of the claim type from which the login information is determined on the identity provider.  
Example: name of an entity  
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier  
If you have determined the configuration data automatically, select a value from the list. |
| Column to search  | Table and column in the One Identity Manager database in which the user information is stored. The table must contain a foreign key with the name UID_Person, which points to the Person table.  
Example: ADSAccount.ObjectGUID |
| User name value   | Full name of the claim type from which the user name is determined on the identity provider. The user name is used, for example, to identify data changes in One Identity Manager (column XUserInserted and XUserUpdated).  
Example: User Principle Name (UPN)  
http://schemas.xmlsoap.org/ws/2005/05/identity/claims/upn  
If you have determined the configuration data automatically, select a value from the list. |

13. Click **Next**.

14. On the **Create OAuth 2.0/OpenID Connect applications**, enter the application
information for the identity provider.

a. Click on next to the Applications input field.

b. On the General tab, enter the general information for the application.

Table 37: General information about the application

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Display name</td>
<td>Display name of the application.</td>
</tr>
<tr>
<td>Description</td>
<td>Spare text box for additional explanation.</td>
</tr>
<tr>
<td>Client ID</td>
<td>ID of the application on the identity provider. For native applications, enable the Default option. Example: urn:OneIdentityManager/Web</td>
</tr>
<tr>
<td>Shared Secret</td>
<td>Application-specific Shared Secret value used for authentication at the token endpoint.</td>
</tr>
<tr>
<td>Resource to request</td>
<td>URN of the resource to be requested, for example for ADFS. Only required if the identity provider requires this value.</td>
</tr>
<tr>
<td>Redirect URI</td>
<td>Forwarding address for redirection of applications. Example: urn:InstalledApplication</td>
</tr>
<tr>
<td>Default</td>
<td>Specifies whether this is a standard application for native applications.</td>
</tr>
</tbody>
</table>

c. On the Certificate tab, enter the information for the application certificate.

Table 38: Information about the application certificate

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Certificate endpoint</td>
<td>Uniform Resource Locator (URL) of the certificate end point on the authorization server. Example: <a href="https://localhost/RSTS/SigningCertificate">https://localhost/RSTS/SigningCertificate</a></td>
</tr>
<tr>
<td>Fingerprint</td>
<td>Fingerprint of the certificate used to verify the security token.</td>
</tr>
<tr>
<td>Subject of the certificate</td>
<td>Subject of the certificate used for verification. The subject or fingerprint must be set.</td>
</tr>
<tr>
<td>Certificate</td>
<td>Content of the certificate. It is used if no certificate is configured.</td>
</tr>
</tbody>
</table>

d. On the Authentication tab, enter the following information
<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authentication method</td>
<td>Authentication method at the token endpoint. Permitted values are:</td>
</tr>
<tr>
<td></td>
<td>• client_secret_basic (default value): HTTP basic authentication method. The Shared Secret is</td>
</tr>
<tr>
<td></td>
<td>transferred in the HTTP header.</td>
</tr>
<tr>
<td></td>
<td>• client_secret_post: The Shared Secret is transferred in the client_secret value of the POST-</td>
</tr>
<tr>
<td></td>
<td>Body.</td>
</tr>
<tr>
<td></td>
<td>• none: No authentication at the token endpoint.</td>
</tr>
<tr>
<td></td>
<td>• client_secret_jwt: The Shared Secret is transferred as a JSON web token (JWT).</td>
</tr>
<tr>
<td></td>
<td>• private_key_jwt: The Shared Secret is transferred as JWT. Encryption with the private key is</td>
</tr>
<tr>
<td></td>
<td>also applied.</td>
</tr>
<tr>
<td>Private key</td>
<td>Private key in text form, which is used for the authentication.</td>
</tr>
</tbody>
</table>

15. To create the identity provider and the application in the One Identity Manager database, click on Next.

16. Click Finish to complete the wizard.

Assigning OAuth 2.0/OpenID Connect configuration to web applications

To use the OAuth2.0/OpenID Connect and OAuth2.0/OpenID Connect (role-based) authentication modules in the One Identity Manager web applications, assign the OAuth2.0/OpenID Connect application to the web application.

