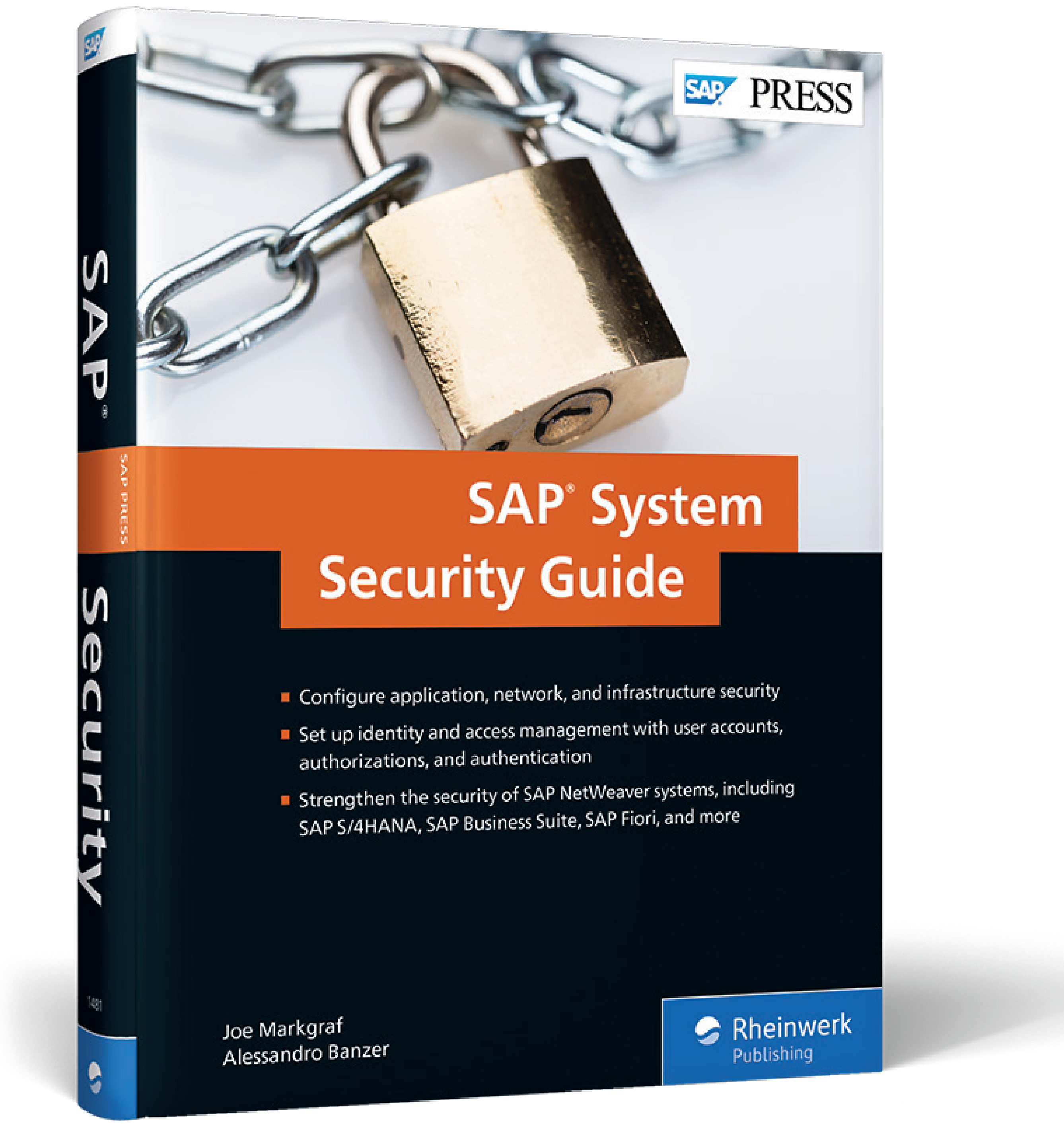
SAP PRESS
First-hand knowledge.
Rheinwerk Publishing



Reading Sample

In these chapters you will learn how to secure SAP system clients and how to log system activities and prepare for audits.

Joe Markgraf and Alessandro Banzer

SAP System Security Guide

574 Pages, 2018, $79.95
ISBN 978-1-4932-1481-5
www.sap-press.com/4307

Chapter 4
Securing Clients

4

Before reading this chapter, you must have a basic understanding of the client concept in an SAP system. You must also understand how to navigate within SAP GUI.

In an SAP NetWeaver system, all business data is isolated on the client level. This means that users that work in one client can't access the data of another client. This architecture is ideal for shared systems that multiple organizations might use. It also allows for the separation of different clients for different activities or use cases. For example, testing clients and development clients could be created on the same SAP installation to allow users to develop and test in the same system without getting in each other's way. Some organizations will choose to have multiple development clients, or multiple test clients. Others will use different clients to separate HR and finance activities. Some organizations will have each of their subsidiaries operate in a separate client in the same master system.

Here are the basic rules that define clients:

- Clients can never read or write to other clients.
- The business data of a client is separated from other clients.
- Clients share the same SID but have different client numbers.
- Multiple clients may exist in an SAP system.
- Clients may be copied or deleted and won't affect other clients.

How is a client different than just having another system? To start, multiple clients can exist in a single system. A client will typically represent a separate organization or company within an SAP system but share the same technical SAP NetWeaver instance. Therefore, the overhead to maintain the instance is shared. However, some organizations will adhere to a strict, productive single client per system. It all depends on the architecture your organization has chosen.

You can think of clients as floors of an office building. Multiple organizations can occupy offices on different floors in an office building. All the building tenants share the same infrastructure (power, water, Internet, heating), but they operate as separate entities isolated from each other by the floor and ceiling. What's said on one floor isn't overheard on another. A disruption on the top floor won't affect the bottom floor.

One key takeaway from this example is that in the office building some infrastructure is shared; in an SAP system, this shared infrastructure is called *client-independent*. Client-independent objects or tables are common for all clients. On the other hand, *client-dependent* objects are never shared with other clients.

Client-Dependent Database Tables: MANDT Field

The technical table field that denotes a client is the `MANDT` field. This field is present in all client-dependent tables. Client-independent tables don't have a `MANDT` field and represent any and all clients. Use care when changing client-independent tables as they affect all clients.

Most SAP NetWeaver systems have at least two clients, if not more. To identify what clients exist in your SAP system, simply look at table `T000`, the clients table (see Figure 4.1).

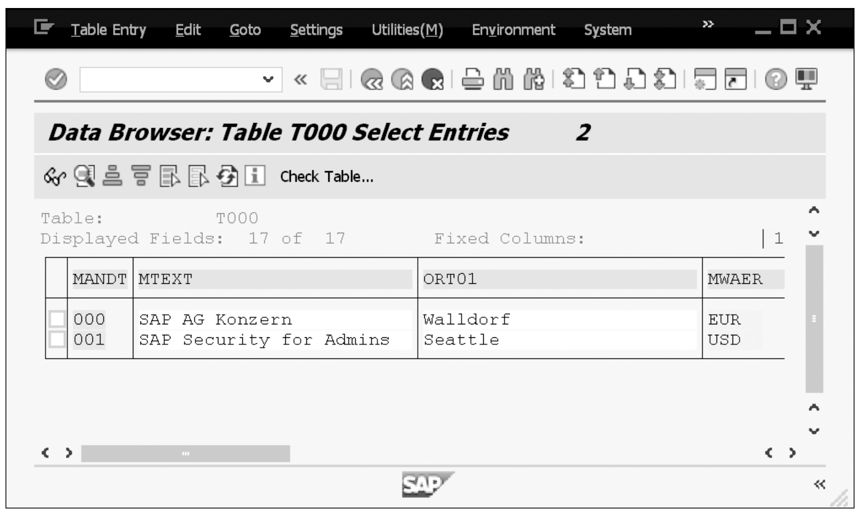


Figure 4.1 Table T000: Client List

Common clients you will see are client 000, client 001, and client 066. These clients are usually delivered/created by SAP. You may see more or fewer clients, depending

on how your SAP system has been set up. You'll also see one or more productive clients, or clients that contain your business data. These are clients that your users will log in to and perform work on. Later in this chapter (Section 4.1.3), we'll cover more about securing clients, but for now let's explore the basics.

Now that you're familiar with the concept of multiple clients, let's explore the possible settings for each client. Some clients will be used to change code, others could be used for testing, and some will always be used by business end users in production. Client settings tell the SAP system what's allowable and what's restricted in each of these clients.

In a production or testing client, you wouldn't want a developer to be able to change objects. On the other hand, in a development client you would want this activity to be allowed. Settings like this are what we use to achieve a desired client scenario. In Section 4.1, we'll walk through how to check the current settings for a client.

