One Identity Manager 9.3

Administration Guide for the SAP R/3 Compliance Add-on

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

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

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

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

Patents

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

Trademarks

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

Legend

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

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

One Identity Manager Administration Guide for the SAP R/3 Compliance Add-on
Updated - 16 December 2024, 13:17

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

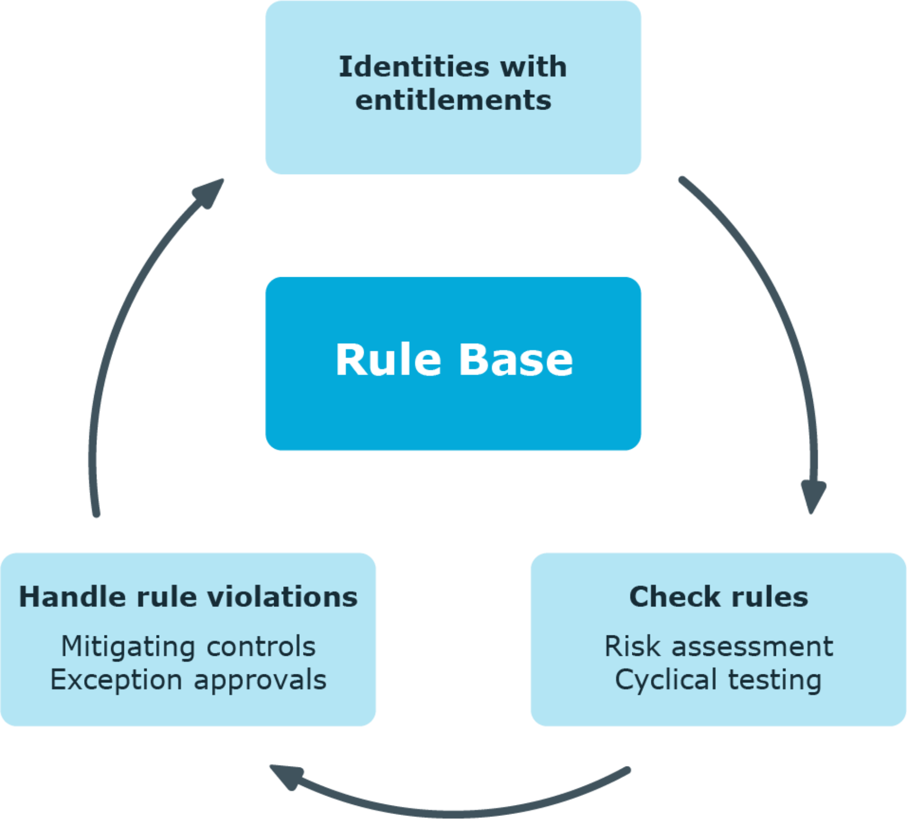
Contents

1

SAP functions and identity audit

One Identity Manager can be used to define rules that maintain and monitor regulatory requirements and automatically deal with rule violations. Define compliance rules to test entitlements or combinations of entitlements in the context of identity audit for identities in the company. On the one hand, existing rule violations can be found by checking rules. On the other hand, possible rule violations can be preemptively identified and thus prevented.

Figure 1: Identity audit in One Identity Manager



In addition to rule checking, One Identity Manager offers a detailed examination of effective authorizations of SAP user accounts for SAP R/3 target systems. Linking SAP user accounts to identities allows combinations of SAP authorizations that an identity receives through different SAP user accounts to be checked. Invalid or potentially dangerous authorizations and combinations of them can easily be recognized this way and the necessary action taken.

SAP authorizations are checked on the basis of the authorization objects permitted for an SAP user account. SAP roles and profile assignments determine which authorization objects are permitted. To check whether invalid or potentially dangerous SAP authorizations are assigned within the company, define SAP functions that describe invalid combinations of authorization objects. One Identity Manager finds all the SAP roles, profiles, and profiles that have exactly these authorization objects assigned to them. Use compliance rules to determine the identities that are linked to these user accounts and therefore have invalid authorizations.

If identities are granted SAP authorizations through IT Shop requests, the invalid authorizations can be detected and handled respectively when the request is made with the appropriate approval processes. For more information about approval processes in the IT Shop, see the *One Identity Manager IT Shop Administration Guide*.

Based on this information, you can made corrections to data in One Identity Manager and transfer them to the connected SAP R/3 systems. The integrated report function in One Identity Manager can be used to provide information for the appropriate tests.

> NOTE: Compliance Rules Module and SAP R/3 Compliance Add-on Module must be installed in order to set up and analyze SAP functions.

> NOTE: You cannot use SAP functions to check the authorizations in the central user administration client.
>
> ### Related topics
>
> - Prerequisites for setting up SAP functions on page 9
> - Basics of the authorization check on page 14
> - Setting up SAP functions on page 31
> - Compliance rules for SAP functions on page 63

One Identity Manager users for managing SAP functions

The following users are used for the administration of SAP functions.

Table 1: Users

Users	Tasks
Compliance rules administrators	Administrators must be assigned to the **Identity & Access Governance | Identity Audit | Administrators** application role. Users with this application role:

Users	Tasks
	• Enter base data for setting up company policies. • Create compliance rules and assign rule supervisors to them. • Can start rule checking and view rule violations as required. • Create reports about rule violations. • Define SAP functions and assign these to managers. • Define function instances and variables sets for SAP functions. • Enter mitigating controls. • Create and edit risk index functions. • Monitor Identity Audit functions. • Administer application roles for rule supervisors, exception approvers and attestors. • Set up other application roles as required.
Responsible for maintaining SAP functions.	Those responsible for maintaining the SAP functions must be assigned to the **Identity & Access Governance | Identity Audit | Maintenance SAP Functions** application role or a child application role. Users with this application role: • Are responsible for SAP function contents. • Edit working copies of function definitions for which they are responsible. • Define function instances and variables sets for SAP functions. • Assign mitigating controls.
One Identity Manager administrators	One Identity Manager administrator and administrative system users Administrative system users are not added to application roles. One Identity Manager administrators: • Create customized permissions groups for application roles for role-based login to administration tools in the Designer as required. • Create system users and permissions groups for non role-based login to administration tools in the Designer as required.

Users	Tasks
	• Enable or disable additional configuration parameters in the Designer as required. • Create custom processes in the Designer as required. • Create and configure schedules as required.
Compliance and security officer	Compliance and security officers must be assigned to the **Identity & Access Governance | Compliance & Security Officer** application role. 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, critical SAP functions and risk index functions. • Edit attestation polices.

Prerequisites for setting up SAP functions

All the information regarding SAP authorizations, SAP users, SAP roles, and SAP profiles must be transferred to the One Identity Manager database so that One Identity Manager can test the effective SAP authorizations based on SAP functions.

Setting Up SAP Functions

1. In the Designer, set the **QER | ComplianceCheck** and the **TargetSystem | SAPR3 | SAPRights** configuration parameters.

   > NOTE: If you disable the configuration parameter at a later date, model components and scripts that are no longer required, are disabled. SQL procedures and triggers are still carried out. For more information about the behavior of preprocessor relevant configuration parameters and conditional compiling, see the *One Identity Manager Configuration Guide*.

2. Set up a synchronization project for synchronizing the necessary SAP schema types and start synchronization.

Detailed information about this topic

- Setting up a synchronization project for synchronizing SAP authorization objects on page 11
- Setting up SAP functions on page 31

Configuration parameters for SAP functions

Use configuration parameters to configure the behavior of the system's basic settings. One Identity Manager provides default settings for various configuration parameters. Check the configuration parameters and modify them as necessary to suit your requirements.

Configuration parameters are defined in the One Identity Manager modules. Each One Identity Manager module can also install configuration parameters. In the Designer, you can find an overview of all configuration parameters in the **Base data > General > Configuration parameters** category.

> IMPORTANT: The **TargetSystem | SAPR3 | SAPRights | TestWithoutTCD** configuration parameter will be deleted in a future version of One Identity Manager and cannot be set anymore in version 9.3.
>
> When updating the One Identity Manager database from a version older than 9.3 to version 9.3, the configuration parameter setting is transferred without alteration. This functionality stays the same. However, the configuration parameter can neither be set nor cleared in the current One Identity Manager version.
>
> For more information, see Ignoring SAP applications on page 18.

Related topics

- Configuration parameters for SAP functions on page 71

Setting up a synchronization project for synchronizing SAP authorization objects

SAP authorizations are verified on the basis of the SAP applications permitted for an SAP user account and the associated authorization objects. Authorization objects and SAP applications must be loaded into the One Identity Manager database first before you can create SAP functions. For each client, create a synchronization project for synchronizing the necessary schema types. A separate project template is required for this.

Use the Synchronization Editor to configure synchronization between the One Identity Manager database and SAP R/3 environment.

> NOTE: Just one synchronization project can be created per target system and default project template used.

To set up a synchronization project for SAP authorization objects.

1. Set up an initial synchronization project as described in the One Identity Manager Administration Guide for Connecting to SAP R/3. The following special features apply:

   > NOTE: You cannot use SAP functions to check the authorizations in the central user administration client. Set up the synchronization project for one client only with the CUA status `None`.

   a. In the project wizard on the **Select project template** page, select the **SAP R/3 authorization objects** project template.

   b. The **Restrict target system access** page is not displayed. The target system is only loaded.

   For more information, see the *One Identity Manager Administration Guide for Connecting to SAP R/3*.

2. Configure and set a schedule to run synchronization regularly.

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

Related topics

- Default project template for the SAP R/3 Compliance Add-on Module on page 73
- Referenced SAP R/3 tables and BAPI calls on page 75
- Synchronizing very large numbers of SAP authorizations on page 13
- Objects in USOBHASH table not completely loaded on page 12

Objects in USOBHASH table not completely loaded

When synchronizing SAP authorization objects, not all objects in the USOBHASH table are loaded into the One Identity Manager database.

Probable reason

Changed implementation of the ABAP function AUTH_TRACE_GET_USOBHASH as of SAP BASIS version 7.57 (SAP S/4HANA 2022).

Solution

- Import the current SAPTRANSPORT_70.ZIP transport into the SAP R/3 system you want to synchronize.

  One Identity Manager version 9.1.3 or later provides an updated BAPI transport SAPTRANSPORT_70.ZIP. This uses the /VIAENET/LISTUSOBHASH function module instead of the AUTH_TRACE_GET_USOBHASH SAP module. When it accesses an SAP R/3 system, the SAP R/3 connector checks whether the /VIAENET/LISTUSOBHASH function module exists and uses that. This synchronizes all objects in the USOBHASH table.

If the function module is not available, the connector uses the AUTH_TRACE_GET_USOBHASH SAP module.

The synchronization log records whether the /VIAENET/LISTUSOBHASH function module is used.

Related topics

- Setting up a synchronization project for synchronizing SAP authorization objects on page 11

Synchronizing very large numbers of SAP authorizations

If your SAP R/3 environment contains a very large number of `ProfileHasAuthObjectField` authorizations (several million), synchronization might quit unexpectedly or just not complete.

Solution

If the total number of authorizations is too large for processing, synchronization can be divided into several synchronization steps.

To split synchronization of `ProfileHasAuthObjectField` into several steps

1. In the Synchronization Editor, edit the synchronization workflow for synchronizing SAP authorization objects (default: **Initial Synchronization**).
2. Enable the **profileHasAuthObjectFieldPart1**, **profileHasAuthObjectFieldPart2**, **profileHasAuthObjectFieldPart3**, and **profileHasAuthObjectFieldPart4** synchronization steps.
    - If these synchronization steps are not available, first apply the VPR#37380 patch.