To assign an OAuth2.0/OpenID Connect application to a web application

1. In the Designer, select Base data | Security settings | Web server configurations.
2. Select the web application in List Editor.
3. In the Properties edit view, assign the application in the OAuth2.0/OpenID Connect application selection list.
4. Save the changes.

TIP: For some web applications, for example the Web Portal, you can customize the OAuth2.0/OpenID Connect configuration in the configuration file (web.config). For more detailed information about configuring the Web Portal, see the One Identity Manager Installation Guide.
Displaying the configuration of the identity provider and the OAuth 2.0/OpenID Connect applications

**To display the configuration of an identity provider**

1. In the Designer, select **Base data | Security settings | OAuth 2.0/OpenID Connect configuration**.
2. Select the identity provider in List Editor. The configuration data is displayed on the following tabs in the edit view.
   - **General**: Displays the general configuration data of the identity provider.
   - **Certificate**: Shows the information about the identity provider certificate.
   - **Applications**: Displays the configuration of the OAuth 2.0/OpenID Connect applications.
   - **Columns for enabling**: Displays the table and the columns that identify a user account as activated.
   - **Columns for disabling**: Displays the table and the columns that identify a user account as deactivated.

**To display the configuration of an OAuth 2.0/OpenID Connect application**

1. In the Designer, select **Base data | Security settings | OAuth 2.0/OpenID Connect configuration**.
2. Select the identity provider in List Editor.
3. In the edit view, select the **Applications** tab.
4. To display the configuration of an application, select the OAuth 2.0/OpenID Connect application in the **Application** view.

**NOTE:**

Click on **Add** to add a new OAuth 2.0/OpenID Connect application to the configuration of the identity provider.

Click on **Remove** to remove an OAuth 2.0/OpenID Connect application that is no longer required from the configuration of the identity provider.

**Related topics**

- Creating the OAuth 2.0/OpenID Connect configuration on page 108
- Defining enabling and disabling columns for the determination of user accounts on page 114
Defining enabling and disabling columns for the determination of user accounts

In the determination of the user account for the OAuth 2.0/OpenID Connect authentication, the system checks whether the user account is enabled or disabled. You define which columns can mark a user account as enabled or disabled.

The columns of the table that you selected in the OAuth 2.0/OpenID Connect configuration of the identity provider in the **Column to search** are offered.

**To define which columns can enable a user account for login**

1. In the Designer, select **Base data | Security settings | OAuth 2.0/OpenID Connect configuration**.
2. In the List Editor, select the configuration.
3. Select the **Columns for enabling** tab in the edit view.
4. In the **Add assignment** view, assign the columns that enable the user account for logon.
5. Save the changes.

**To define which columns can disable a user account for login**

1. In the Designer, select **Base data | Security settings | OAuth 2.0/OpenID Connect configuration**.
2. In the List Editor, select the configuration.
3. Select the **Columns for disabling** tab in the edit view.
4. In the **Add assignment** view, assign the columns that disable the user account for logon.
5. Save the changes.
### Multi-factor authentication in One Identity Manager

#### Table 40: Multi-factor authentication configuration parameters

<table>
<thead>
<tr>
<th>Configuration parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>Extend the functional scope of One Identity Manager with a One Identity Hybrid Subscription, which offers a range of additional cloud functions and services. Use the universal Starling Two-Factor Authentication to protect administrator access. Force additional authentication when you request a critical access or to enable out-of-brand user verification for password requirements.</td>
<td></td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
</tbody>
</table>

You can set up multi-factor authentication for specific security-critical actions in One Identity Manager. You can use these, for example, for attestations or when approving requests in Web Portal.
For multi-factor authentication, One Identity Manager uses One Identity Starling Two-Factor Authentication. This service is supplied by default by means of a One Identity Hybrid Subscription. If your company does not use a One Identity Hybrid Subscription, select the conventional Starling Two-Factor Authentication integration. Use configuration parameters to specify which of the two solutions are applied in your company.

To be able to use multi-factor authentication

1. Register your company in Starling Two-Factor Authentication.
   For more detailed information, see the Starling Two-Factor Authentication documentation.