As security administrators, we're interested in client settings because we'd like to prevent users from being able to change objects unless absolutely necessary. Even if the client settings are correct in one client, an errant setting in another client could lead to changes being made and passed to another client within the same system, *even if that client had the correct settings*. It's imperative that client settings are closely managed for all clients within both an SAP system and all SAP systems within a landscape.

From time to time an administrator may be asked to change the settings of a client. This activity should always be done temporarily because a client should have a steady state in which its settings are fixed. Often, clients are opened for simple changes and are then forgotten about and stay open until the next audit—or even worse, a malicious user—discovers the issue. Take care not to let this happen in your organization.

4.1 Client Settings

It's important for a security administrator to know and understand the different possible client settings and what they may be used for. Before we explore specific settings in depth, let's walk through how to check client settings in the system.

To check client settings, follow these steps:

1. Navigate to Transaction SCC4 (see Figure 4.2).
2. Double-click the client you'd like to view. For this example, client 001 has been selected (Figure 4.3).

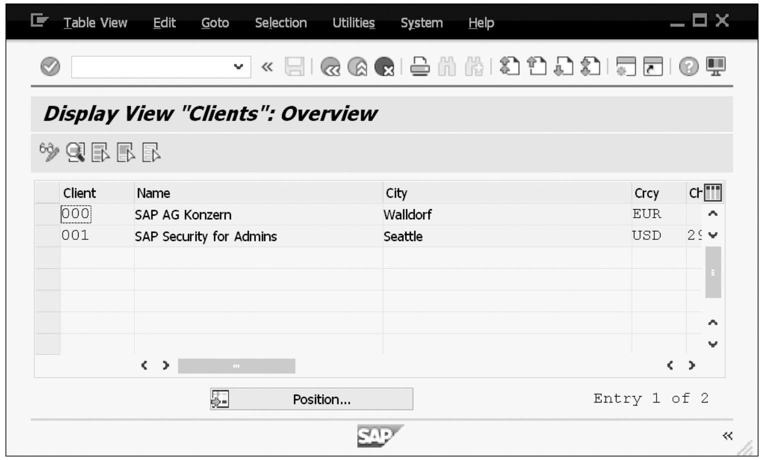


Figure 4.2 Transaction SCC4: Display Clients

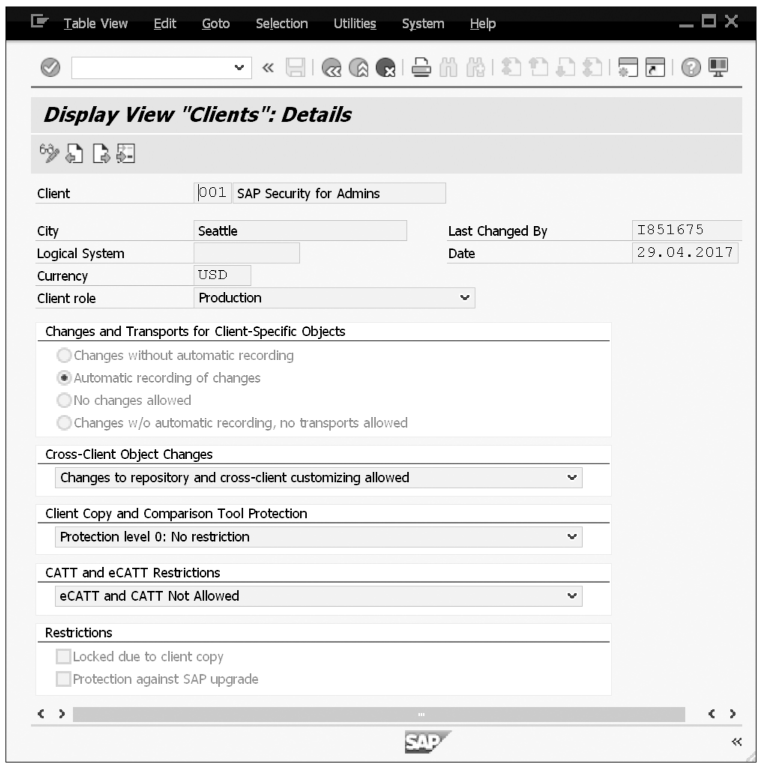


Figure 4.3 Details of Client 001 in a Demo System

Now that you know how to navigate to these settings, let's explore the information on this screen in more detail.

4.1.1 Client Setting Fields

When viewing a client's settings, you'll see the following fields:

- **Client number**
  This is a three-digit number that identifies the client within the system. This must be unique and is assigned when the client is created.
- **Client name or short text**
  Each client can have a short name assigned to it that helps identify it.
- **City**
  The city designation helps differentiate different clients when multiple organizations or divisions are used.
- **Logical System**
  The logical system is a technical identifier that comes into play when using system-to-system communication. It's very important to have a proper logical system name defined.
- **Currency**
  This field denotes what standard currency the client uses.
- **Last Changed By**
  This field denotes which user last changed the settings of the client. It is often checked for auditing purposes.
- **Date**
  This field denotes the date the client was last changed.
- **Client Role**
  Possible choices are as follows:
  - **Production**
    For the active use of business users. It's essential that no changes are made in this client.
  - **Test**
    Developers use this client setting to test their Customizing settings and workbench developments.
  - **Customizing**
    For the creation of Customizing settings and workbench developments.

  - **Demo**
    For demonstration or prototyping purposes.
  - **Training/Education**
    Typically used to train users on changes before import into production.
  - **SAP Reference**
    Clients used by SAP.
- **Changes and Transports for Client-Specific Objects**
  Client-specific objects have values based on a client value. This means that a client-specific object can have a different value based on what client it's contained in. These options cover changes to these objects and how they're transported using the transport system. Possible choices are as follows:
  - **Changes without automatic recording**
    This means that changes in the customizing settings of the client are allowed. They aren't automatically captured in a transport for moving to other systems or clients. Changes can be manually transported to other clients or systems.
  - **Automatic recording of changes**
    This means that changes to the customizing settings of the client are allowed. They're automatically captured in a transport for moving to other systems or clients.
  - **No changes allowed**
    Changes to the customizing settings of the client aren't allowed with this setting.
  - **Changes w/o automatic recording, no transports allowed**
    Changes are allowed to the customizing settings of the client but may not be transported with this setting.
- **Cross-Client Object Changes**
  Cross-client objects have a single value for the entire system. This means that cross-client objects have the same value regardless of what client the user's logged into. These options cover changes to these objects and how they're transported using the transport system. Possible choices are as follows:
  - **Changes to repository and cross-client customizing allowed**
    There are no restrictions on the changes of cross-client objects for the client when this setting is used. Both cross-client Customizing objects and objects of the SAP repository can be changed.

  - **No changes to cross-client customizing objects**
    Cross-client Customizing objects can't be changed in a client with this setting.
  - **No changes to repository objects**
    Objects of the SAP repository can't be maintained in a client with this setting.
  - **No changes to repository/cross-client customizing objects**
    Combination of both previous restrictions: neither cross-client Customizing objects nor objects of the SAP repository can be changed in a client with this setting.
- **CATT and eCATT Restrictions**
  This setting either allows or restricts the Computer-Aided Test Tool (CATT) and enhanced CATT (eCATT), which are scripting utilities used for automated testing. This setting either permits these scripts to run or prevents them from doing so.
- **Restrictions**
  This setting outlines other restrictions that can be made to the client. The options are:
  - **Locked due to client copy**
    This checkbox will indicate when the client is locked against logon. It's used during a client copy to prevent data changes during the copy. It's not a selectable box because it only indicates status.
  - **Protection against SAP Upgrade**
    This checkbox will prevent an upgrade from taking place on this client when the system itself is being upgraded. It's only used in exceptional cases.

4.1.2 Suggested Client Settings

Table 4.1 through Table 4.4 list the suggested client settings for typical use cases. To summarize, production and test clients shouldn't be open to changes. However, development clients should be because their purpose is to implement changes. As always, client 000 should also be protected from changes because it's the SAP-delivered reference client.

Settings	Client 000, Any System
Client role	SAP reference
Changes to client-specific objects	No changes allowed

Table 4.1 Suggested Client Settings for Client 000 in All Systems

Changes to cross-client objects	No changes to SAP repository or Customizing
Client copy protection	Protection level 1: no overwriting

Table 4.1 Suggested Client Settings for Client 000 in All Systems (Cont.)

Settings	Productive Clients
Client role	Production
Changes to client-specific objects	No changes allowed
Changes to cross-client objects	No changes to SAP repository or Customizing
Client copy protection	Protection level 1: no overwriting

Table 4.2 Suggested Client Settings for Productive Clients

Settings	Testing Clients
Client role	Test
Changes to client-specific objects	No changes allowed
Changes to cross-client objects	No changes to SAP repository or Customizing
Client copy protection	Protection level 0: no restrictions

Table 4.3 Suggested Client Settings for Testing Clients

Settings	Development Clients
Client role	Customizing
Changes to client-specific objects	Changes are automatically recorded
Changes to cross-client objects	Changes allowed to SAP repository or Customizing
Client copy protection	Protection level 1: no overwriting

Table 4.4 Suggested Client Settings for Development Clients

4.1.3 Changing Client Settings

Now, let's walk through how to change client settings. Follow these steps:

1. Navigate to Transaction SCC4.
2. In the upper-left menu, click **Table View**, then select **Display • Change** (Figure 4.4).