      This patch creates the synchronization steps in synchronization projects that were set up in versions of One Identity Manager older than 9.2.
3. Disable the **profileHasAuthObjectField** synchronization step.
4. Save the changes.

In subsequent synchronizations, all `ProfileHasAuthObjectField` objects are divided into four blocks and processed independently of each other.

For more information about editing synchronization steps and applying patches, see *One Identity Manager Target System Synchronization Reference Guide*.

Related topics

- Setting up a synchronization project for synchronizing SAP authorization objects on page 11

3

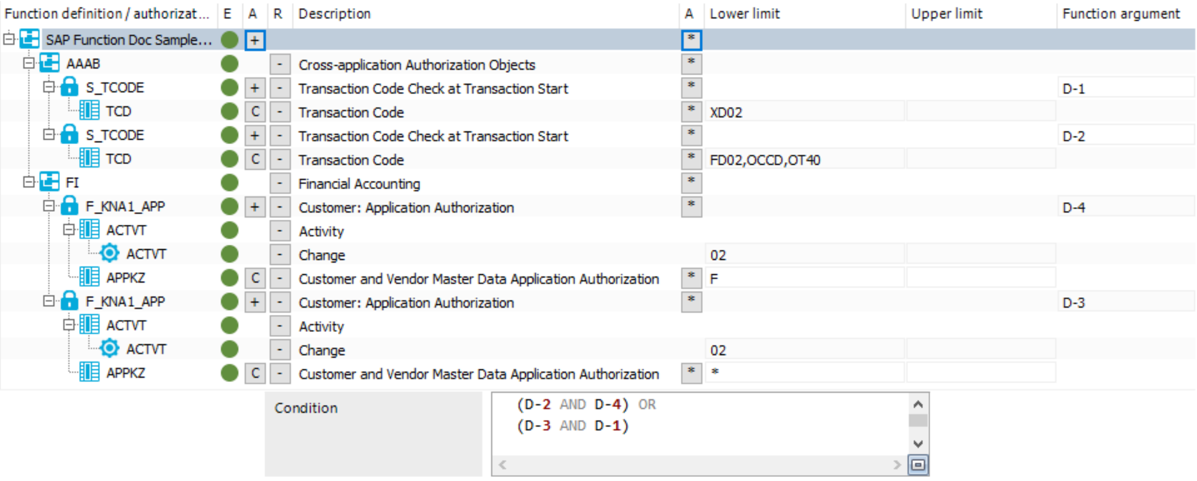
Basics of the authorization check

Apart from being able check rules, One Identity Manager offers detailed checking for SAP R/3 target systems of authorizations in effect for SAP users. To do this, One Identity Manager determines detailed authorizations of all SAP roles, profiles, and user accounts and checks whether they are permitted. You define the criteria for this check in the SAP function authorization definitions.

SAP authorizations are checked on the basis of the authorization objects permitted for a technical profile (`AuthLevel`). An authorization definition groups all the authorization objects to check together with specific values. One Identity Manager compares all authorization objects assigned to individual profiles against the authorization definition. This checks whether all function elements and their values defined for an authorization object occur within one technical profile. The single profiles that contain these technical profiles determine all the SAP roles, composite roles, and user accounts to which authorization objects are assigned.

An authorization definition can either contain a single authorization object or a complex combination of multiple authorization objects. Multiple authorization objects are linked together with logical operators. The function definition is used to save all the authorization object links as a condition. A function argument is generated for each authorization object to clearly identify the authorization objects in the condition. These function arguments are used to formulate the condition.

Figure 2: Example of an authorization definition with condition



Function definition / authorizat...	E	A	R	Description	A	Lower limit	Upper limit	Function argument
SAP Function Doc Sample...	●	+			*			
AAAB	●		-	Cross-application Authorization Objects	*			
S_TCODE	●	+	-	Transaction Code Check at Transaction Start	*			D-1
TCD	●	C	-	Transaction Code	*	XD02		
S_TCODE	●	+	-	Transaction Code Check at Transaction Start	*			D-2
TCD	●	C	-	Transaction Code	*	FD02,OCCD,OT40		
FI	●		-	Financial Accounting	*			
F_KNA1_APP	●	+	-	Customer: Application Authorization	*			D-4
ACTVT	●		-	Activity				
ACTVT	●		-	Change		02		
APPKZ	●	C	-	Customer and Vendor Master Data Application Authorization	*	F		
F_KNA1_APP	●	+	-	Customer: Application Authorization	*			D-3
ACTVT	●		-	Activity				
ACTVT	●		-	Change		02		
APPKZ	●	C	-	Customer and Vendor Master Data Application Authorization	*	*		

Condition

```
(D-2 AND D-4) OR
(D-3 AND D-1)
```

The requirements and guidelines used for defining SAP profiles in your SAP R/3 environment determine which authorizations are checked by an authorization definition and how many authorization objects are combined in an authorization definition.

Linking SAP user accounts to identities allows combinations of SAP authorizations that an identity receives through different SAP user accounts to be checked. Invalid or potentially dangerous authorizations and combinations of them can easily be recognized this way and the necessary action taken. There are SAP functions included in the rule condition of compliance rules for this check. For more information, see Compliance rules for SAP functions on page 63.

Authorization checks with SAP functions can provide answers to the following questions:

1. Are there SAP roles or user accounts with invalid authorization combinations?
2. Are there identities that own invalid authorization combinations through their SAP user accounts.

Related topics

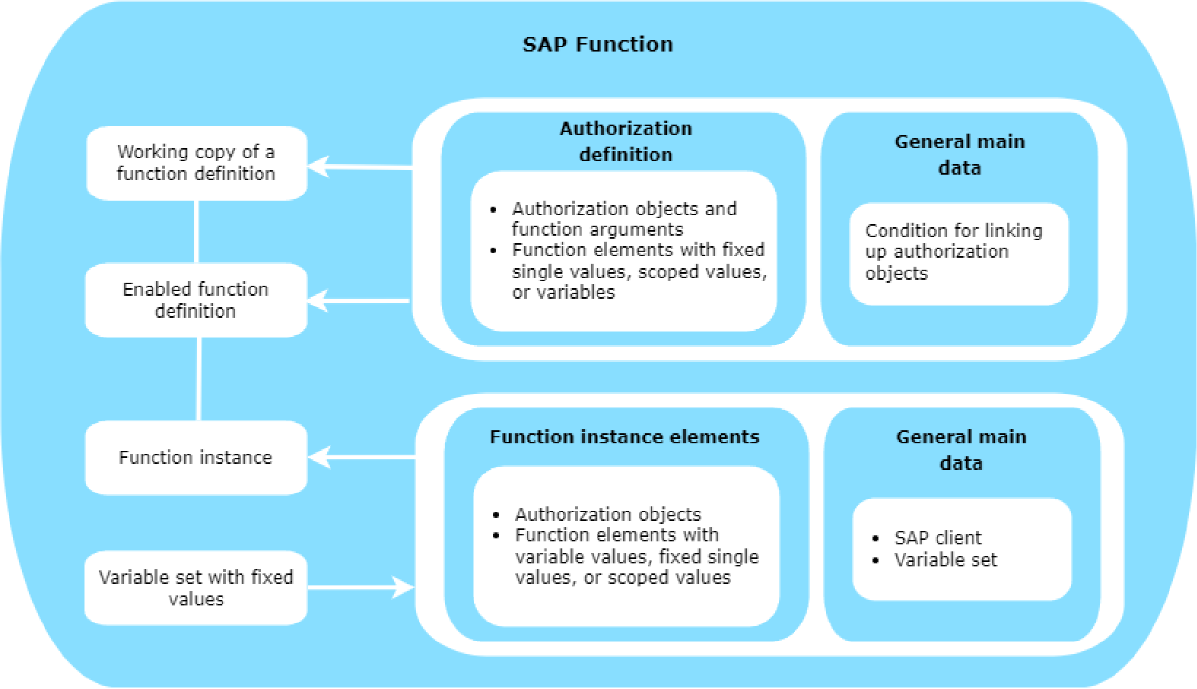
- Authorization definition properties and their values on page 37
- Combining authorization objects and function elements on page 16
- Ignoring SAP applications on page 18
- Examples of SAP functions on page 19
- Recommendations for setting up SAP functions on page 29

The SAP function structure

Create function definitions, function instances, and variable sets for SAP functions. You can use an SAP function for different instances. To do this, use variables in the function definition. Fixed variable values are grouped in variable sets and used in the function instances.

A function definition contains the authorization definition as well as general main data. An authorization definition contains at least one authorization object. Each authorization object consists of at least one function element (activity or authorization field) with fixed values. These are given as single values or as upper and lower limits. Function elements can be listed more than once per authorization object.

Figure 3: SAP function elements


SAP Function
Working copy of a function definition
Enabled function definition
Function instance
Variable set with fixed values
Authorization definition
Authorization objects and function arguments
Function elements with fixed single values, scoped values, or variables
General main data
Condition for linking up authorization objects
Function instance elements
Authorization objects
Function elements with variable values, fixed single values, or scoped values
General main data
SAP client
Variable set

Related topics

- Setting up SAP functions on page 31

Combining authorization objects and function elements

All authorization objects, function elements, and values from the authorization definition are logically combined together for the authorization check. The following rules apply:

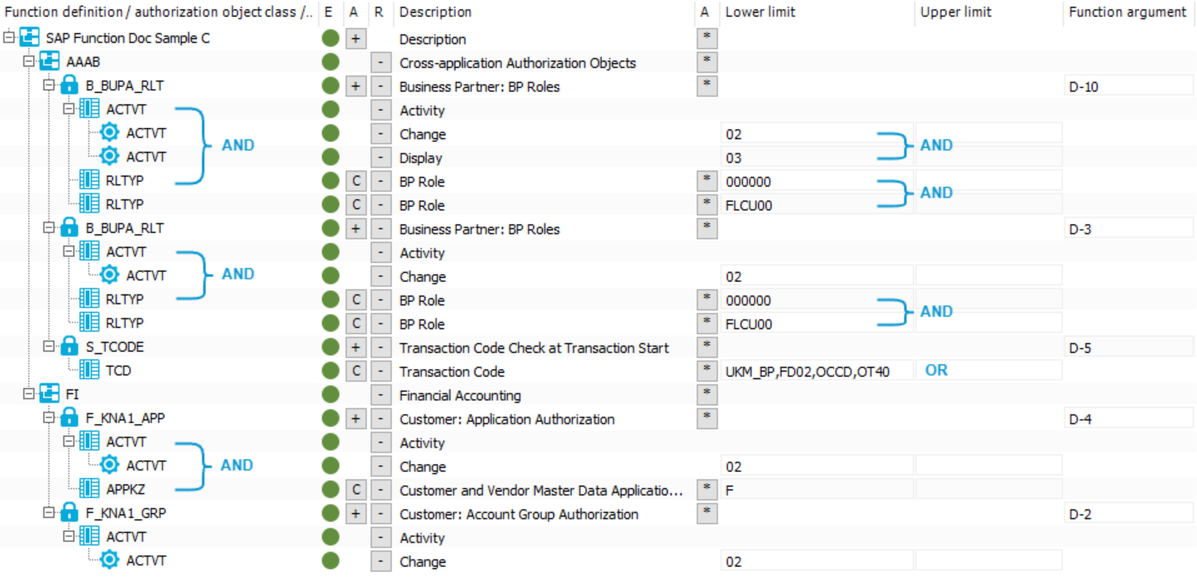
- All different function elements that belong to an authorization object are AND-ed together (AND).
- Different values of one and the same function element can be AND-ed as well as OR-ed together (OR).
- Function arguments can be either AND-ed or OR-ed together.

The following rules apply to function arguments:

- Each function argument is permitted for use in an authorization definition for just one authorization object.