2. Specify which authentication solution is used.
   - To use One Identity Hybrid Subscription
     a. Start the Launchpad.
     b. Select *Connection to One Identity Hybrid Subscription* and click *Run*.
        The Starling Hybrid configuration wizard starts.
     c. Follow the instructions of the Starling Hybrid configuration wizard.
        The configuration parameters under *QER | Person | Starling* are enabled and the authentication information is entered.
   - To use conventional Starling Two-Factor Authentication integration
     a. In Designer, enable the *QER | Person | Defender* configuration parameter.
        - Enable the *QER | Person | Defender | ApiKey* configuration parameter and enter your company’s subscription key as the value for accessing the Starling Two-Factor Authentication interface.
        The default URL of the Starling 2FA API end point is already entered in the *QER | Person | Defender | ApiEndpoint* configuration parameter.

3. Enable assigning by event for the table *PersonHasQERResource*. For more information, see *Editing table properties* on page 117.

4. (Optional) Specify whether the security code must be requested from the Starling 2FA app. For more information, see *Requesting a security code* on page 118.

5. In Manager, enable the **New Starling 2FA token** service item. For more information, see *Preparing the Starling 2FA token request* on page 117.

If the user’s telephone number has changed, cancel the current Starling 2FA token and request a new one. If the Starling 2FA token is no longer required, cancel it anyway.

For detailed information, see the following guides:

<table>
<thead>
<tr>
<th>Theme</th>
<th>Guide</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preparing the IT Shop for multi-factor authen-</td>
<td>One Identity Manager IT Shop Admin-</td>
</tr>
</tbody>
</table>

---

For detailed information, see the following guides:
<table>
<thead>
<tr>
<th>Theme</th>
<th>Guide</th>
</tr>
</thead>
<tbody>
<tr>
<td>Setting up multi-factor authentication for attestation</td>
<td>One Identity Manager Attestation Administration Guide</td>
</tr>
<tr>
<td>Setting up Starling Two-Factor Authentication in the web project</td>
<td>One Identity Manager Web Application Configuration Guide</td>
</tr>
<tr>
<td>Requesting the Starling 2FA Token</td>
<td>One Identity Manager Web Portal User Guide</td>
</tr>
<tr>
<td>Requesting products requiring multi-factor authentication</td>
<td></td>
</tr>
<tr>
<td>Approving requests with multi-factor authentication</td>
<td></td>
</tr>
<tr>
<td>Attestation with multi-factor authentication</td>
<td></td>
</tr>
</tbody>
</table>

**Editing table properties**

> NOTE: If the Assign by event option is enabled, the HandleObjectComponent process is placed in the job queue as soon as a resource assignment is added to or removed from an employee.

**To enable assigning by event for a table**

1. In Designer, select the One Identity Manager Schema category.
2. Select the PersonHasQERResource table and start Schema Editor using the Show table definition task.
3. In the Table properties view, select the Table tab and enable the Assign by event option.
4. Save the changes.

For detailed information about editing table definitions, see the One Identity Manager Configuration Guide.

**Preparing the Starling 2FA token request**

One Identity Manager users must be registered with Starling Two-Factor Authentication in order to use multi-factor authentication. To register, a user must request the Starling 2FA Token in the Web Portal. Once the request has been granted approval, the user receives a link to the Starling Two-Factor Authentication app and a Starling 2FA user ID. The app
generates one-time passwords, which are required for authentication. The Starling 2FA user ID is saved in the user's employee master data.

NOTE: The user’s default email address, mobile phone and country must be stored in their master data. This data is required for registering.

To facilitate requesting a Starling 2FA token

1. Select the category IT Shop | Service catalog | Predefined.
2. Select New Starling 2FA token in the results list.
3. Select Change master data.
4. Disable Not available.
5. Save the changes.

The Starling 2FA token request must be granted approval by the request recipient’s manager.

Requesting a security code

Table 41: Configuration Parameter for Requesting Starling 2FA Security Codes

<table>
<thead>
<tr>
<th>Configuration parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
</tbody>
</table>

If the security code is requested for an attestation, request or request approval, the user decides how the security code is sent. The following options are available:

- By Starling 2FA app
- By SMS
- By phone call

By default, Starling 2FA is forced to send the OTP by SMS or by phone call if the user has selected one of these options. However, for security reasons, the user should use the Starling 2FA app to generate the OTP. If the app is installed on the user’s mobile phone, Starling 2FA can refuse the SMS or phone demand and the user must generate the OTP using the app.
To use this method

- If you use the One Identity Hybrid Subscription, enable the QER | Person | Starling | DisableForceParameter configuration parameter in Designer.
- OR -

- If you use classic Starling Two-Factor Authentication, integration, enable the QER | Person | Defender | DisableForceParameter configuration parameter in Designer.