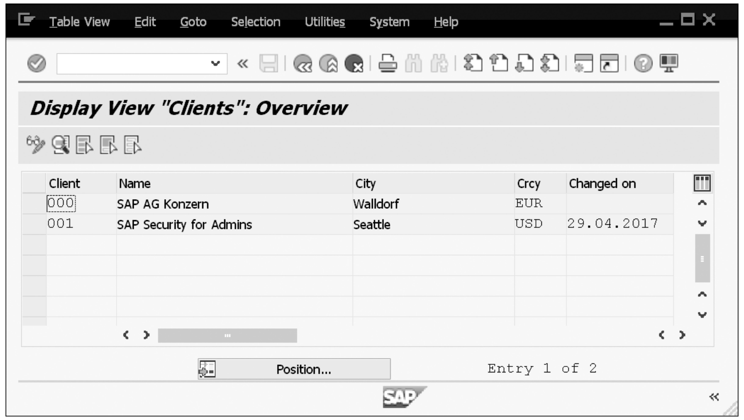


Figure 4.4 Change Table View

3. The system will prompt you with a warning about the table being cross-client (Figure 4.5). Click the check button to proceed.

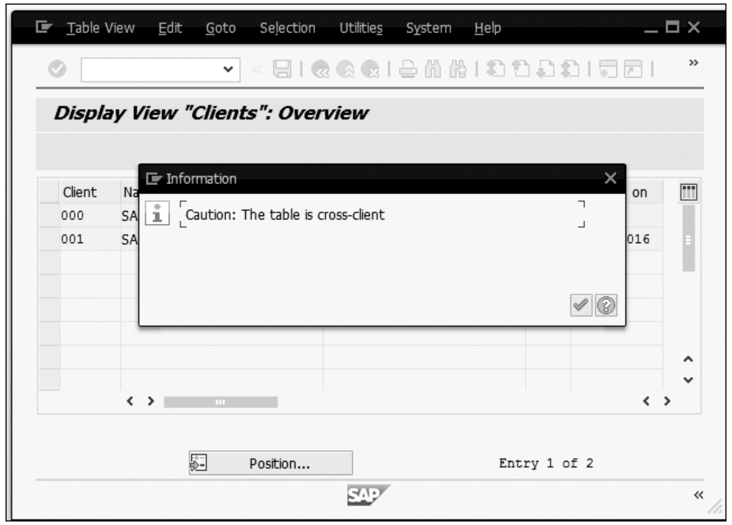


Figure 4.5 Cross-Client Warning

4. Double click on the row of the client you'd like to change settings for (Figure 4.6).

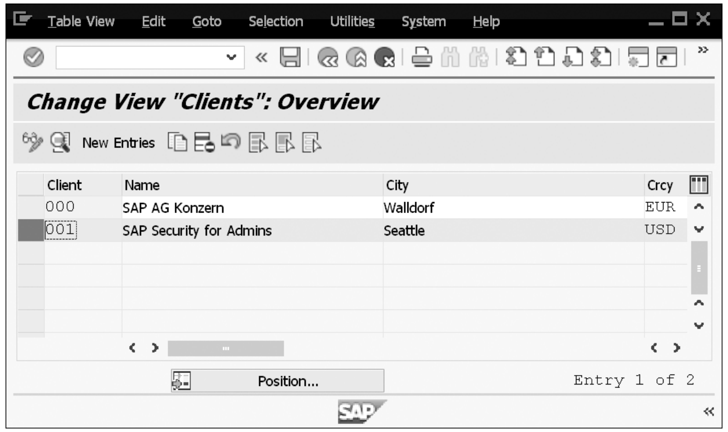


Figure 4.6 Select Client to Change Settings

5. The system will now display, in change mode, the settings for the client you have selected (Figure 4.7).

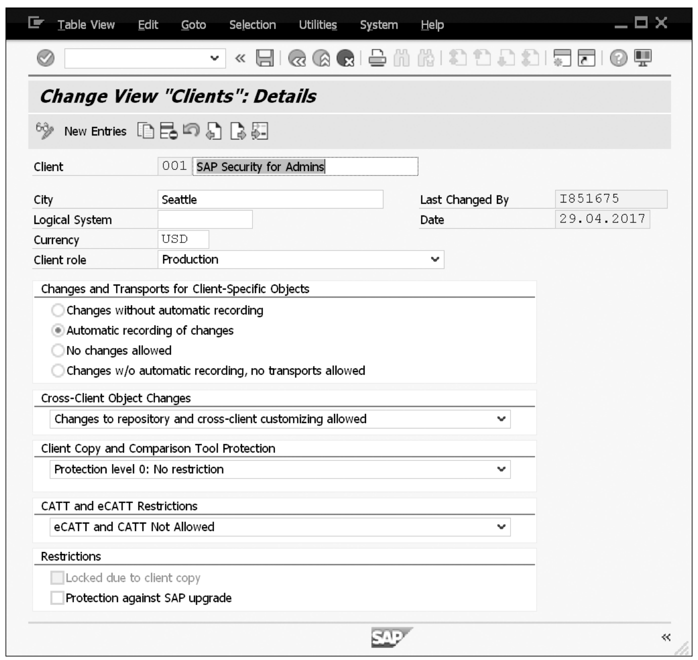


Figure 4.7 Change Mode in Transaction SCC4

6. Once you've made your changes, click the **Save** icon.

Depending on your chosen client settings, you may see a transport request. This is to ensure that your settings can be move to any other systems you choose. If you don't want to transport your client settings, delete the transport that you create to contain this change.

4.2 Client Logon Locking

Occasionally, you'll need to lock a client. This may be for an upgrade or a system maintenance activity. Locking the client will prevent users from logging into the client that is locked. A similar effect can be gained by locking all users in a client using Transaction SE10, but the method described in this section is more quickly implemented. Locking using Transaction SE10 will be covered in Chapter 6.

Remote Locking

This procedure can be done in any client, to any client, or with an RFC connection to a remote system with the proper authorizations.

To lock a client and prevent logon, follow these steps:

1. Navigate to Transaction SE37.
2. Enter the **Function Module** name "SCCR_LOCK_CLIENT" and click the **Test/Execute** button in the toolbar (Figure 4.8).

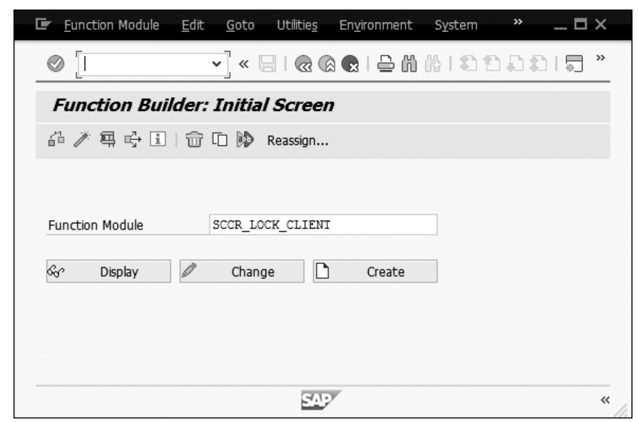


Figure 4.8 Enter Lock Client Function Module Name

3. Enter the number of the client for which you'd like to prevent logon (Figure 4.9). Click the **Execute** button in the toolbar.

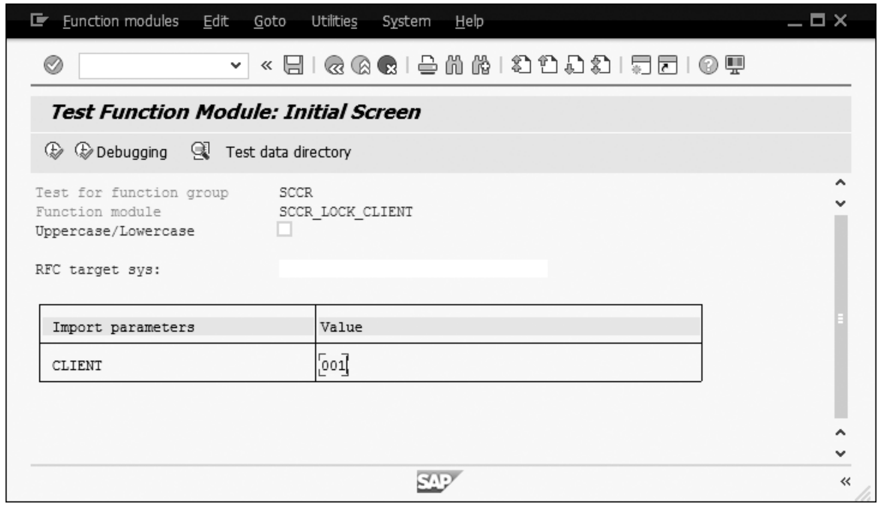


Figure 4.9 Enter Number of Client to Lock

Now, if a user attempts to access the locked client, he will receive the notification seen in Figure 4.10.