- Each authorization object is assigned to exactly one function argument with its function elements and values.
- An authorization object can be used multiple times with different values within an authorization definition. An new function argument is created for each instance.

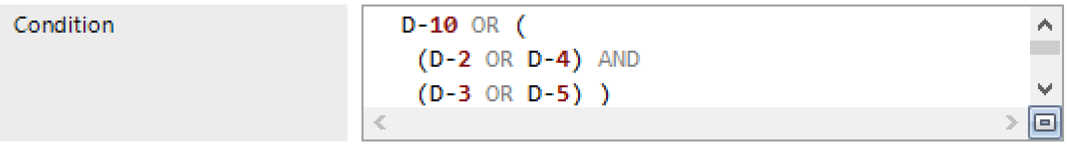
Figure 4: Combining function elements and values together



Function definition / authorization object class /..	E	A	R	Description	A	Lower limit	Upper limit	Function argument
SAP Function Doc Sample C	●	+		Description	*			
AAAB	●		-	Cross-application Authorization Objects	*			
B_BUPA_RLT	●	+	-	Business Partner: BP Roles	*			D-10
ACTVT	●		-	Activity				
ACTVT	●		-	Change		02		
ACTVT	●		-	Display		03		
RLTYP	●	C	-	BP Role	*	000000		
RLTYP	●	C	-	BP Role	*	FLCU00		
B_BUPA_RLT	●	+	-	Business Partner: BP Roles	*			D-3
ACTVT	●		-	Activity				
ACTVT	●		-	Change		02		
RLTYP	●	C	-	BP Role	*	000000		
RLTYP	●	C	-	BP Role	*	FLCU00		
S_TCODE	●	+	-	Transaction Code Check at Transaction Start	*			D-5
TCD	●	C	-	Transaction Code	*	UKM_BP,FD02,OCCD,OT40		
FI	●		-	Financial Accounting	*			
F_KNA1_APP	●	+	-	Customer: Application Authorization	*			D-4
ACTVT	●		-	Activity				
ACTVT	●		-	Change		02		
APPKZ	●	C	-	Customer and Vendor Master Data Applicatio...	*	F		
F_KNA1_GRP	●	+	-	Customer: Account Group Authorization	*			D-2
ACTVT	●		-	Activity				
ACTVT	●		-	Change		02		

The SAP function stores the combined function arguments in a condition. You can also use brackets in this condition to group function arguments.

Figure 5: Condition with combined function arguments



Condition

```
D-10 OR (
 (D-2 OR D-4) AND
 (D-3 OR D-5) )
```

For example, the function definition in these figures determines SAP roles with the following authorizations:

- Role **A** has
  - a technical profile with the **B_BUPA_RLT** authorization object with the activities **02** and **03** and the **RLTYP** authorization field with the values **000000** and **FLCU00**
- Role **B** has
  - a technical profile with the **F_KNA1_GRP** authorization object with the activity **02** and
  - a technical profile with the **S_TCODE** authorization object with the **TCD** authorization field with the value **FD02**

Related topics

- Basics of the authorization check on page 14

Ignoring SAP applications

The **TargetSystem | SAPR3 | SAPRights | TestWithoutTCD** configuration parameter is evaluated by authorization checks. The configuration parameter specifies whether the authorization check ignores the authorization objects required to identify SAP applications.

> IMPORTANT: The **TargetSystem | SAPR3 | SAPRights | TestWithoutTCD** configuration parameter will be deleted in a future version of One Identity Manager and cannot be set anymore in version 9.3.
>
> When updating the One Identity Manager database from a version older than 9.3 to version 9.3, the configuration parameter setting is transferred without alteration. This functionality stays the same. However, the configuration parameter can neither be set nor cleared in the current One Identity Manager version.
>
> To change the processing logic of the authorization definition, change the logical expression that is entered as a condition in each function definition.
>
> Conditions in existing function definitions from versions older than 9.3 are calculated when the One Identity Manager database is updated. This takes the configuration parameter setting into account.
>
> - After updating the database, check whether the authorization definitions and the generated conditions meet your requirements.

The configuration parameter is not set (default)

Behavior in version 9.3:

- The authorization check takes all authorization objects from the authorization definition into account and links them according to the condition.

Behavior when transferring function definitions from versions older than 9.3:

- Create a function argument for each authorization object in a function definition. The properties of the authorization objects determine the names of the function arguments.

  You can rename the function arguments as required in the Manager.
- There is a generated condition. In the condition, group all the function arguments that belong to an SAP application inside a pair of brackets and AND them with each other. All brackets are OR-ed together.

The configuration parameter is set

Behavior in version 9.3:

- The authorization check ignores the authorization objects required to identify SAP applications. It does not take the following authorization objects and function elements into account:
  - External service: S_Service with SRV_NAME and SRV_TYPE
  - TADIR object: S_START with AUTHOBJNAM, AUTHOBJTYP, and AUTHPGMID
  - RFC function module: S_RFC with RFC_NAME and RFC_TYPE
  - Transaction: S_TCODE with TCD

Behavior transferring function definitions from versions older than 9.3:

- Create a function argument for each authorization object in a function definition. The properties of the authorization objects determine the names of the function arguments.

  You can rename the function arguments as required in the Manager.
- There is a generated condition. All function arguments AND-ed in the condition.

Related topics

- Importing function definitions from versions older than 9.3 on page 60
- Renaming function arguments on page 44
- Basics of the authorization check on page 14
- Examples of SAP functions on page 19

Examples of SAP functions

If you create an authorization definition, you need to think about which authorization combinations are not compliant. You can differentiate between two use cases:

1. Are there SAP roles or user accounts with invalid authorization combinations?

   Create an SAP function for authorizations that cannot occur together with an SAP role or an SAP user account. The authorization check identifies all SAP roles and user accounts where the sum total of their authorizations have this invalid combination of authorizations.
2. Are there identities that own invalid authorization combinations through their SAP user accounts?
   1. Create a compliance rule that checks whether there are identities with SAP user accounts that match the SAP functions.
   2. Create different SAP functions for authorizations that in combination are invalid. Create compliance rules that combine these SAP functions. The compliance check finds all identities that have such invalid authorization combinations over the sum of all authorizations of their SAP user accounts.

Related topics

- Example of SAP roles or user accounts with invalid authorizations on page 20
- Example of identities with invalid SAP authorizations on page 26
- Basics of the authorization check on page 14
- Rule conditions for SAP functions on page 63

Example of SAP roles or user accounts with invalid authorizations

The policies on valid SAP authorizations have been changed. Now the new policies must be checked to see if existing authorizations comply. SAP roles and user accounts with invalid combinations of authorizations must be identified so that they can be modified to meet the new requirements.

An SAP function is created for each invalid authorization combination.

Table 2: Example of an authorization definition

SAP function	Function argument	Authorization objects	Field	Value
F-A	D1	S_TCODE	TCD	TR1,TR2
	D2	AO2	NAME	*
	D3	AO3	ACTVT	04
	D3	AO3	NAME	H_XYZ
	D4	AO3	ACTVT	04
	D4	AO3	NAME	R*
Condition:	`D1 AND D2 AND (D3 OR D4)`			
F-B	D1	S_TCODE	TCD	TR3
	D2	S_TCODE	TCD	TR4,TR5
	D3	AO4	ACTVT	02
	D3	AO4	NAME	G
	D4	AO4	ACTVT	02
	D4	AO4	NAME	*
Condition:	`(D1 AND D3) OR (D2 AND D4)`			

The following SAP roles are available:

Table 3: Defined SAP roles

SAP role	Authorization objects	Field	Value
R1	AO1	ACTVT	*
	AO1	NAME	*
	AO2	NAME	GEF*
	AO3	ACTVT	*
	AO3	NAME	H_XYZ
	S_TCODE	TCD	TR1
R2	AO2	NAME	*
	AO3	ACTVT	01, 02, 04
	AO3	NAME	R_ST
	S_TCODE	TCD	TR4
R3	AO3	ACTVT	04
	AO3	NAME	H_XYZ
	AO4	ACTVT	02, 03
	AO4	NAME	*
	S_TCODE	TCD	TR6
R4	AO4	ACTVT	02
	AO4	NAME	*
	S_TCODE	TCD	TR3

The composite role R5 is assigned the single roles R2 and R3.

Table 4: Defined composite role

SAP role	Authorization objects	Field	Value
R5 has the following authorizations via the single roles R2 and R2	AO2	NAME	*
	AO3	ACTVT	01, 02, 04
	AO3	NAME	R_ST
	S_TCODE	TCD	TR4
	AO3	ACTVT	04
	AO3	NAME	H_XYZ
	AO4	ACTVT	02, 03
	AO4	NAME	*
	S_TCODE	TCD	TR6

The following user accounts are available:

- User account AC1 with composite role R5
- User account AC2 with SAP roles R2 and R3

  AC2 then has the same authorizations as AC1.
- User account AC3 with the SAP role R2
- User account AC4 with the SAP role R3

The authorization check determines all SAP roles and user accounts that are assigned the authorization objects and values listed in the authorization definitions. These roles and user accounts match the SAP functions. The authorization check produces the following results if the **TargetSystem | SAPR3 | SAPRights | TestWithoutTCD** configuration parameter is set.

Table 5: Authorization check results

Roles and user accounts	SAP function F-A	SAP function F-B
R1	match The role fulfills the condition `D1 AND D2 AND D3`.	no match
R2	no match	no match
R3	no match	no match
R4	no match	match The role fulfills the condition `D1`

Roles and user accounts	SAP function F-A	SAP function F-B
		AND D3.
R5	no match	match The role fulfills the condition D2 AND D4.
AC1	no match	match The user account fulfills the condition D2 AND D4.
AC2	no match	match The user account fulfills the condition D2 AND D4.
AC3	no match	no match
AC4	no match	no match

The SAP roles R1 and R4 and the composite role R5 as well as the user accounts AC1 and AC2 so not comply with the new policies and must be adjusted.

The authorization check ignores the authorization object S_TCODE if the **TargetSystem | SAPR3 | SAPRights | TestWithoutTCD** is set. The following conditions therefore apply to the test:

- F-A: `D2 AND (D3 OR D4)`
- F-B: `D3 OR D4`

Table 6: Results of the authorization check if TestWithoutTCD is set

Roles and user accounts	SAP function F-A	SAP function F-B
R1	match The role fulfills the condition D2 AND D3.	no match
R2	match The role fulfills the condition D2 AND D4.	no match
R3	no match	match The role fulfills the condition D3 OR D4.
R4	no match	match The role fulfills the condition D3 OR D4.

Roles and user accounts	SAP function F-A	SAP function F-B
R5	match The role fulfills the condition D2 AND D3 as well as D2 AND D4.	match The role fulfills the condition D3 OR D4.
AC1	match The user account fulfills the condition D2 AND D3 as well as D2 AND D4.	match The user account fulfills the condition D3 OR D4.
AC2	match The user account fulfills the condition D2 AND D3 as well as D2 AND D4.	match The user account fulfills the condition D3 OR D4.
AC3	match The user account fulfills the condition D2 AND D4.	no match
AC4	no match	match The user account fulfills the condition D3 OR D4.

If the **TestWithoutTCD** configuration parameter is set for the authorization check, then all SAP roles and user accounts match at least one SAP function and therefore do not comply with the new policies.

Example using different technical profiles

The decisive factor for the authorization check is whether a profile with multiple technical profiles belongs to one SAP role. The authorization check determines whether all the function elements and their values defined for an authorization object occur in a technical profile. If the authorization object has different values in different technical profiles, the SAP function does not find the role. The following example shows the difference.