Starling 2FA can refuse to transmit the OTP by SMS or phone call if the Starling 2FA app is installed on the phone. Then the OTP must be generated by the app.

If the configuration parameter is not set (default), Starling 2FA is forced to send the OTP by SMS or phone call.

Allowing approval decisions using the Starling 2FA app

Table 42: Configuration parameters for approving with the Starling 2FA app

<table>
<thead>
<tr>
<th>Configuration parameter</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
<tr>
<td>QER</td>
<td>Person</td>
</tr>
</tbody>
</table>

To provide approvers who are temporarily unable to access One Identity Manager tools, with the option of making approval decisions for requests, you can set up approval by Starling 2FA app. This means, approvers are prompted by the Starling 2FA app to approve or deny a request. This option is only available if you use One Identity Hybrid Subscription for multi-factor authentication and the approvers are registered with Starling Two-Factor Authentication.

To use the Starling 2FA app for approval decisions

- In the Designer, set the QER | Person | Starling | UseApprovalAnywhere configuration parameter.
  The approver must make the approval decision within 5 minutes. If this times out, the approver must use the Web Portal to approve the request.

  To change the timeout, modify the value in the QER | Person | Starling | UseApprovalAnywhere | SecondsToExpire configuration parameter. Enter a timeout in seconds.
Granulated permissions for the SQL Server and database

To implement a One Identity Manager database or a One Identity Manager History Database on an SQL Server or a managed instance in Azure SQL Database, you are provided with SQL Server logins and database users for administrative users, configuration users and end users. Permissions at server and database level are matched to suit the user’s tasks.

Normally, you cannot edit users and permissions. It may be necessary to set up an additional database user to use a One Identity Manager History Database.

For detailed information about users and their permissions, see the One Identity Manager Installation Guide and the One Identity Manager Data Archiving Administration Guide.

Related topics

- Minimum access levels of One Identity Manager tools on page 120
- Displaying database server logins on page 122
- Displaying users’ access levels on page 123
- Displaying server and database permissions on page 123

Minimum access levels of One Identity Manager tools

**NOTE:**

- Connections that do not use the expected access level for SQL Server logins are not shown in the connection dialog.
- If you select an existing database connection in the connections dialog, the access level of the login to be used is shown in a tooltip.

You require the following minimum access level for One Identity Manager tools.
Table 43: Access level for One Identity Manager tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Minimum access level</th>
</tr>
</thead>
<tbody>
<tr>
<td>Analyzer</td>
<td>End user</td>
</tr>
<tr>
<td>Application server</td>
<td>End user or configuration user (depending on the application server's task)</td>
</tr>
<tr>
<td>AppServer.Installer.CMD.exe</td>
<td>Configuration user</td>
</tr>
<tr>
<td>API Designer</td>
<td>Configuration user</td>
</tr>
<tr>
<td>API Server</td>
<td>End user</td>
</tr>
<tr>
<td>Configuration Wizard</td>
<td>Administrative user</td>
</tr>
<tr>
<td>Crypto Configuration</td>
<td>Configuration user</td>
</tr>
<tr>
<td>Data Import</td>
<td>End user</td>
</tr>
<tr>
<td>DataImporterCMD.exe</td>
<td>Configuration user (saves import definition)</td>
</tr>
<tr>
<td>Database Compiler</td>
<td>Configuration user</td>
</tr>
<tr>
<td>DCompilerCMD.exe</td>
<td>Configuration user</td>
</tr>
<tr>
<td>Database Transporter</td>
<td>Configuration user</td>
</tr>
<tr>
<td>DBTransporterCMD.exe</td>
<td>Administrative user</td>
</tr>
<tr>
<td>DBClone</td>
<td>Administrative user</td>
</tr>
<tr>
<td>DBComparer</td>
<td>Configuration user</td>
</tr>
<tr>
<td>Designer</td>
<td>Configuration user</td>
</tr>
<tr>
<td>Job Queue Info</td>
<td>Configuration user</td>
</tr>
<tr>
<td>Launchpad</td>
<td>End user</td>
</tr>
<tr>
<td>License Meter</td>
<td>End user</td>
</tr>
<tr>
<td>Manager</td>
<td>End user</td>
</tr>
<tr>
<td><strong>Some application that are started from the Launchpad, required different access levels</strong></td>
<td></td>
</tr>
<tr>
<td>HistoryDB Manager</td>
<td>End user</td>
</tr>
<tr>
<td>Object Browser</td>
<td>End user</td>
</tr>
</tbody>
</table>
| One Identity Manager Service              | Configuration users for process collection with the
### Tool | Minimum access level
--- | ---
MSSQLJobProvider | MSSQLJobProvider
Report Editor | Configuration user
Schema Extension | Configuration user
SchemaExtensionCmd.exe | Configuration user
Software Loader | Configuration user
SoftwareLoaderCMD.exe | Configuration user
Synchronization Editor | Configuration user
System Debugger | Configuration user
Web Designer | Configuration user
Web Designer Configuration Editor | Configuration user
VI.WebDesigner.CompilerCmd.exe | Configuration user
WebDesigner.InstallerCMD.exe | Configuration user
Web Portal | End user
Password Reset Portal | End user
Operations Support Web Portal | End user
Quantum.MigratorCmd.exe | Administrative user