Figure 4.10 Client Locked against Logon Notification

To unlock a client, follow these steps:

1. Navigate to Transaction SE37.
2. Enter the **Function Module** name "SCCR_UNLOCK_CLIENT" and click the **Test/Execute** button in the toolbar (Figure 4.11).

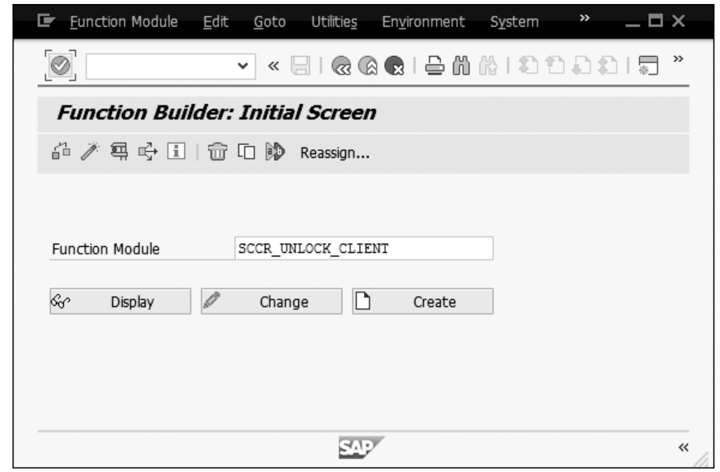


Figure 4.11 Enter Unlock Client Function Module Name

3. Enter the number of the client you'd like to unlock for logon (Figure 4.12). Click the **Execute** button in the toolbar.

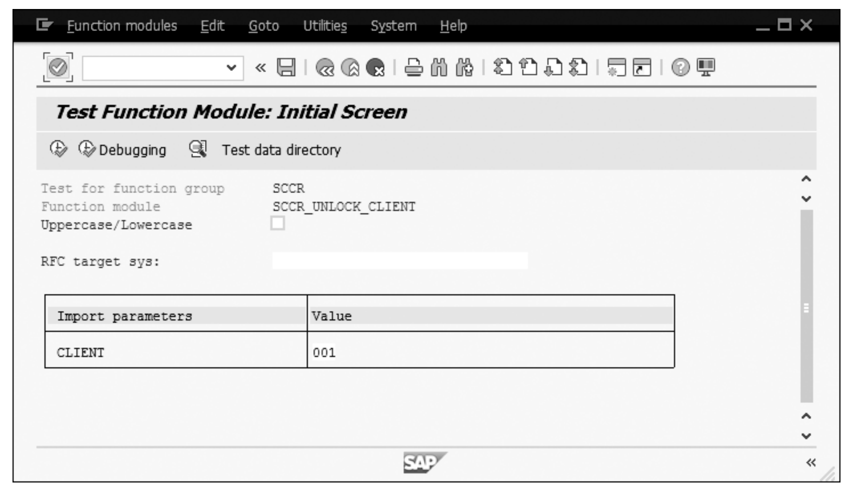


Figure 4.12 Enter Number of Client to Unlock

4.3 Summary

In this chapter, you learned about client settings and how they're used to control what's allowed in each client. We covered what settings are appropriate in specific client roles and what the production client should be set to. You also learned how to lock users out of a client and how to reverse that lock.

In the next chapter, you will learn about the set of executables that make up the SAP NetWeaver AS ABAP system, called the kernel. The kernel is an integral part of the system that administrators must keep up to date.

Chapter 11

Auditing and Logging

To keep a system secure, it's essential to have eyes on all parts of the system and the changes being made therein. Security audit logging records all security events for later analysis; table logging records changes made to tables, including when the changes were made and by whom. In this chapter, you'll learn to configure and enable security audit logging and table logging.

Certain activities in the SAP system are periodically checked and reviewed by an auditor. Therefore, you must ensure that those activities are recorded in the system. The security audit log provides a framework to record security-related events in the system—for example, Remote Function Calls (RFCs), logon attempts, changes to the audit configuration, and so on. The security audit log doesn't log changes to the data within the SAP system that are stored in the database. However, with the table-logging functionality, you can record changes to a table. It's not recommended to log all table changes—only the ones that are considered important and hence for which changes must be traceable. For example, important tables include table `T000` (clients), table `TCUR` (exchange rates), and others.

To analyze the workload of the SAP system, you can use the Workload Monitor, which is also a neat tool to analyze a user's history. The Workload Monitor records historical usage data and allows you to drill down on a user level.

As data protection laws gain ground, protecting your data becomes more and more important. To protect the privacy of and personal information in your SAP system, along with sensitive and classified data, you can use Read Access Logging (RAL) to record read activity.

In this chapter, you'll learn how the different logging functionalities work, what makes them unique, and the impact on your system.

11.1 External Audits

Often, a security administrator will find herself being asked to help with an external audit. Before we tackle the task of assisting with an audit, first we'll cover what these audits do for a company.

External audits are typically financial; that is, they center on the financial records of the company. These audits typically focus on any customer running the SAP ERP or SAP S/4HANA finance functionality on SAP NetWeaver AS ABAP. Two common audits that organizations go through are to check compliance with the Sarbanes-Oxley Act (SOX) and the International Financial Reporting Standard (IFRS). Each of these audits is performed by an *external auditor*, an organization outside of your own that performs the audit. This organization will send one or several auditors who will be tasked with observing and recording proof that the practices of your organization comply with the controls required for your audit.

The Sarbanes-Oxley Act of 2002 set forth internal financial auditing controls in the United States that must be adhered to when preparing financial information for reporting purposes. US-based financial systems are routinely audited to SOX standards.

IFRS is an audit of accounting systems such that they can be compared between countries reliability. It's common to see IFRS audits performed for multinational companies.

Besides these two, there are many other audits that vary country by country. These auditing standards generally are prepared by a country's government-mandated accounting standards organizations and commonly follow Generally Accepted Auditing Standards (GAAS).

The external auditor will be working off a set of controls, in which the security administrator will most likely be the person that is running the queries in the SAP system to satisfy those queries. Most queries are run through the User Information System (UIS; Transaction SUIM). We'll cover the use of the AIS later in this chapter. Auditors may also ask for the output of some standard reports, among other things.

Often, auditors may also ask for access to your system to run reports on their own. Unless this is legally required, it's a good idea to deny this request. When given the choice, it's a more efficient practice for the SAP security administrator to run queries given to them by an auditor. This is done to keep the security administrator in control of the scope of the audit. If an audit is for financial compliance, the auditor should be looking at finance-related authorization objects. Too often, auditors are

given free access to a system, which tends to change the scope of the audit to whatever the auditor feels like digging into.

Often, external audits are focused into categories similar to the following:

- Internal controls
- Network activity
- Database activity
- Login activity (success and failures)
- Account or user activity
- Information access

For each such category, the auditor will require proof that the controls for that category are being applied. They may also ask for a random sample of users or transports, or even provide a time frame and ask to see logs or proof that controls were being adhered to for that time.

11.2 Internal Audits

Internal audits are performed by individuals within your own organization. Often, they focus on preparing for an external audit. However, this isn't always the case. Internal audits can be used to ensure that a specific control or policy is being followed by examining system activity, logs, or even user master records. This type of activity is usually mandated by either the security administrator or an internal audit department for the purposes of verification.

Quite often, when an internal audit is performed, the objective is to improve adherence to the controls that will be followed for an external audit. This will often leave the security administrator with a to-do list to satisfy the audit requirements. In addition, the security administrator may be consulted to help create controls that will help keep compliance such that it's not a major effort when an external audit is performed.

One of the common tasks for an internally led audit is to manage the number of users that have powerful authorizations, like SAP_ALL, or access to perform business-critical tasks, like pay vendors or create accounts. This is done by evaluating the roles and authorization objects that each user master record contains.

The internal audit is also a good time to determine the effectiveness of your general security operations and process. Defining a set of controls and evaluating your system and users based on those controls can help enforce a strong, consistent level of security.

11.3 Auditing Tools

SAP systems are equipped with a set of tools that can be used for auditing. Such tools include the security audit log, the system log, table logging, the Workload Monitor, as well as Read Access Logging and the User Information System. All these tools can be utilized to extract and analyze data about certain activities in the system, such as who logged on to a system, who changed a certain table, who accessed certain data, and more. We'll explore each of these tools in more detail in the next sections.