Table 7: Defined SAP roles with technical profiles

SAP role	Authorization objects	Field	Value	Technical profile
R6	AO1	ACTVT	02, 03	TP1
	AO1	NAME	*	
	S_TCODE	TCD	TR1, TR2	

SAP role	Authorization objects	Field	Value	Technical profile
R7	AO1	ACTVT	02	TP2
	AO1	NAME	*	
	S_TCODE	TCD	TR2	
	AO1	ACTVT	03	TP3
	AO1	NAME	Z*	
	S_TCODE	TCD	TR1	

Table 8: Authorization definition

SAP function	Function argument	Authorization objects	Field	Value
F-TP1	D1	S_TCODE	TCD	TR1
	D1	S_TCODE	TCD	TR2
	D2	AO1	NAME	*
	D3	AO1	ACTVT	02
	D3	AO1	ACTVT	03
Condition:	`(D1 AND D2) OR (D2 AND D3)`			

Results of the authorization check if the **TestWithoutTCD** configuration parameter is not set:

- The SAP role R6 matches the SAP function because the technical profile TP1 fulfills the condition.
- The SAP role R7 does not match the SAP function because **`S_TCODE`** with the value **`TR1`** and **`S_TCODE`** with the value **`TR2`** belong to different technical profiles. **`BO1`** with the value **`02`** and **`BO1`** with the value **`03`** also belong to different technical profiles.
- The role R7 is also found by making the following adjustment to the SAP function:

Table 9: Authorization definition

SAP function	Function argument	Authorization objects	Field	Value
F-TP2	D1	S_TCODE	TCD	TR1
	D2	S_TCODE	TCD	TR2
	D3	AO1	NAME	*
	D4	AO1	ACTVT	02
	D5	AO1	ACTVT	03
Condition:	`(D1 AND D2 AND D3) OR (D3 AND D4 AND D5)`			

Related topics

- Example of identities with invalid SAP authorizations on page 26
- Examples of SAP functions on page 19

Example of identities with invalid SAP authorizations

The aim is to check which identities have invalid authorizations.

a. Create a compliance rule that checks whether there are identities with SAP user accounts that match the SAP functions.

b. Create different SAP functions for authorizations that in combination are invalid. Create compliance rules that combine these SAP functions. The compliance check finds all identities that have such invalid authorization combinations over the sum of all authorizations of their SAP user accounts.

The following SAP roles are available:

Table 10: Defined SAP roles

SAP role	Authorization objects	Field	Value	Technical profile
R8	AO3	ACTVT	01, 02, 04	TP1
	AO3	NAME	R_ST	
	S_TCODE	TCD	TR4	TP2
R9	AO3	ACTVT	04	TP1
	AO3	NAME	H_XYZ	
	AO4	ACTVT	02, 03	TP2
	AO4	NAME	*	
	S_TCODE	TCD	TR6	TP3
R10 as a composite role with the single roles R8 and R9	AO3	ACTVT	01, 02, 04	TP1-R8
	AO3	NAME	R_ST	
	AO3	ACTVT	04	TP1-R9
	AO3	NAME	H_XYZ	

SAP role	Authorization objects	Field	Value	Technical profile
	AO4	ACTVT	02, 03	TP2-R9
	AO4	NAME	*	
	S_TCODE	TCD	TR4	TP2-R8
	S_TCODE	TCD	TR6	TP3-R9

The following user accounts and identities are available:

- User A with user account AC5 with the composite role R10
- User B with user account AC6 with the SAP roles R8 and R9
- User C with user account AC7 with the SAP role R8 and user account AC8 with the SAP role R9

An identity must not own the authorizations of R8 and R9 at the same time. A compliance rule CR-X finds all identities that match the SAP function F-C.

CR-X: The identity owns at least the SAP function F-C.

Table 11: Authorization definition for the SAP function F-C

SAP function	Function argument	Authorization objects	Field	Value
F-C	D1	S_TCODE	TCD	TR4
	D2	S_TCODE	TCD	TR6
	D3	AO3	ACTVT	01,02,04
	D3	AO3	NAME	*
	D4	AO4	ACTVT	02,03
	D4	AO4	NAME	*
Condition:	`D1 AND D2 AND D3 AND D4`			

Results of the authorization check if the **argetSystem | SAPR3 | SAPRights | TestWithoutTCD** is not set.

Table 12: Authorization check results

Roles and user accounts	SAP function F-C
R8	no match
R9	no match
R10	match

Roles and user accounts	SAP function F-C
AC5	match
AC6	match
AC7	no match
AC8	no match

Compliance check results:

- User A violates the rule because the user account AC5 matches the SAP function F-C.
- User B violates the rule because the user account AC6 matches the SAP function F-C.
- User C does not violate the rule because the user accounts AC7 and AC8 do not match the SAP function F-C.

Viewed individually, user accounts AC7 and AC8 have valid authorizations. Only by linking these user accounts to an identity does the combination of these authorizations become invalid.

Compliance rules can detect invalid authorization combinations on identities. For them to do this, the SAP functions must be so structured that the user accounts AC7 and AC8 match. In the compliance rule, these SAP functions are combined so that identities with both user accounts violate the rule.

Table 13: Other SAP functions

SAP function	Function argument	Authorization objects	Field	Value
F-D	D1	S_TCODE	TCD	TR4
	D2	AO3	ACTVT	01,02,04
	D2	AO3	NAME	*
Condition:	`D1 AND D2`			
F-E	D1	S_TCODE	TCD	TR6
	D2	AO3	ACTVT	04
	D2	AO3	NAME	*
	D3	AO4	ACTVT	02,03
	D3	AO4	NAME	*
Condition:	`D1 AND D2 AND D3`			

Table 14: Authorization check results

Roles and user accounts	SAP function F-D	SAP function F-E
R8	match	no match
R9	no match	match
R10	match	match
AC5	match	match
AC6	match	match
AC7	match	no match
AC8	no match	match

A compliance rule finds all identities that match both these SAP functions.

CR-Y: The identity owns at least the SAP function F-D AND the identity owns at least the at least the SAP function F-E.

Compliance check results:

- User A violates the rule because the user account AC5 matches both the SAP functions.
- User B violates the rule because the user account AC6 matches both the SAP functions.
- User C violates the rule because the user account AC7 matches the SAP function F-D and the user account AC8 matches the SAP function F-E.

This means that the compliance rule CR-Y can be used to determine all identities that are assigned the SAP roles R8 and R9 through their user accounts.

Related topics

- Example of SAP roles or user accounts with invalid authorizations on page 20
- Examples of SAP functions on page 19
- Rule conditions for SAP functions on page 63

Recommendations for setting up SAP functions

Requirements and policies within your company determine how SAP functions are set up, authorization definitions are created, and compliance rules are used. First consider what you want to achieve with the authorization check.

1. Determine every SAP role and profile with invalid combinations of authorizations.
    - To do this, create SAP functions that determine invalid authorization combinations. The authorization check identifies all SAP roles and user accounts where the sum total of their authorizations have this invalid combination of authorizations.
    - To find all identities that have access to such user accounts, create compliance rules for these SAP functions.
2. Find all identities that own invalid combinations of authorizations through their various SAP user accounts.
    - The single SAP roles and user accounts have valid authorizations. Only an identity having access to multiple user accounts causes invalid authorization combinations.
    - Create different SAP functions for authorizations that are valid on their own. It is the combination that makes these authorizations invalid, so only the combination of these SAP functions leads to a policy violations.
    - Create compliance rules that combine these SAP functions. Combine all SAP functions that together reveal invalid authorization combinations. The compliance check finds all identities that join such invalid authorization combinations across all their SAP user accounts.

> TIP: If you create SAP functions for both use cases, you can use function categories to group the function definitions. This makes it easier to select SAP functions in the rule editor and displays function definitions in the Manager better.

Related topics

- Examples of SAP functions on page 19
- Basics of the authorization check on page 14
- Rule conditions for SAP functions on page 63
- SAP function categories on page 54

4

Setting up SAP functions

Create function definitions, function instances, and variable sets for SAP functions. You can use an SAP function for different instances. To do this, use variables in the function definition. Fixed variable values are grouped in variable sets and used in the function instances.

A function definition contains the authorization definition as well as general main data. An authorization definition contains at least one authorization object. Each authorization object consists of at least one function element (activity or authorization field) with fixed values. These are given as single values or as upper and lower limits. Function elements can be listed more than once per authorization object.

If an authorization definition includes several authorization objects, use logical operators to determine how these authorization objects are linked. The function definition is used to save all the authorization object links as a condition. A function argument is generated for each authorization object to clearly identify the authorization objects in the condition. These function arguments are used to formulate the condition.

The following rules apply to function arguments:

- Each function argument is permitted for use in an authorization definition for just one authorization object.
- Each authorization object is assigned to exactly one function argument with its function elements and values.
- An authorization object can be used multiple times with different values within an authorization definition. An new function argument is created for each instance.
- Within a function definition, the names of the function arguments must be unique.
- The name pattern for function arguments is defined in the **TargetSystem | SAPR3 | SAPRights | AbilityNamePattern** configuration parameter. If necessary, adjust the value of the configuration parameter to suit your requirements.

The following rules apply to conditions:

- Permitted operators are `AND`, `OR`, and priority brackets `()`.
- Permitted commentary characters are `/* */` for multi-line comments and `--` for single line comments.

Use variables for the values in the authorization definition. This means you can use a function definition for different function instances. The variables are provided in variable sets.

Function instances specify the client that uses the function definition and the specific values that apply to the test. To do this, assign values to the variables in a function instance and define the client.

To set up an SAP function

1. Create a function definition.
   - (Optional) If necessary, assign a function category or functional area to the managers.
2. Create the authorization definition.
   - Consider the explanations for determining invalid authorizations.
   - Take the notes on authorization definitions into account.
   - (Optional) Use variables for the values or range limits.
3. (Optional) Provide a new name for the function arguments complying with the naming convention given in the **TargetSystem | SAPR3 | SAPRights | AbilityNamePattern** configuration parameter.
4. Check the condition in which the function arguments are logically linked.
5. (Optional) Assign mitigating controls to the function definition to be implemented when invalid authorizations are detected by the SAP function.
6. To be able to use the function definition for authorization checking, enable the working copy of this function definition.
7. Create at least one function instance for this function definition.

To find all the identities that match this SAP function through their SAP user accounts, apply the SAP function in compliance rules.

Detailed information about this topic

- Creating function definitions on page 33
- Base data for SAP functions on page 53
- Creating authorization definitions in the Authorization Editor on page 35
- Basics of the authorization check on page 14
- Notes on authorization definitions on page 36
- Using variables on page 39
- Assigning mitigating controls to SAP functions on page 51
- Enabling working copies on page 41
- Defining function instances on page 46
- Compliance rules for SAP functions on page 63

- Renaming function arguments on page 44
- Recommendations for setting up SAP functions on page 29

Creating function definitions

There is a working copy created for each new function definition. Enabling the working copy allows the function definition to be used productively. SAP authorizations are only checked on the basis of active function definitions.

To create a new function definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definitions** category.
2. Click + in the result list.
3. Enter the function definition main data.
4. Save the changes.

   This adds a working copy.
5. Select the **Authorization Editor** task and set up the authorization definition.
6. Select the **Enable working copy** task and confirm the security prompt with **Yes**.

   This adds an enabled function definition in the database. The working copy is retained and can be used to make changes later.

Related topics

- General main data of a function definition on page 33
- Creating authorization definitions in the Authorization Editor on page 35
- Enabling working copies on page 41

General main data of a function definition

Enter the following main data of a function category.