**Related topics**
- Granulated permissions for the SQL Server and database on page 120

## Displaying database server logins

**To display login information**

1. In Designer, select the category **Base data | Security settings | Database server permissions | Database server login**.
2. Select the database server login. The following information is displayed:
   - **Login name**: The user's SQL Server login.
   - **Database server login**: Type of database user.
   - **Access level**: The access level for logging in. The access level **End user**, **Configuration user**, **Administrative user**, **System administrator** or **Unknown** is displayed.
3. To show the database roles and server roles that are assigned, select the **Database or server role** tab.

**Displaying users' access levels**

<table>
<thead>
<tr>
<th>NOTE:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- If you select an existing database connection in the login dialog, the access level of the login to be used is shown in a tooltip.</td>
</tr>
<tr>
<td>- Some user interfaces expect configuration user permissions at least. Logging in as an end user is not possible in this case.</td>
</tr>
</tbody>
</table>

**To find the access level of the logged in user**

- To display user information, double-click the icon in the program status bar.
  
  On the **System user** tab, in the **SQL access level** field, you will see the access level for the current login. The access level **End user, Configuration user, Administrative user, System administrator** or **Unknown** is displayed.

**Related topics**

- [Displaying database server logins](#) on page 122

**Displaying server and database permissions**

Server and database permissions are predefined and cannot be modified.

<table>
<thead>
<tr>
<th>NOTE:</th>
</tr>
</thead>
<tbody>
<tr>
<td>The <strong>End user role</strong> database role is permitted for custom schema extensions.</td>
</tr>
</tbody>
</table>

**To display server and database permissions**

- In the Designer, in the **Base data | Security settings | Database server permissions | Database server login** category, select the server or database role.
  
  This opens the List Editor showing a list of permissions.
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 or other inquiries, visit https://www.oneidentity.com/company/contact-us.aspx or call +1-800-306-9329.