11.3.1 Security Audit Log

The security audit log (SAL) records security-related activities in the system, such as changes to user master records, logon attempts, RFCs, and so on. This tool is designed for auditors to log and review the activities in the system. With the SAL, an auditor can reestablish a series of events that happened in the system.

The SAL offers wide flexibility in its usage. You can activate and deactivate it, as well as change the filters as necessary. For example, you can activate the SAL before an audit takes place and deactivate it once the audit has been performed. Also, you can change the filters and, for example, monitor a user if you've detected suspicious activity in the system.

The audit log must be activated before it can be used. To activate the audit log, you have to specify which activities you want to record in the security audit log. The following activities are available:

- Successful and unsuccessful dialog logon attempts
- Successful and unsuccessful RFC logon attempts
- RFCs to function modules
- Changes to user master records
- Successful and unsuccessful transaction starts
- Successful and unsuccessful report starts
- Changes to the audit configuration

In addition to these events, the security audit log also logs certain activities that aren't categorizable, such as the following:

- Activation and deactivation of the HTTP security session management or instances in which HTTP security sections were hard-exited
- File downloads
- Access to the file system that coincides with the valid logical paths and file names specified in the system (particularly helpful in an analysis phase to determine where access to files takes place before activating the actual validation)
- ICF recorder entries or changes to the administration settings
- The use of digital signatures performed by the system
- Viruses found by the Virus Scan Interface
- Errors that occur in the Virus Scan Interface
- Unsuccessful password checks for a specific user in a specific client

Once activated, the system will record the activities into a log file on the application server.

Warning

Be cautious when activating the security audit log because it contains personal information that may be protected by data protection regulations—especially with the new GDPR regulation from the European Union but also other protection laws in other regions. Make sure that you adhere to the regulations in your area.

Versions

Your SAP_BASIS component affects your version of the security audit log. With SAP NetWeaver 7.5 SP 03 for SAP_BASIS, SAP has introduced new functionality in the security audit log.

In the old version, the main transactions for the security audit log were Transactions SM18, SM19, and SM20. In the new version, SAP introduced several new transactions:

- **Transaction RSAU_CONFIG**
  Maintenance of the kernel parameters and selection profiles relevant for the security audit log

- **Transaction RSAU_CONFIG_SHOW**
  Printable display version of Transaction RSAU_CONFIG
- **Transaction RSAU_READ_LOG**
  Audit log evaluation
- **Transaction RSAU_READ_ARC**
  Audit log evaluation in archive data
- **Transaction RSAU_ADMIN**
  Administration of integrity protection for files; reorganization of log data
- **Transaction RSAU_TRANSFER**
  File-based transfer of an audit profile

With the enhanced functionality and the new transaction codes, SAP delivers new features as well:

- Save the audit log into the database, either in full or in part.
- Filter by user groups with the user attribute **User Group for Authorization Check** from the **Logon Data** tab in Transaction SU01.
- Increase the number of filters from 10 to 90.
- Check the file integrity.
- Use an enhanced authorization concept with authorization object `S_SAL`.
- An API for evaluating log data is provided with the class `CL_SAL_ALERT_API`.

> **Tip**
> If you use the new security audit log, we recommend locking the old transactions with Transaction SM01_CUS in client 000. Parallel usage of the old and new functionality is possible but not recommended.

Usage Scenarios

Depending on your requirements, you can define usage scenarios differently. With the new security audit logging capability, you can define how and where you want to store the audit log, as well as how to access it. With the old security audit log, you could only save data on the file system of the application server; with the new functionality, you can either save on the file system of the application server or in its database. Also, shared scenarios are possible in which some parts will be stored in the database and the some in the file system.

Classical Approach

In the classical approach, similar to the old version, the audit log is only stored on the file system of the application server. You can read the data from the file system, as well as archive and delete old audit log files.

Database Logging

With the new functionality, it's possible to save the audit log into the database. However, system events are stored in the file system as well. Storing the audit log in the database might result in a quick growth of table `RSAU_BUF_DATA`, which holds the data. With the archiving object `BC_SAL` you can, however, archive the data in that table. With the database, you have an improved experience when accessing the data because it's quicker and the requirements for data privacy are met.

Mixed Scenarios

With the enhanced functionality, you can also activate mixed scenarios in which you generally save the logs on the file system but selective events in the database. When saving selective events in the database, you can access the data faster, which results in a significantly increased performance. That makes sense especially when using statistical data or if you run large evaluations against the log data.

In a second scenario, you can use APIs to transfer data from the security audit log to a central monitoring system (e.g., SAP Solution Manager). In that scenario, the SAL saves the data in the file system of the application server. Certain events that are relevant for the central monitoring systems, such as those to create alerts, are stored in the file system and in database table `RSAU_BUF_DATA`. The API that transfers the data will read the data from the table and then automatically delete it. Your logs are still available in the file system but will be removed from the database table and hence don't require archiving activities in the database.

Configuration

The new security audit log offers an enhanced configuration via Transaction RSAU_CONFIG. Let's explore configuration in detail now.

In general, the security audit log requires some parameters and the definition of filters that define which events will be logged.

To define the parameters, enter Transaction RSAU_CONFIG and open the **Parameter** folder (Figure 11.1).

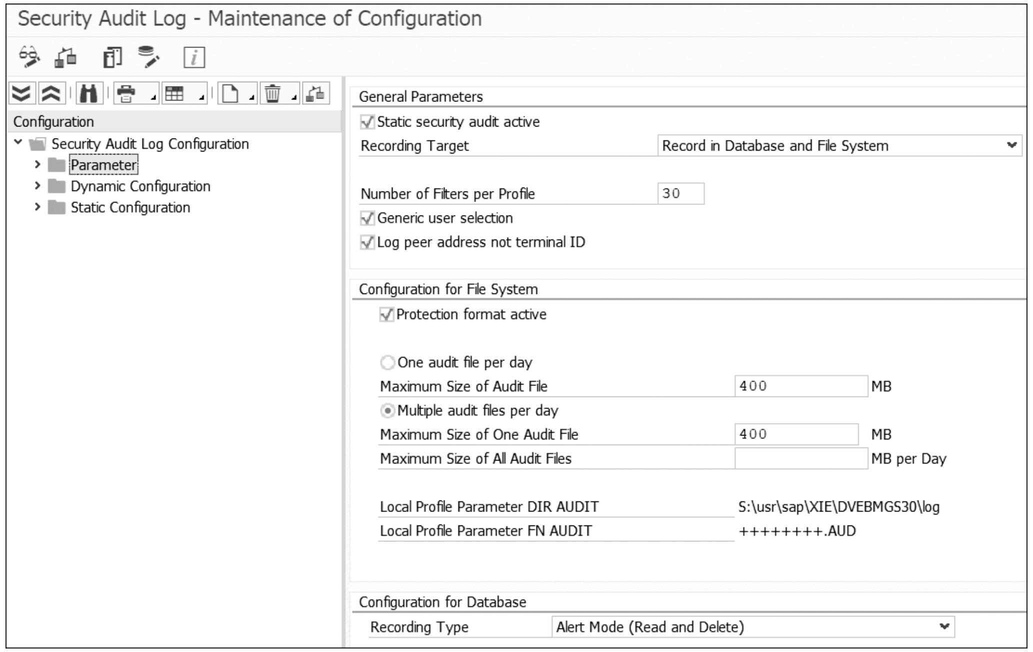


Figure 11.1 Parameter Maintenance in Security Audit Log Configuration in Transaction RSAU_CONFIG

The following can be configured:

- Activate or deactivate logging.
- Define the recording target, whether it's on the file system, in the database, or a combined recording in both the file system and database.
- Define the number of filters per profile, up to 90.
- Define if you'll allow generic user selection with an asterisk (*) character in the filters.
- Define if you log the IP address of the originator and not the terminal ID.
- Activate or deactivate integrity protection format for log files in the file system.
- Define the memory space usage when file system storage is used.
- Define the recording type in the database, whether it's temporary data or permanent data.

In the profiles, you define which events will be logged. To create a new profile or an additional profile, simply right-click the **Static Configuration** folder and choose

Create New Profile. Once the profile is created, you can go ahead and define the settings. Remember that each profile must have at least one filter. To add additional filters, you can simply right-click the **Profile** folder and choose **Create Filter**.

Regardless of the filter you create and specify, it's important to activate the filter once defined by clicking the **Activate** button (Figure 11.2). Only active filters will be selected at the next system start. You can define as many filters as you have defined in the parameter maintenance for each profile.