Table 15: Main data for a function definition

Property	Description
Function definition	Name of the SAP function.
Functional area	The SAP function is valid for this functional area.
Function category	Grouping criteria for the SAP function. To create a new function

Property	Description
	categories, click +. Enter the name and a description of the function category.
Manager/supervisor	Application role whose members are responsible for the function definition in terms of content. To create a new application role, click +. Enter the application role name and assign a parent application role.
Authorization object details	Spare text field for entering information about the authorization objects that are used in the function definitions.
Risk index	Defines the risk for the company if an SAP user account matches this SAP function. Use the slider to enter a value between `0` and `1`. `0`: No risk. `1`: Every SAP user account that matches the SAP function poses a problem. This field is only visible if the **QER | CalculateRiskIndex** configuration parameter is set.
Risk index (reduced)	Show the risk index taking mitigating controls into account. An SAP function’s risk index is reduced by the significance reduction of all mitigating controls assigned to it. The risk index (reduced) is calculated only for the active SAP function. This input field is only visible if the **QER | CalculateRiskIndex** configuration parameter is set and the active SAP function is shown. This field is not shown in working copies. The value is calculated by One Identity Manager and cannot be edited.
Severity code	Specifies what it means to the company or the assigned functional area when an SAP user matches this SAP function. Enter a value between `0` and `1`. `0`: Just for information `1`: Any SAP user account that matches the SAP function requires changes to the affected SAP authorizations.
Significance	Specifies a verbal description of the effects on the company or the assigned functional area if an SAP user account matches this SAP function. Select a value from the list.
Description	Text field for additional explanation.
Working copy	Specifies whether this is a working copy of the function definition.
Condition	Expression that defines how to logically combine the function

Property	Description
	arguments in the evaluation. Only the operators **AND** and **OR** as well as precedence brackets **()** are permitted. When an authorization definition is created, a condition is automatically generated. Check the condition and adjust it to your requirements.

For more information about risk assessment, see the *One Identity Manager Risk Assessment Administration Guide*.

Detailed information about this topic

- SAP function categories on page 54
- Maintaining SAP functions on page 56
- Mitigating controls for SAP functions on page 67
- Functional areas for SAP functions on page 55
- Combining authorization objects and function elements on page 16

Creating authorization definitions in the Authorization Editor

Use the Authorization Editor to set up the SAP function authorization definition. To do this, compile the authorization objects to check with the SAP function.

To compile an authorization definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
2. Select the function definition in the result list.
3. Select the **Authorization Editor** task.
4. Select one of the following tasks.
    - **1. Add via menu template**

      Select from which menu you want to select the menu items and the SAP system whose menu tree should be displayed. Then select a menu item from the hierarchy.

        - **Show SAP menu** : Specifies whether you can select menu items from the SAP menu of the SAP GUI.
        - **All other menus**: Specifies whether you can select menu items from all other SAP menus.

    - **System**: SAP system to be used to display the menu tree.
    - **Menu**: Menu hierarchy for selecting menu items.

      Transaction codes that are linked to a menu item are shown in brackets in the menu tree as additional information.

  This loads all authorization objects that can be called via the selected menu item or its submenu items.

- **2. Add using SAP application**

  Select the **Type of SAP application** and the **SAP application** whose authorization objects should be loaded into the Authorization Editor. You can define a **Filter** to list the limit the number of SAP applications available.

  This adds all the authorization object that are linked to the selected SAP application.

- **3. Add using existing function definition**

  Select an existing **Function definition** with an authorization definition to load in the Authorization Editor.

  Only enabled function definitions can be selected.

- **4. Add via authorization object**

  Select an **Authorization object** to load into the Authorization Editor. You can define a **Filter** to list the limit the number of authorization objects available.

5. Define properties of the individual function elements in the Authorization Editor.
6. Save the changes.
7. Select **Change main data**.
8. Check the **Condition** and adjust it to your requirements.
9. Save the changes.

Detailed information about this topic

- Authorization definition properties and their values on page 37
- Notes on authorization definitions on page 36
- Using variables on page 39

Notes on authorization definitions

Take the following advice into account when you create an authorization definition in the authorization editor.

- To add an additional function element to an authorization object, click **+** next to the authorization object. Select whether you want to insert an activity or an authorization field.

- To add an additional value for an authorization field to an authorization object, click **C** next to the authorization field.
- You can enter more than one activity value by ORing (`OR`) them together. Delimit individual values with commas.
- The same authorization object added multiple times to an authorization definition if the function elements have different values.
- All rows in the function definition that belong to the same function argument are ANDed together (`AND`).

Related topics

- Creating authorization definitions in the Authorization Editor on page 35
- Basics of the authorization check on page 14
- Authorization definition properties and their values on page 37

Authorization definition properties and their values

The functionality of the Authorization Editor is based on the SAPGUI Authorization Editor. The columns in the Authorization Editor have the following meaning.

Table 16: Properties of an authorization definition

Property	Description
Function definition/ Authorization object class /Authorization object / Function element	Function definition hierarchy. The authorization objects and function elements are mapped in a hierarchical structure.
Processing status	Processing status of hierarchy objects. 🟠: No value is specified for the function element. 🟢: A value is specified for the function element.
Add	Click **+**, to add more objects to the authorization definition. This adds a sub object. Click **C**, to copy the function element.
Remove	Click **-**, to remove objects from the authorization definition.
Description	Object description.
Any	Click *****, to define the value of a function element as * (any value).
Value / lower limit	Values permitted for the function element. For example, you can

Property	Description
	limit SAP authorizations to specific SAP groups. When you specify a range, enter the lower limit here. Values can be added as variables. System variables can also be used. Wildcards can be used in the values. For more information, see Syntax examples for values on page 38.
Upper scope limit	Upper limit for the range of a function element Values can be added as variables. Values combined with **,** and ***** are not permitted. If **value / lower range limit** contains values combined with **,** or *****, no upper range limit can be entered.
Function argument	Name of the function argument for the authorization object. The name is formatted automatically according to the naming convention given in the **TargetSystem | SAPR3 | SAPRights | AbilityNamePattern** configuration parameter. You can change it manually.

Table 17: Syntax examples for values

Syntax (example)	SAP authorization is tested for	Input value examples
*	Any value Only use as a single value. You cannot specify an upper scope limit.	ab or 1234
Any string (from)	Exact given value	abc
[*]	The value *****	*
String[*] (abc [*])	Values that contain exactly this string and *****.	from*
String* (abc[*])	Values beginning with the given string and ending with any string Only use as a single value. You cannot specify an upper scope limit.	abcd or ab*
OR link (01,02,78)	One of the values contained in the list Do not use OR combinations for the upper range limit. Only use as a single value. You cannot specify an upper scope limit.	01 or 02 or 78

Syntax (example)	SAP authorization is tested for	Input value examples
Variable ($Var$)	Value stored in the variable	
System variable ($var)	Value stored in the system variable	

To edit the properties of a function element

- Double-click on a function element in the Authorization Editor.

  You can edit the description of the function element and the upper and lower limits.

Table 18: Function element properties

Property	Description
Type	Specifies whether the selected function element is an activity or a authorization field.
Name	Name of the function element.
Lower limit, upper limit	Values permitted for the function element. When you specify a range, enter a lower and an upper limit. Values can be added as variables. Click ▣ to select variables from the existing variable sets.
Description	Detailed description of the function elements.

Detailed information about this topic

- Notes on authorization definitions on page 36
- Using variables on page 39
- Creating and editing variable sets for authorization definitions on page 48
- Creating authorization definitions in the Authorization Editor on page 35
- Renaming function arguments on page 44

Using variables

You can set fixed values for function elements in authorization definitions. Otherwise, you can implement variables to use a function definition for different function instances. For this, the following is valid:

- Variable name
  - Begins with a letter
  - Only contains letters, numbers, and underscore

- Is enclosed in $ signs

Example: Var_01

NOTE: Variable names cannot begin with system variable names.

- Value

Syntax (example)	SAP authorization is tested for	Input value examples
*	Any value Only use as a single value. You cannot specify an upper scope limit.	ab or 1234
Any string (from)	Exact given value	abc
[*]	The value *****	*
String[*] (abc [*])	Values that contain exactly this string and *****.	from*
String* (abc [*])	Values beginning with the given string and ending with any string Only use as a single value. You cannot specify an upper scope limit.	abcd or ab*
OR link (01,02,78)	One of the values contained in the list Do not use OR combinations for the upper range limit. Only use as a single value. You cannot specify an upper scope limit.	01 or 02 or 78

You can also use system variables as well as self-defined variables in the authorization definition. System variables have the following syntax: `${character}+` (example: $AUFART).

Variables must be uniquely identifiable by the authorization check. Therefore, names of self-defined variables may not match system variables or begin with system variable name.

Related topics

- Creating authorization definitions in the Authorization Editor on page 35
- Main data for a variable set on page 49

Enabling working copies

SAP authorizations are only checked on the basis of active SAP functions. When you enable a new working copy, it adds an active function definition. Changes to an existing working copy are accepted by enabling the active function definition.

To transfer changes from a working copy to a function definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
2. Select the function definition in the result list.
3. Select the **Enable working copy** task.
4. Confirm the security prompt with **OK**.

Related topics

- Creating working copies on page 43
- Creating function definitions on page 33

Editing function definitions

A working copy is added to the database for every function definition. You can edit the working copies to change the function definitions. Enabling the working copy allows the function definition to be used productively. SAP authorizations are only checked on the basis of active function definitions.

> NOTE: One Identity Manager users with the **Identity & Access Governance | Identity Audit | Maintain SAP functions** application role can edit existing working copies if they are entered as the manager in the main data.

To edit an existing function definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
    a. Select a working copy in the result list.
    b. Select the **Change main data** task.

   - OR -

   In the Manager, select the **Identity Audit > SAP functions > Function definitions** category.
    a. Select the function definition in the result list.
    b. Select the **Create working copy** task.

The data from the existing working copy are overwritten with the data from the active function definition, after prompting. The working copy is opened and can be edited.

2. Edit the working copy's main data.
3. Save the changes.
4. Select the **Enable working copy** task and confirm the security prompt with **Yes**.

   The changes to the working copy are transferred to the active function definition.

Related topics

- Creating working copies on page 43
- General main data of a function definition on page 33
- Authorization definition properties and their values on page 37
- Enabling working copies on page 41

Authorization overview

Function elements are displayed in a flat structure in the authorization overview.

To display an overview of all function elements for an active function definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definitions** category.
2. Select the function definition in the result list.
3. Select the **Authorization overview** task.

To display an overview of all function elements for a working copy

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
2. Select the function definition in the result list.
3. Select the **Authorization overview** task.

   You can edit all the object properties here.

Related topics

- Creating authorization definitions in the Authorization Editor on page 35

Creating working copies

To modify an existing function definition, you require a working copy of the function definition. You can create a working copy from the active function definition. After confirming the prompt, the data of an existing working copy is overwritten with the data from the active function definition.

To create a working copy

1. In the Manager, select the **Identity Audit > SAP functions > Function definitions** category.
2. Select the function definition in the result list.
3. Select the **Create working copy** task.
4. Confirm the security prompt with **Yes**.

Related topics

- Enabling working copies on page 41
- Editing function definitions on page 41

Exporting individual function definitions

To transfer SAP functions from a development environment to a production environment, for example, you can export function definitions to CSV files. These CSV files can be imported into other databases.

To export the function definition to a CSV file

1. In the Manager, select the **Identity Audit > SAP functions > Function definitions** category.
2. Select the function definition in the result list.
3. Select the **Change main data** task.
4. Select the **Export** task.
5. Specify the file name and storage location for the CSV file.
6. Click **Save**.

To export the function definition of a working copy to a CSV file

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
2. Select the function definition in the result list.
3. Select the **Change main data** task.