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
Index

A
application
  assign authentication module 95
  configuration data 101
application role 8
  administrators 10, 12, 14-21, 23-24
  approver 17-18
  approver (IT) 17-18
  assign employees 28-29
  assign extended property 32
  assign reports 31
  attestors 12, 14, 17-18, 20
  auditors 12
authorize as One Identity Manager
  administrator 25
base roles 10
  administrators 10, 25
  employee managers 10
  everyone (change) 10
  everyone (lookup) 10
  internal permissions 10
  operations support 10
chief approval team 15, 20
cloud administrators 23
compliance and security officer 11
conflicting 30
custom 24
  administrators 24
  manager/supervisor 24
dynamic 29
edit 25-26
employee managers 10
exception approvers 14
extend edit permissions 28
identity and access governance 11-12, 14-16
  attestation 15
    administrators 15
    chief approval team 15
  auditors 12
  company policies 14
    administrators 14
    attestors 14
    exception approvers 14
    policy supervisors 14
  compliance & security officer 11
  identity audit 12
    administrators 12
    attestors 12
    maintenance SAP features 12
    rule administrators 12
    subscribable reports 16
    administrators 16
identity management 17
  business roles 17
    administrators 17
    approver 17
    approver (IT) 17
    attestors 17
    employees 19
    administrators 19
  management level 17
organizations 18
administrators 18
approver 18
approver (IT) 18
attestors 18
implement 25
internal permissions 10
management level 17
overview 9
permissions group 26, 28
policy supervisors 14
product owner 20
reports 32
request and fulfillment 20
  IT Shop 20
    administrators 20
    attestors 20
    chief approval team 20
    product owner 20
rule administrators 12
target system manager 21
target systems
  administrators 21
target system manager 21
Universal Cloud Interface
  administrator 23
assignment resource
  for an application role 32
authentication
  check 105
authentication module
  account-based system user 76
  Active Directory user account 77
  Active Directory user account
    (dynamic) 81
Active Directory user account
  (manual input/role-based) 80
Active Directory user account
  (manual) 79
Active Directory user account (role-based) 78
assign application 95
component authentication module 91
crawler 92
employee 72
employee (dynamic) 73
employee (role-based) 72
enable 95
generic single sign-on (role-based) 70
HTTP header 86
HTTP header (role-based) 87
initial data 97
LDAP user account (dynamic) 84
LDAP user account (role-based) 82
OAuth 2.0/OpenID Connect 88
OAuth 2.0/OpenID Connect (role-based) 89
password reset 92
password reset (role-based) 93
synchronization authentication module 90
system users 69
user account 74
user account (role-based) 75
web agent authentication module 91
authorizations
  database 120
database role 123
server role 123
SQL Server 120
D
database role
  display permissions 123
database server
  access level 122
  authorizations 120
  database user 122
  login 122
decision
  by app 119
dynamic role
  application role 29

E
employee
  authorize as One Identity Manager administrator 25
event
  object event 67
  program function 67

L
Launchpad
  actions
    program function 68

M
Multi-factor authentication 115

O
OAuth 2.0/OpenID Connect
  application 113
  authentication 106
  authentication module 88-89
  certificate 108
  configuration 106, 113
  configurations 108
  disabling columns 114
  enabling columns 114
  identity provider 108, 113
  openid 108
  scope 108
  Shared Secret 108
  use case 108
  web application 112
object
  permissions 59
  object event 67
    program function 67
One Identity Hybrid Subscription 115
One Identity Hybrid Subscription security code 118
One Identity Starling software-as-a-service 115

P
permissions
  column permissions 55
  copying 56
  determine 38
  edit 51
  object 59
  permissions filter 53
  permissions group 51
  rules 38
  simulation 57
  table 52
Table permissions 53
permissions group

Permissions Editor 51
assign application 60
dependencies 42-43
hierarchy 42
only use for role-based authentication 41

permissions 51
predefined 35
program function 65-67
QBM_BaseRights 35
QER_OperationsSupport 35
role-based 35
setting up 40-41, 45
vi_4_ADMIN_LOOKUP 35
VI_4_ALLUSER 35
VI_Everyone 35
VI_View 35
vid 35
VID_Features 35
program function 62, 65, 67
Launchpad actions 68
permissions group 65-67
permissions group
program function 65
script 65
task definition 66

Security code

demand
by app 118
by call 118
by text 118
server role
display permissions 123
Starling 2FA 115, 117
Starling 2FA security code 118
Starling Two-Factor Authentication 115
system user
administrative user 48
dynamic 35
dynamically determine 101
employees 50
logins 48
password 47-48
password never expires 47-48
permissions group 49
predefined 35
read-only 48
sa 35
service account 48
setting up 45-46
support 35
synchronization 35
user 50
viadmin 35
viHelpdesk 35
viITShop 35
system users
dynamic 50

S
script
program function 65
T

table
  permissions 52

task definition
  program function 66

token 117

U

use case
  assign permissions group 60

user
  access level 123
  authentication module 60
  dynamic 60
  permissions 60
  permissions groups 60
  program function 60
  read permissions 60
  system user 60