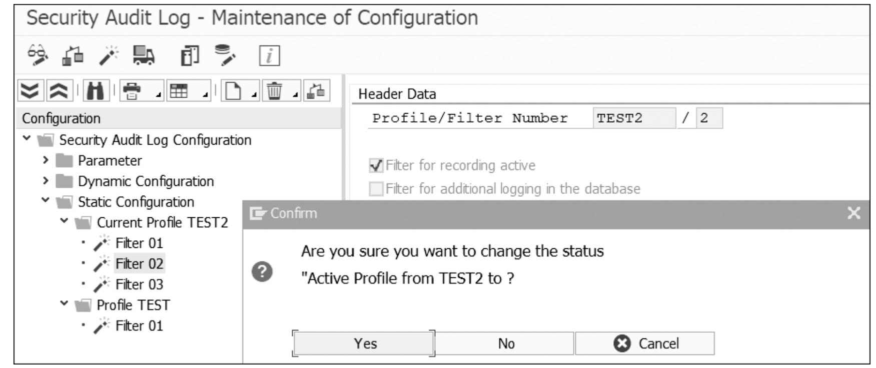


Figure 11.2 Activation of Filters in Transaction RSAU_CONFIG

Each filter that you add to a profile is linked via an OR connector. So, for example, if you have two filters, and the first filter logs everything for user group SUPER and the second filter everything for users starting with RFC*, then those two filters are *OR linked*. That means that all users that belong to user group SUPER and all users starting with RFC* will be logged. Note that user groups only allow for a specific value and that you can't use wildcards as you can for the user name.

In the **Standard Selection** screen, shown in Figure 11.3, where you define the client and whether you want to restrict the logging of a user name or user group, you can select the user group either positively or negatively. **Select by User Group (Positive)** means that you will log all users that are part of that user group. If you use the negative selection, the system logs the events for all users who aren't part of the user group. Possible scenarios for a negative selection can include wanting to log RFC function calls for all users who aren't technical and hence aren't part of a certain user group because those users shouldn't perform RFCs.

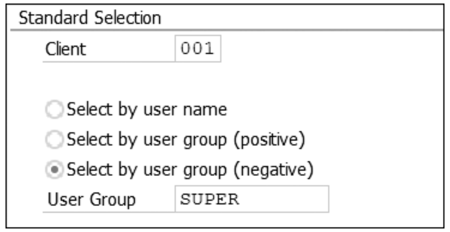


Figure 11.3 Standard Selection in Security Audit Log Configuration

In the **Event Selection** screen (Figure 11.4), you can define which events you want to log. In the **Classic event selection**, you get the same options as in the old security audit log.

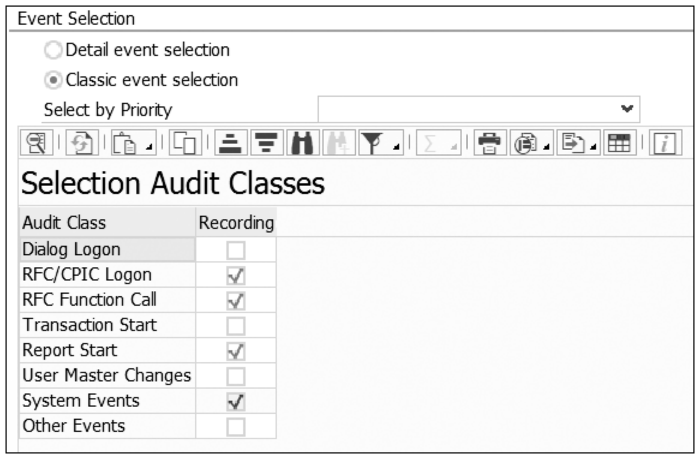


Figure 11.4 Classical Event Selection in Security Audit Log Configuration

In the **Detail event selection** (Figure 11.5), you can slice and dice on a more granular level and pick and choose events more specifically. For example, in the classical selection, you choose **Dialog Logon**, whereas in the detailed selection you can decide whether you want successful logons or failed logons.

If you defined the selection in the classic event selection, the underlying detailed events will be selected.

To start the logging of a filter, it's important that the switch **Filter for Recording Active** is selected. You can have active and inactive filters. Therefore, it's important to keep an eye on the **Active** checkbox, as well as if the filter has been activated.

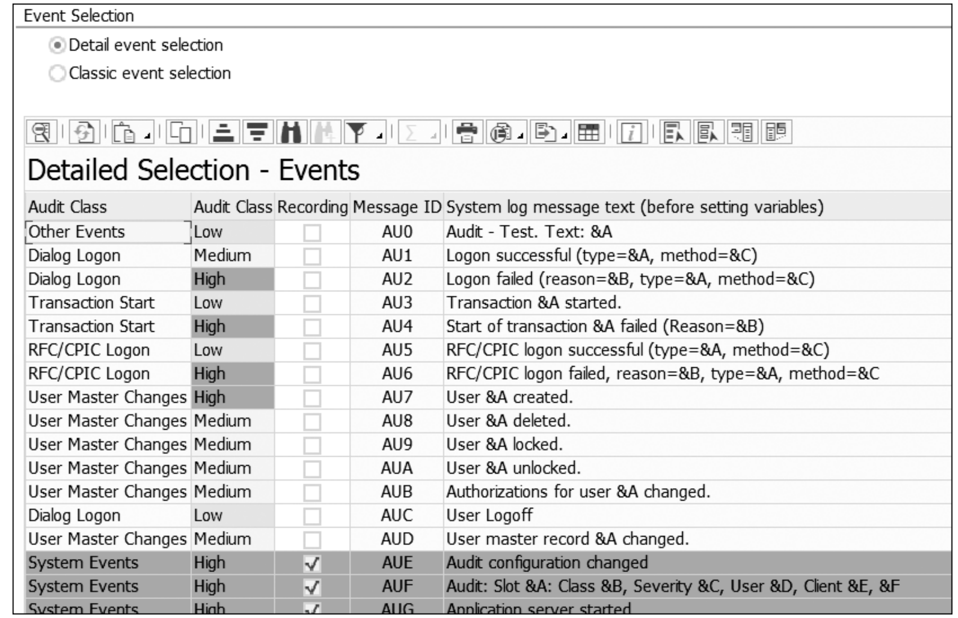


Figure 11.5 Detailed Event Selection Options in Security Audit Log Configuration

Administration of Log Data

The administration of log data takes place in Transaction RSAU_ADMIN (Figure 11.6). In the administration cockpit, you can check the integrity of the file-based log data and reorganize obsolete files. For the database tables, you can use this cockpit to reorganize table RSAU_BUF_DATA by means of deletion or archiving.

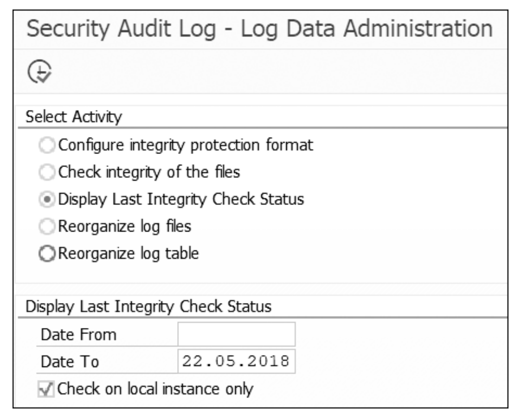


Figure 11.6 Log Data Administration Initial Screen

Integrity Protection

With the integrity protection setting of the SAL, you can protect the security audit log from manipulation of its log files on the file system. However, it doesn't prevent the manipulation of the file but it will tell you if it was manipulated.

To protect the integrity of your files, you can create one hash-based message authentication code (HMAC) per system. To create the HMAC key, choose **Configure Integrity Protection Format** from the initial screen (Figure 11.7) and define your secret **Passphrase**.

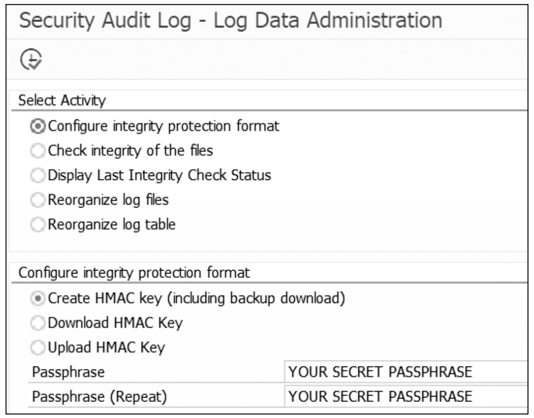


Figure 11.7 Configure Integrity Protection Format

If you wish to restore the key, a local backup file is generated that can be used in combination with the passphrase. Make sure to store the backup file and passphrase so that you can check the integrity of the system later.

If you decide not to create an individual system HMAC key, your integrity is at risk because the integrated key must be considered to be known, as it is set to a default value. That means that you can check the files only against unintentional corruption or change and not against malicious manipulation.

Once configured, all log files written forward will be checked by the integrity protection format. To check the integrity of the files, you can choose **Check Integrity of the Files** on the initial screen (Figure 11.8).