4. Select the **Export** task.
5. Specify the file name and storage location for the CSV file.
6. Click **Save**.

The following properties are exported:

Table 19: Exported main data of a function definition

Property	Data field in the CSV file.
Name of the function definition (SAPFunction.Ident_SAPFunction)	Function
Assigned function category (SAPFunctionCategory.Ident_SAPFunctionCategory)	Process
Description (SAPFunction.Description)	Function Description
Effect (SAPFunction.SignificancyClass)	Risk Level
Authorization object (SAPFunctionDetail.Ident_SAPAuthObject)	Object
Authorization fields (SAPFunctionDetail.ElementName)	Field
Description of the authorization fields (SAPFunctionDetail.Description)	Field Description
Value/Lower limit (SAPFunctionDetail.LowerLimit)	Value From
Upper limit (SAPFunctionDetail.UpperLimit)	Value To
Function argument (SACAbility.AbilityName)	Ability Name
Condition (SAPFunction.ConditionString)	Condition String

The import status (State) is included with each data record in the CSV file as additional information. The import status is set to **1** by default on export. This data is evaluated when function definitions are imported.

Related topics

- Importing function definitions on page 61
- Exporting function definitions on page 57

Renaming function arguments

All function arguments within a SAP function must have a unique name. To ensure this, the name is formed by default from a character string and a consecutive number. The name pattern is defined in the **TargetSystem | SAPR3 | SAPRights | AbilityNamePattern** configuration parameter. You can adjust the value of the configuration parameter to suit your requirements if necessary. When you create an authorization definition it automatically names the function arguments.

You can change the names of the function arguments of an authorization definition manually at any time or reapply the name pattern. For example, use this functionality to assign uniform names for function arguments if the SAP functions have been imported or migrated from other or older databases.

To rename the function arguments of an authorization definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
2. Select the function definition in the result list.
3. Select the **Authorization Editor** task.
4. Select the **Rename function arguments** task.
5. Confirm the security prompt with **Yes**.
6. Save the changes.

Related topics

- Configuration parameters for SAP functions on page 71
- Authorization definition properties and their values on page 37

Function definition overview

You can display the most important information about a function definition on the overview form.

To obtain an overview of a function definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definitions** category.
2. Select the function definition in the result list.
3. Select the **Function definition** task.

To obtain an overview of a working copy

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
2. Select the function definition in the result list.
3. Select the **Function definition** task.

Related topics

- Setting up SAP functions on page 31

Defining function instances

One and the same function definition can be used for different concrete instances. Specify an SAP client in the function instances to which to apply the SAP function. In addition, the variables that are assigned to the authorization fields are given specific values. Function instances can only be created for SAP functions that are enabled.

To create a function instance

1. In the Manager, select the **Identity Audit > SAP functions > Function instances** category.
2. Click + in the result list.
3. Edit the function instance's main data.
4. Save the changes.

To edit a function instance

1. In the Manager, select the **Identity Audit > SAP functions > Function instances** category.
2. In the result list, select a function instance and run the **Change main data** task.
3. Edit the function instance's main data.
4. Save the changes.

> NOTE: One Identity Manager users with the **Identity & Access Governance | Identity Audit | Maintain SAP functions** application role can create and edit function instances for the SAP functions if they are listed as the manager.

Detailed information about this topic

- Main data for function instances on page 46
- Checking field variable definitions on page 47
- Function instance overview on page 48

Main data for function instances

Enter the following main data of a function instance.

Table 20: Function instance properties

Property	Description
Function definition	The function instance is created for this function definition.
Client	SAP client to which the SAP function should be applied.
Variable set	Variable set with functions defined, which are used in the function definition. The variable set and the function instance must be assigned to the same SAP client.
Manager/supervisor	Application role whose members are responsible for the function instance and variable sets in terms of content. To create a new application role, click . Enter the application role name and assign a parent application role.
Display name	Function instance display name. This is formatted from the function definition name, the assigned client and variable set.
Description	Text field for additional explanation. A new function instance takes the description from the function definition.
Function Instance Elements	Displays authorization objects and function elements of the SAP function with specified values that are determined from the assigned variable set. Changes to the variables or variable set are displayed as soon as the DBQueue Processor has processed the corresponding authorization tasks.

Related topics

- Creating and editing variable sets for authorization definitions on page 48
- Maintaining SAP functions on page 56
- Checking field variable definitions on page 47

Checking field variable definitions

Before you use function instances in compliance rules, check whether all variable which are used in the function definition are defined in the variable set. If there is no function definition or variable set assigned to the function instance, the check-in fails with an error message. Variables that are not defined in the associated variable set are listed in the error message.

To check variable definitions

1. In the Manager, select the **Identity Audit > SAP functions > Function instances** category.
2. Select the function instance in the result list.

3. Select the **Change main data** task.
4. Select the **Check variable definitions** task.

Related topics

- Main data for function instances on page 46
- Main data for a variable set on page 49

Function instance overview

You can display the most important information about a function instance on the overview form.

To obtain an overview of a function instance

1. In the Manager, select the **Identity Audit > SAP functions > Function instances** category.
2. Select the function instance in the result list.
3. Select the **Function instance** task.

Related topics

- Defining function instances on page 46

Creating and editing variable sets for authorization definitions

Use variable sets to group variables together that are used in an authorization definition and give then fixed values.

To create a variable set

1. In the Manager, select the **Identity Audit > SAP Functions > Variable sets** category.
2. Click + in the result list.
3. Edit the variable set's main data.
4. Save the changes.

To edit a variable set

1. In the Manager, select the **Identity Audit > SAP Functions > Variable sets** category.
2. In the result list, select the variable set and run the **Change main data** task.
3. Edit the variable set's main data.
4. Save the changes.

Related topics

- Main data for a variable set on page 49
- Adding variables used in SAP functions on page 50
- Creating authorization definitions in the Authorization Editor on page 35
- Variable set overview on page 51
- Copying variable sets on page 51

Main data for a variable set

Enter the following main data of variable sets.

Table 21: Main data for a variable set

Property	Description
Variable set	Unique variable set identifier.
Client	Valid SAP client for the variable set.
Department	Relevant department for the variable set.
Functional area	Functional area relevant to the variable set.
Description	Text field for additional explanation.
SAP field variables	List of defined variables.

To create a field variable in the variable set

- Click **Add** and enter the following properties.
  - **Variable**: Name of the variable in `${alphanum}+$` notation.

    > NOTE: Variable names cannot begin with system variable names. Variable sets with variables like this cannot be saved.

  - **Value**: Fixed values for the variable to be copied to the function instance.
  - **Description**: Text field for additional explanation.

- **Authorization object**: Reference to the authorization object to use in the variable in.

There is help for your selected on the form. On the form, there is help available for selecting authorization fields for an authorization object to be used for defining variables.

To delete a field variable from the variable set

1. Select a line in the list of field variables.
2. Click **Remove selected**.

> TIP: You can add variable sets without defining variables. Use these variables set for function definitions that do not have variables entered as values. Checking the field variables then does not produce an error message.

Related topics

- Using variables on page 39
- Adding variables used in SAP functions on page 50

Adding variables used in SAP functions

Variables used in authorization definitions of SAP functions can be added to variable sets.

To transfer variables to a variable set

1. Select the **Identity Audit > SAP Functions > Variable sets** category.
2. Select the variable set in the result list.
3. Select the **Change main data** task.
4. Select the **Apply chosen variables** task.
5. Mark all function definitions or working copies from which you want to copy the variables into the variable set.

   Multi-select is possible.
6. Click **OK** to transfer the variables.

   All variables from the selected function definitions are add to the list of field variables.
7. Edit the variables' properties.
8. Save the changes.

Related topics

- Main data for a variable set on page 49
- Using variables on page 39

Copying variable sets

To copy a variable set

1. In the Manager, select the **Identity Audit > SAP Functions > Variable sets** category.
2. In the result list, select the variable set and run the **Change main data** task.
3. Select the **Copy variable set** task.
4. Click **Yes** to immediately edit the copy's main data.
5. Edit the copy's main data.
6. Save the changes.

Related topics

- Main data for a variable set on page 49

Variable set overview

You can display the most important information about a variable set on the overview form.

To obtain an overview of a variable set

1. In the Manager, select the **Identity Audit > SAP Functions > Variable sets** category.
2. Select the variable set in the result list.
3. Select the **Variable set overview** task.

Related topics

- Creating and editing variable sets for authorization definitions on page 48

Assigning mitigating controls to SAP functions

Mitigating controls can be stored with SAP functions. These reduce the effects on the company when SAP user accounts match with SAP functions. At the same time, you specify how to deal with SAP user accounts, SAP roles, or SAP profiles that match the SAP function. For example, changing a user assignment to an SAP role in the SAP system can be used as a mitigating control for an SAP function.

Mitigating controls can also be used as controlling measures for compliance rules. Mitigating controls assigned to the SAP functions for testing are automatically transferred into compliance rules about SAP functions.

Prerequisites:

- Enabled compliance rules are assigned to a functional area and a department.
- The SAP functions for testing are assigned to the same functional area and then associated variable set of the same department.

To edit mitigating controls

- In the Designer, enable the **QER | CalculateRiskIndex** configuration parameter.

Related topics

- Assigning mitigating controls to a function definition on page 52
- Creating mitigating controls for SAP functions on page 53
- Mitigating controls for SAP functions on page 67

Assigning mitigating controls to a function definition

Assign mitigating controls to working copies of function definitions. Enabling the working copy transfers them to the active function definition.

To assign mitigating controls to a function definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
2. Select the working copy in the result list.
3. Select the **Assign mitigating controls** task.

   In the **Add assignments** pane, assign the mitigating controls.

   > TIP: In the **Remove assignments** pane, you can remove mitigating control assignments.
   >
   > ***To remove an assignment***
   >
   > - Select the mitigating control and double-click ✓.
4. Save the changes.
5. Enable the working copy.

Related topics

- Assigning mitigating controls to SAP functions on page 51

Creating mitigating controls for SAP functions

To create a mitigating control for SAP functions

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
2. Select a working copy in the result list.
3. Select the **Assign mitigating controls** task.
4. Select the **Create mitigating controls** task.
5. Enter the main data of the mitigating control.
6. Save the changes.
7. Select the **Assign function definitions** task.
8. In the **Add assignments** pane, double-click the function definitions you want to assign.
9. Save the changes.

Related topics

- Entering main data for mitigating controls on page 68
- Assigning mitigating controls to SAP functions on page 51

Base data for SAP functions

The following base data is relevant for SAP Functions:

- SAP function categories

  Use SAP function categories to group SAP functions by specific criteria.

  For more information, see SAP function categories on page 54.
- Functional areas

  Functional areas can be used as an additional group characteristic for SAP functions. Furthermore, you can use functional areas to analyze rule violations in context of Identity Audit for different SAP functions and to approve requests in the IT Shop or attestation cases by peer group analysis.

  For more information, see Functional areas for SAP functions on page 55.
- Maintaining SAP functions

  SAP functions can be assigned identities that manage the SAP functions and therefore can edit the working copies.

For more information, see Maintaining SAP functions on page 56.

SAP function categories

Use function categories to group SAP functions by specific criteria. For example, in a function category you can group together all SAP functions for a specific use case.

To create a function category

1. In the Manager, select the **Identity Audit > Basic configuration data > SAP function categories** category.
2. Click + in the result list.
3. Edit the function category's main data.
4. Save the changes.

To edit a function category

1. In the Manager, select the **Identity Audit > Basic configuration data > SAP function categories** category.
2. In the result list, select a function category and run the **Change main data** task.
3. Edit the function category's main data.
4. Save the changes.

Enter the following main data of a function category.

Table 22: SAP function category properties

Property	Description
Category	The function category’s name.
Parent category	Parent category for organizing function categories hierarchically.
Description	Text field for additional explanation.