Shorter time frames can be analyzed in the foreground. However, larger periods will be run in the background. Once the check has been performed, you'll see an overview of all the files and their attributes (Figure 11.9). Also, you'll see the status, which indicates whether the file has integrity issues or not.

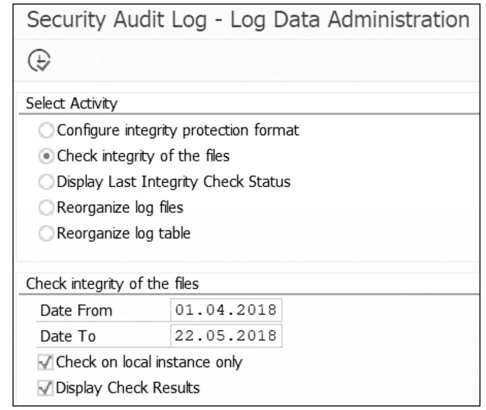


Figure 11.8 Check Integrity of Files

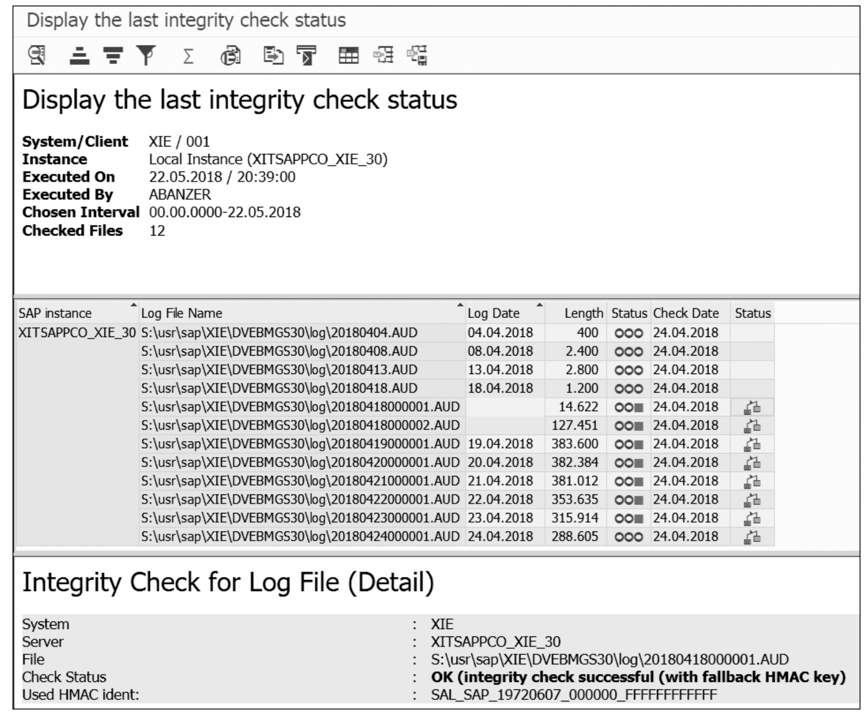


Figure 11.9 Display the Last Integrity Check Status

To quickly navigate back to the last integrity check, you can choose the **Display Last Integrity Check Status** option in the selection screen.

Reorganize Log Files

To reorganize log files by means of deleting the physical file from the file system, you can choose **Reorganize log files** from the initial screen (Figure 11.10). You can delete or display the data to be reorganized, as well as run a simulation mode first. The simulation mode lets you see what will happen if you deselect the checkbox.

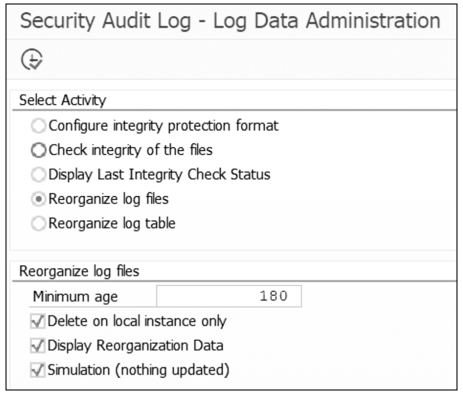


Figure 11.10 Reorganize Log Files

The minimum age decides which files will be deleted. Once executed, you'll see a results screen (Figure 11.11) indicating which old files will be deleted (if run without simulation).

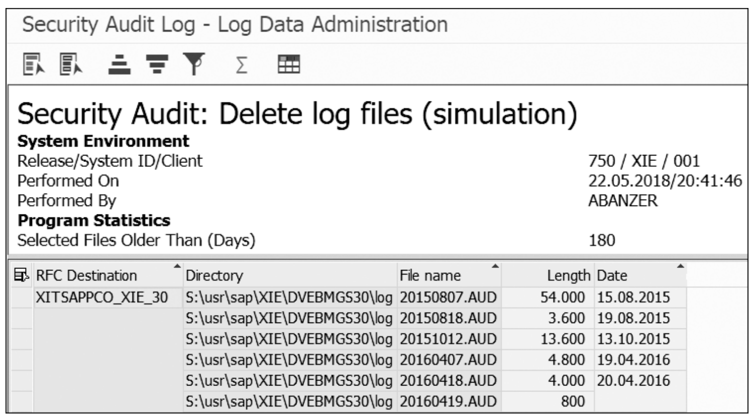


Figure 11.11 Delete Log Files

Remember, the deletion of log files should be carried out through this transaction because it performs an authorization check and follows the deletion process for files in the integrity protection format. Deleting files manually from the file system is considered a manipulation.

Reorganize Log Table

Reorganization of the database table is important when logging is activated to be stored exclusively in the database table. For all other scenarios, reorganization is not necessary; for example, APIs will delete the data after the transfer.

To delete data from the table, choose the **Reorganize log table** selection (Figure 11.12) and enter the date before which you want data to be deleted.

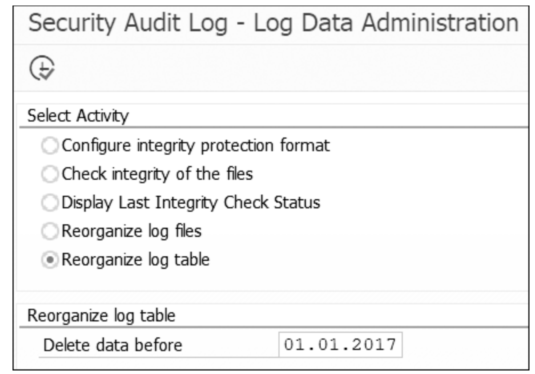


Figure 11.12 Reorganize Log Table in Database

Evaluation of Log Data

You can evaluate the log data in Transaction RSAU_READ_LOG (Figure 11.13). You can either evaluate the logs online in the foreground or send the report into the background. In the selection screen, you can set the time restrictions along with multiple other options.

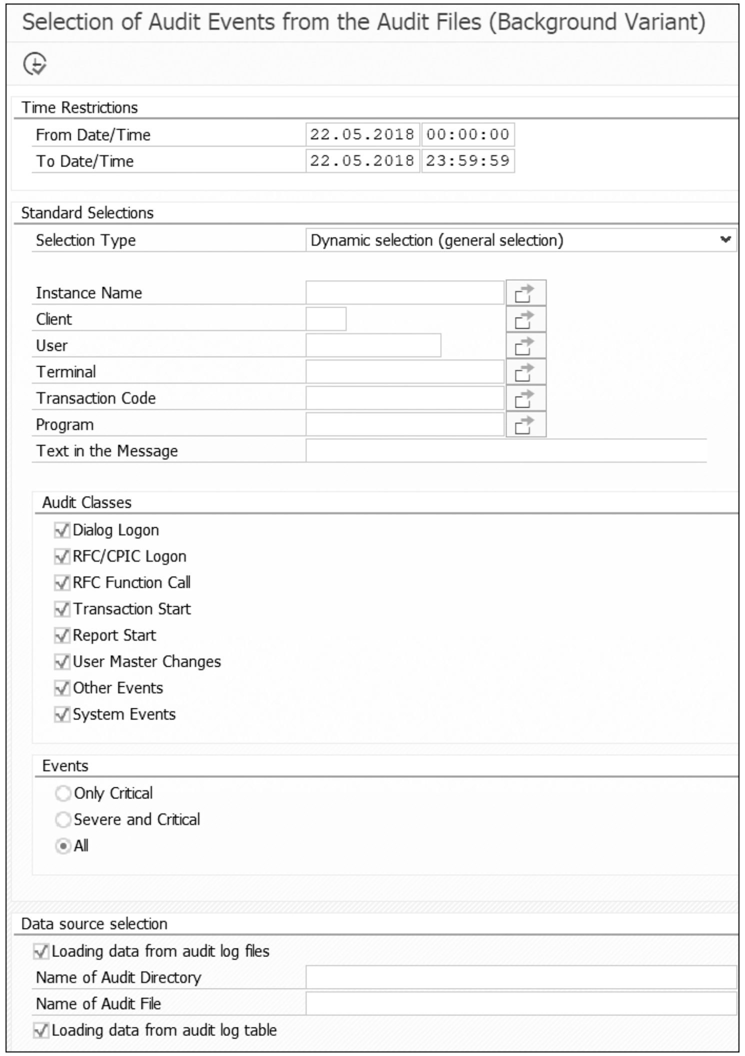


Figure 11.13 Evaluate Log Data Initial Screen

In the **Standard Selections** (Figure 11.14), you can set the selection type and, for example, search based on a specific user, client, terminal, or audit class or based on the criticality of the event. Also, you can reuse your filters and search for specific filters only.