To assign a function category to a function definition

1. In the Manager, select the **Identity Audit > SAP functions > Function definition working copies** category.
   a. Select a working copy in the result list.
   b. Select the **Change main data** task.
2. In the **Function category** drop-down, select a function category.
3. Save the changes.

Related topics

- General main data of a function definition on page 33
- Recommendations for setting up SAP functions on page 29

Functional areas for SAP functions

You can use functional areas to analyze rule violations in context of Identity Audit for different SAP functions. You can enter criteria that provide information about risks from rule violations for functional areas and SAP functions.

To analyze rule checks for different areas of your company in the context of identity audit, you can set up functional areas. Assign functional areas to hierarchical roles, compliance rules, SAP functions, and service items. To assess the risks, specify how many rule violations are permitted in a functional area or role. You can enter separate assessment criteria for each role, such as a risk index or transparency index.

Moreover, functional areas can be replaced by peer group analysis during request approvals or attestation cases.

Example: Use of functional areas

To assess the risk of rule violations for cost centers. Proceed as follows:

1. Set up functional areas.
2. Assign cost centers to the functional areas.
3. Define assessment criteria for the cost centers.
4. Specify the number of rule violations allowed for the functional area.
5. Assign the functional areas to the SAP functions or compliance rules that are relevant for the assessment.
6. Use the One Identity Manager report function to create a report that prepares the result of rule checking for the functional area by any criteria.

To create or edit a functional area

1. In the Manager, select the **Identity Audit > Basic configuration data > Functional areas** category.
2. In the result list, select a function area and run the **Change main data** task.

   - OR -

   Click + in the result list.
3. Edit the function area main data.
4. Save the changes.

Enter the following data for a functional area.

Table 23: Functional area properties

Property	Description
Functional area	Description of the functional area
Parent Functional area	Parent functional area in a hierarchy. Select a parent functional area from the list for organizing your functional areas hierarchically.
Max. number of rule violations	List of rule violation valid for this functional area. This value can be evaluated during the rule check.
Description	Text field for additional explanation.

Mitigating controls assigned to the function definitions to be tested are automatically copied to rules about SAP functions. Conditions:

- Active rules are assigned to a functional area and a department.
- The function definitions to be tested are assigned to the same functional area and to the variable set associated with the same department.

Related topics

- Mitigating controls for SAP functions on page 67
- General main data of a function definition on page 33

Maintaining SAP functions

You can assign SAP functions to identities that are responsible for the content of those SAP functions. To do this, assign the an application for maintaining SAP functions to an application role. Assign to this application role, the identities that are authorized to enable and edit working copies of this function definition and can define function instances.

A default application role exists for maintaining One Identity Manager functions in SAP. Create more application roles if required. For more information about implementing and editing application roles, see the *One Identity Manager Authorization and Authentication Guide*.

Table 24: Default application roles for maintaining SAP functions

User	Tasks
Responsible for maintaining	Those responsible for maintaining the SAP functions must be assigned to the **Identity & Access Governance | Identity Audit | Maintenance SAP Functions** application role or a child application role.

User	Tasks
SAP functions.	Users with this application role: • Are responsible for SAP function contents. • Edit working copies of function definitions for which they are responsible. • Define function instances and variables sets for SAP functions. • Assign mitigating controls.

To add identities to the default application role for maintaining SAP functions

1. In the Manager, select the **Identity Audit > Basic configuration data > Maintain SAP functions** category.
2. Select the **Assign identities** task.
3. In the **Add assignments** pane, add identities.

   > TIP: In the **Remove assignments** pane, you can remove assigned identities.
   >
   > ***To remove an assignment***
   >
   > - Select the identity and double-click ✓.

4. Save the changes.

Related topics

- General main data of a function definition on page 33

Exporting function definitions

To transfer SAP functions from a development environment to a production environment, for example, you can export function definitions to CSV files. These CSV files can be imported into other databases.

Export all function definitions to a single CSV file using a plugin.

To export all function definitions to a CSV file

1. In the Manager, select the **Identity Audit** category.
2. Select the **Plugins > Export all SAP function definitions** menu item.
3. To only export working copies, click **Yes**.

   - OR -

   To only export enabled SAP functions, click **No**.
4. Specify the file name and storage location for the CSV file.

5. Click **Save**.

   All function definitions are written to file in sequence.

The following properties are exported:

Table 25: Exported main data of a function definition

Property	Data field in the CSV file.
Name of the function definition (SAPFunction.Ident_SAPFunction)	Function
Assigned function category (SAPFunctionCategory.Ident_SAPFunctionCategory)	Process
Description (SAPFunction.Description)	Function Description
Effect (SAPFunction.SignificancyClass)	Risk Level
Authorization object (SAPFunctionDetail.Ident_SAPAuthObject)	Object
Authorization fields (SAPFunctionDetail.ElementName)	Field
Description of the authorization fields (SAPFunctionDetail.Description)	Field Description
Value/Lower limit (SAPFunctionDetail.LowerLimit)	Value From
Upper limit (SAPFunctionDetail.UpperLimit)	Value To
Function argument (SACAbility.AbilityName)	Ability Name
Condition (SAPFunction.ConditionString)	Condition String

The import status (State) is included with each data record in the CSV file as additional information. The import status is set to **1** by default on export. This data is evaluated when function definitions are imported.

> NOTE: SAP function managers can only export those function definitions for which they are responsible, as entered in the main data.

Related topics

- Importing function definitions on page 58
- Exporting individual function definitions on page 43

Importing function definitions

To transfer SAP functions from a development environment to a production environment, for example, you can export function definitions to CSV files. These CSV files can be imported into other databases.

When importing SAP functions from an existing CSV file, the function definitions contained in the CSV file are transferred to the database as working copies.

Related topics

- Requirements and notes for importing function definitions on page 59
- Importing function definitions from versions older than 9.3 on page 60
- Importing function definitions on page 61
- Exporting function definitions on page 57
- Exporting individual function definitions on page 43

Requirements and notes for importing function definitions

The following data fields must be in the CSV file so that function definitions can be imported.

Table 26: Data fields for importing function definitions

Data field in the CSV file. **(header)**	**Object properties in One Identity Manager**
Function	Function definition (`SAPFunction.Ident_SAPFunction`)
Object	Authorization object (`SAPFunctionDetail.Ident_SAPAuthObject`)
Field	Authorization field (`SAPFunctionDetail.ElementName`)
Value From	Value/Lower limit (`SAPFunctionDetail.LowerLimit`)
Value To	Upper limit (`SAPFunctionDetail.UpperLimit`)
State	No equivalent. The import status controls which data records are imported into One Identity Manager. **1**: Import
Ability Name (optional)	Function argument (`SACAbility.AbilityName`)
Condition String (optional)	Condition (`SAPFunction.ConditionString`)
Process (optional)	Function category (`SAPFunctionCategory.Ident_`

Data field in the CSV file. (header)	Object properties in One Identity Manager
	SAPFunctionCategory)
Function Description (optional)	Description of the function definition. (SAPFunction.Description)
Risk Level (optional)	Effect (SAPFunction.SignificancyClass) Possible values: • 0|<empty>|none • 1|verylow|very low • 2|low • 3|medium • 4|high • 5|veryhigh|very high • 6|critical
Field description (optional)	Description of the authorization fields and authorization objects. (SAPFunctionDetail.Description)

NOTE:

- The order of the data fields is arbitrary.
- All required data fields must be defined in the header and must be present in the data sets.
- Mark data fields without values with two sequential delimiters.
- Data sets with empty mandatory fields are not imported.

Related topics

- Importing function definitions on page 58
- Importing function definitions on page 61

Importing function definitions from versions older than 9.3

The design of authorization definitions was fundamentally changed with One Identity Manager 9.3. Importing function definitions from versions older than 9.3 also updates the authorization definitions. The **TargetSystem | SAPR3 | SAPRights | TestWithoutTCD** configuration parameter setting is taken into account.

- Create a function argument for each authorization object in a function definition. The properties of the authorization objects determine the names of the function arguments.

  You can rename the function arguments as required in the Manager.
- There is a generated condition.
  - The configuration parameter is not set:

    In the condition, group all the function arguments that belong to an SAP application inside a pair of brackets and AND them with each other. All brackets are OR-ed together.
  - The configuration parameter is set:

    All function arguments AND-ed in the condition.

After importing older function definitions, check whether the authorization definition and the generated condition meet your requirements.

Related topics

- Importing function definitions on page 58
- Ignoring SAP applications on page 18
- Renaming function arguments on page 44

Importing function definitions

Import CSV files with data from function definitions into the One Identity Manager database.

To import function definitions

1. In the Manager, select the **Identity Audit** category.
2. Select the **Plugins > Import SAP function definitions** menu item.
3. Select the CSV file you want to import and click **Open**.
4. Confirm the security prompt with **Yes**.

   The functions definitions are transferred to the database as working copies. If there is already a working copy with the same name in the database, it is overwritten by the import.
5. Open the working copy and check whether the authorization definition and the condition meet your requirements.

Related topics

- Requirements and notes for importing function definitions on page 59
- Exporting function definitions on page 57

- Exporting individual function definitions on page 43
- Importing function definitions on page 58
- Editing function definitions on page 41

5

Compliance rules for SAP functions

In addition to the permissions assigned to an identity in an SAP R/3 system on the basis of its user accounts, group memberships, and role memberships, you can also check which write permissions are in effect using compliance rules. Effective write permissions are tested through SAP functions. To do this, SAP functions are added to rule conditions. By linking SAP user accounts to identities, combinations of SAP authorizations that an identity obtains through different SAP user accounts can be checked.

The validity period of role assignments is taken into account in the rule check.

For more information about compliance rules, see the *One Identity Manager Compliance Rules Administration Guide*.

Related topics

- Rule conditions for SAP functions on page 63
- Mitigating controls for compliance rules with SAP functions on page 65
- More rule violation reports on page 65

Rule conditions for SAP functions

Determine whether identities have invalid combinations of authorizations in an SAP R/3 system by including SAP functions in the rule conditions of compliance rules.

- To find identities that have invalid authorizations across **multiple** user accounts, create different SAP functions. Create a separate rule block for each SAP function in the rule condition.
- To find identities that have invalid authorizations through **one** user account, create just one rule block in the rule condition.

To define new rules for SAP functions

1. In the Manager, select the **Identity Audit > Rules** category.
2. Click 🗋 in the result list.

3. Enter the main data of the rule.
4. Set the **Rule for cyclical testing and risk analysis in IT Shop** option.
5. Limit the affected permissions with the **at least one function** option and select the SAP functions to test.
    a. If you have selected more than one SAP functions, under **number of entitlements assigned**, specify how many SAP functions must be matched to violate the rule.
    b. If SAP authorizations in combination result in a rule violation, enter a rule block for each SAP function.
6. Save the changes.

    This adds a working copy.
7. Select the **Enable working copy** task and confirm the security prompt with **Yes**.
8. To enable the original rule, click **Yes**.

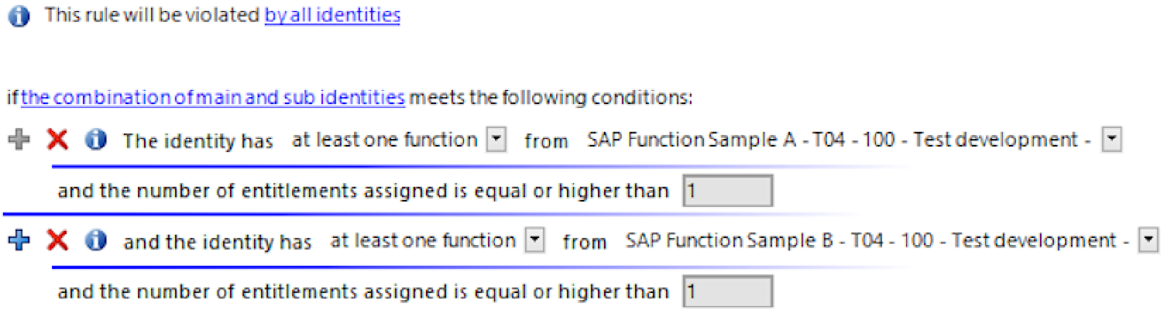
    This adds an enabled rule to the database.

    If you do not want the original rule to be enabled immediately, click **No**.

    This add a disabled rule to the database.

    The working copy is retained and can be used to make changes later.

Figure 6: Condition for SAP functions



This rule will be violated by all identities

if the combination of main and sub identities meets the following conditions:

The identity has at least one function from SAP Function Sample A - T04 - 100 - Test development -

and the number of entitlements assigned is equal or higher than 1

and the identity has at least one function from SAP Function Sample B - T04 - 100 - Test development -

and the number of entitlements assigned is equal or higher than 1

When One Identity Manager tests rules, it finds all the identities whose assigned SAP users match the SAP functions that are given in the rule. An SAP user also matches an SAP function when:

- A reference user matches the SAP function

  - AND -
- The SAP user account is assigned this reference user

For more information about creating rule conditions, see the *One Identity Manager Compliance Rules Administration Guide*.

Related topics

- Example of identities with invalid SAP authorizations on page 26
- Recommendations for setting up SAP functions on page 29

Mitigating controls for compliance rules with SAP functions

Mitigating controls assigned to the function definitions to be tested are automatically copied to rules about SAP functions. Conditions:

- Active rules are assigned to a functional area and a department.
- The function definitions to be tested are assigned to the same functional area and to the variable set associated with the same department.

Related topics

- Assigning mitigating controls to SAP functions on page 51

More rule violation reports

One Identity Manager makes various reports available containing information about the selected base object and its relations to other One Identity Manager database objects. Additional reports can be created for enabled compliance rules for SAP functions.

Table 27: Reports about rule violations with SAP functions

Report	Description
Rule violations with SAP applications	This report groups together all rule violations for the selected rule. It supplies results for rules that verify SAP functions. All function instances are listed with their SAP applications for each identity through which they violated the rule. SAP profiles and their authorization objects that match the SAP function are displayed for each SAP function.
Rule violations with SAP roles	This report groups together all rule violations for the selected rule. It supplies results for rules that verify SAP functions. The SAP groups, SAP roles, and SAP profiles of each identity that caused the identity to violate the rule are listed along with their authorization objects.

Report	Description
SAP roles and profiles with rule violations	The report shows all SAP roles and profiles that match SAP functions and thereby violate the selected rule.

Related topics

- Compliance rules for SAP functions on page 63

6

Mitigating controls for SAP functions

Violation of regulatory requirements can harbor different risks for companies. To evaluate these risks, you can apply risk indexes to SAP functions. These risk indexes provide information about the risk involved for the company if this particular SAP function is violated. Once the risks have been identified and evaluated, mitigating controls can be implemented.

Mitigating controls are independent on One Identity Manager’s functionality. They are not monitored through One Identity Manager.

Mitigating controls describe controls that are implemented if an SAP function was violated. The next calculation should not find any invalid authorizations for this SAP function once the controls have been applied.

To edit mitigating controls

- In the Designer, set the **QER | CalculateRiskIndex** configuration parameter and compile the database.

If you disable the configuration parameter at a later date, model components and scripts that are no longer required, are disabled. SQL procedures and triggers are still carried out. For more information about the behavior of preprocessor relevant configuration parameters and conditional compiling, see the *One Identity Manager Configuration Guide*.

For more information about mitigating controls, see the *One Identity Manager Risk Assessment Administration Guide*.

Detailed information about this topic

- Entering main data for mitigating controls on page 68
- Displaying mitigating controls overview for SAP functions on page 69
- Assigning function definitions to mitigating controls on page 68
- Calculating mitigating controls for SAP functions on page 69

Entering main data for mitigating controls

To create or edit mitigating controls

1. In the Manager, select the **Risk index functions > Mitigating controls** category.
2. Select a mitigating control in the result list and run the **Change main data** task.

   - OR -

   Click + in the result list.
3. Edit the mitigating control main data.
4. Save the changes.

Enter the following main data of mitigating controls.

Table 28: General main data of a mitigating control

Property	Description
Measure	Unique identifier for the mitigating control.
Significance reduction	When the mitigating control is implemented, this value is used to reduce the risk of denied attestation cases. Enter a number between `0` and `1`.
Description	Detailed description of the mitigating control.
Functional area	Functional area in which the mitigating control may be applied.
Department	Department in which the mitigating control may be applied.

Related topics

- Mitigating controls for SAP functions on page 67

Assigning function definitions to mitigating controls

Use this task to specify the function definitions for which a mitigating control is valid. You can only assign function definitions that are enabled on the assignment form.

To assign SAP function definitions to mitigating controls

1. In the Manager, select the **Risk index functions > Mitigating controls** category.
2. Select the mitigating control in the result list.

3. Select the **Assign function definitions** task.

   In the **Add assignments** pane, assign the function definitions.

   > TIP: In the **Remove assignments** pane, you can remove function definitions assignments.
   >
   > ***To remove an assignment***
   >
   > - Select the mitigating control and double-click ✓.

4. Save the changes.

Related topics

- Assigning mitigating controls to SAP functions on page 51
- Mitigating controls for SAP functions on page 67

Calculating mitigating controls for SAP functions

The reduction in significance of a mitigating control supplies the value by which the risk index of an SAP function is reduced when the control is implemented. One Identity Manager calculates a reduced risk index based on the risk index and the significance reduction. One Identity Manager supplies default functions for calculating reduced risk indexes. These functions cannot be edited with One Identity Manager tools.

The reduced risk index is calculated from the SAP function and the significance reduced sum of all assigned mitigating controls.

```
Risk index (reduced) = Risk index - sum significance reductions
```

If the significance reduction sum is greater than the risk index, the reduced risk index is set to `0`.

Related topics

- Mitigating controls for SAP functions on page 67

Displaying mitigating controls overview for SAP functions

You can display the most important information about a mitigating control on the overview form.

To obtain an overview of a mitigating control

1. In the Manager, select the **Risk Index Functions** category.
2. Select the **Mitigating controls** category.
3. Select the mitigating control in the result list.
4. Select **Mitigating control overview** category.

Related topics

- Mitigating controls for SAP functions on page 67

Appendix A

Configuration parameters for SAP functions

The following configuration parameters are additionally available in One Identity Manager after the module has been installed.

Table 29: Configuration parameters for the module

Configuration parameter	Description
TargetSystem | SAPR3 | SAPRights	Preprocessor relevant configuration parameter for controlling component parts for testing authorizations in SAP R/3 using SAP functions. If the parameter is set, the components are available. Changes to the parameter require recompiling the database. If you disable the configuration parameter at a later date, model components and scripts that are no longer required, are disabled. SQL procedures and triggers are still carried out. For more information about the behavior of preprocessor relevant configuration parameters and conditional compiling, see the *One Identity Manager Configuration Guide*.
TargetSystem | SAPR3 | SAPRights | AbilityNamePattern	Pattern for creating new names for function arguments. All function arguments within a SAP function must have a unique name. This pattern is used when new names are created for function arguments. **`{0}`** is replaced by a number. It is also possible to use formatting strings. Example: `{0:00}` - This adds leading zeros to the numeric part of the generated name.
TargetSystem | SAPR3 | SAPRights | TestWithoutTCD	Checks SAP authorizations without taking SAP applications into account. The configuration parameter will be deleted in a future version of One Identity Manager and can no longer be set in version 9.3.

The following configuration parameters are also required.

Table 30: Additional configuration parameters

Configuration parameter	Description
QER | CalculateRiskIndex	Preprocessor relevant configuration parameter controlling system components for calculating the risk index. Changes to the parameter require recompiling the database. If the parameter is enabled, values for the risk index can be entered and calculated. If you disable the configuration parameter at a later date, model components and scripts that are no longer required, are disabled. SQL procedures and triggers are still carried out. For more information about the behavior of preprocessor relevant configuration parameters and conditional compiling, see the *One Identity Manager Configuration Guide*.
QER | ComplianceCheck	Preprocessor relevant configuration parameter for controlling the database model components for checking the rule base. Changes to the parameter require recompiling the database. If the parameter is enabled, you can use the model components. If you disable the configuration parameter at a later date, model components and scripts that are no longer required, are disabled. SQL procedures and triggers are still carried out. For more information about the behavior of preprocessor relevant configuration parameters and conditional compiling, see the *One Identity Manager Configuration Guide*.

Appendix B

Default project template for the SAP R/3 Compliance Add-on Module

A default project template ensures that all required information is added in One Identity Manager. This includes mappings, workflows, and the synchronization base object. If you do not use a default project template you must declare the synchronization base object in One Identity Manager yourself.

Use a default project template for initially setting up the synchronization project. For custom implementations, you can extend the synchronization project with the Synchronization Editor.

Use the **SAP R/3 authorization objects** project template to synchronize authorization objects and transactions. The project template uses mappings for the following schema types.

Table 31: Mapping SAP R/3 schema types to tables in the One Identity Manager schema

Schema type in the target system	Table in the One Identity Manager schema
TACT	SAPActivity
AUTHX	SAPField
FieldHasRcTable	SAPFieldHasSAPRCTable
MenuHasTransaction	SAPMenuHasSAPTransaction
TOBJ	SAPAuthObject
ObjectClass	SAPAuthObjectClass
ObjectHasActivity	SAPAuthObjectHasSapActivity
ObjectHasField	SAPAuthObjectHasField
ProfileHasAuthObjectField	SAPProfileHasAuthObjectElem
ProfileHasAuthObjectFieldPart1	SAPProfileHasAuthObjectElemPart1
ProfileHasAuthObjectFieldPart2	SAPProfileHasAuthObjectElemPart2

Schema type in the target system	Table in the One Identity Manager schema
ProfileHasAuthObjectFieldPart3	SAPProfileHasAuthObjectElemPart3
ProfileHasAuthObjectFieldPart4	SAPProfileHasAuthObjectElemPart4
RcTable	SAPRCTable
Variable	SAPRCVariable
RFCFUNCTION	SAPTransaction
TMENU01	SAPMenu
Transactions	SAPTransaction
TRANSACTIONHASTOBJ	SAPTransactionHasSAPAuthObject
USOBHASH	SAPTransaction

Appendix C

Referenced SAP R/3 tables and BAPI calls

The following overview provides information about all the tables referenced by SAP authorization objects in an SAP R/3 system and the BAPI calls that are run. The tables and BAPIs accessed by the SAP R/3 connector when SAP R/3 basis administration is synchronized are listed in the *One Identity Manager Administration Guide for Connecting to SAP R/3*.

Table 32: Referenced tables and BAPIs

Tables	BAPI Calls
AUTHX	AUTH_TRACE_GET_USOBHASH
OBJCT	RFC_READ_TABLE or /VIAENET/READTABLE
TACT	AUTH_TRACE_GET_USOBHASH or /VIAENET/LISTUSOBHASH
TACTZ	/VIAENET/LISTMENU01
TFDIR	
TFTIT	
TMENU01	
TMENU01R	
TMENU01T	
TOBJ	
TOBCT	
TSTC	
TSTCT	
USOBHASH	
USOBX_C	
USR10	
UST10S	

Tables	BAPI Calls
UST12	
USVART	

About us

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

Contacting us

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

Technical support resources

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

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

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

Index