The **Instance Name** field lets you input the instance that you want to evaluate. If you have multiple application servers and want to only include the current application instance, you can use the value <LOCAL>.

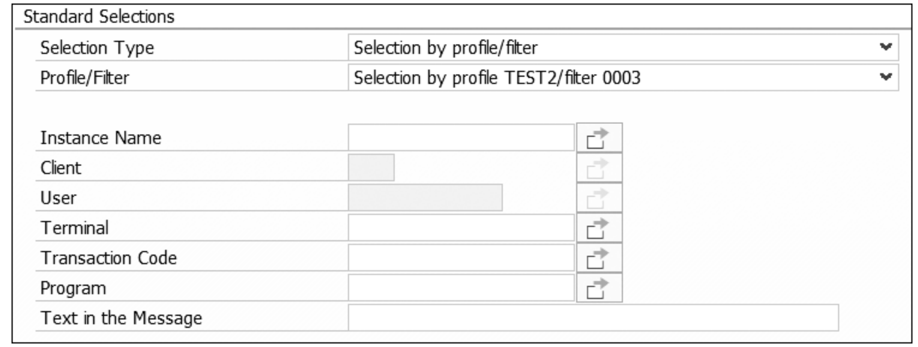


Figure 11.14 Standard Selection in Evaluation of Log Data

In the **Data source selection** (Figure 11.15), you can define if you want to read all your files, a specific file or directory, or your database tables.

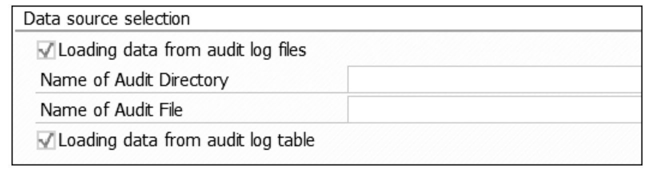


Figure 11.15 Data Source Selection in Evaluation of Log Data

The result screen shows the logged events in detail. For example, in Figure 11.16, you can see successful logons by user WF-BATCH.

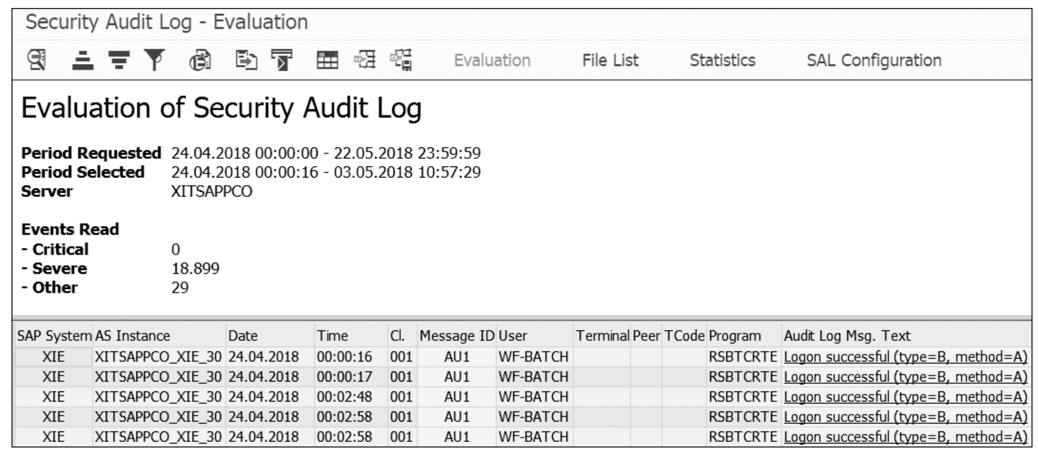


Figure 11.16 Evaluation of Log Data Result Screen

Evaluate Archived Log Data

To evaluate archived log data, you can use Transaction RSAU_READ_ARC (Figure 11.17). In the selection screen, you can set the period, as well as other selections like the client, user, terminal, and so on.

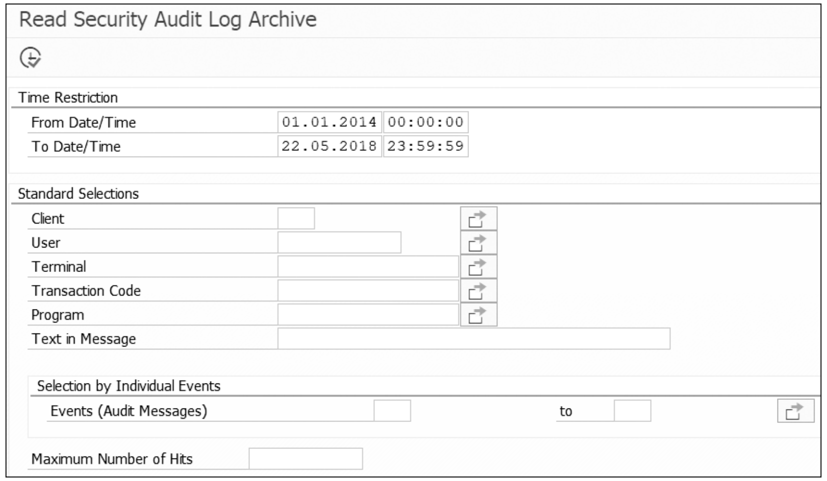


Figure 11.17 Read Security Audit Log Archive Data

11.3.2 System Log

Whereas the security audit log records security-related information about the system, the system log records information that may signal system problems. As an administrator, the system log is an important tool to maintain the healthiness of your system and keep the system up and running with good performance. The system log records warnings, error messages, database read errors, rollbacks, and so on.

The system log offers different types of logging depending on the host. On an UNIX host, you have local and central logging available. If you run on a Microsoft Windows NT host, you'll only have local logging. In the local scenario, the log is stored locally on the application server in a ring buffer. The ring buffer is overwritten once full. Therefore, the system log is only available for a certain time frame as the size is limited. In the central log, each individual application server sends its local log to a central server. Similar to the local log, the size of the central log is limited and hence it doesn't hold the information indefinitely.

In either scenario, we recommend analyzing the system log on a regular basis. Most administrators check the system log daily to avoid any disruption to the SAP system.

The local log is always up to date, whereas the central log might have a slight delay as the data must be written from the local application server to the central server.

The main transaction to analyze the system log is Transaction SM21 (Figure 11.18), in which you can read the system log and its messages. In the selection screen, you can define basic and extended attributes to get to the messages that are most important to you.

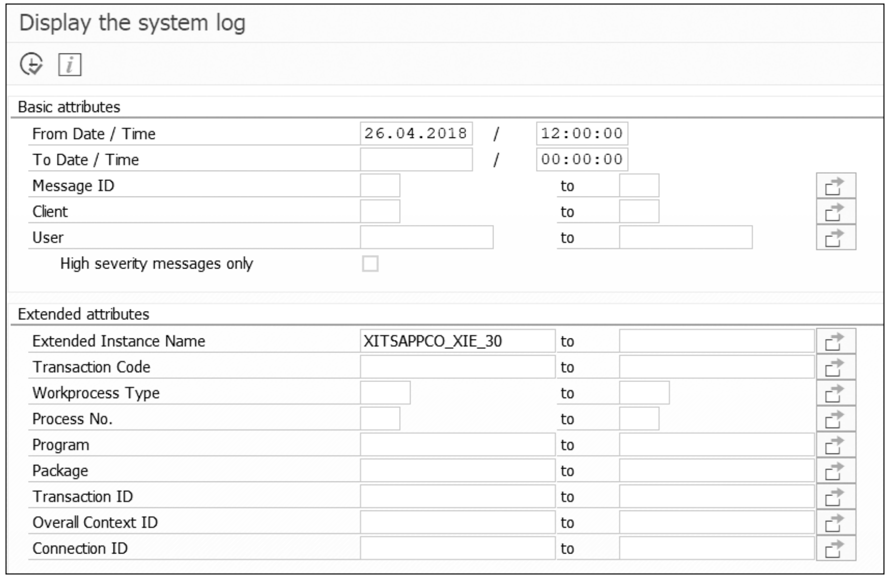


Figure 11.18 Initial Screen in Transaction SM21 to Display System Log

In the result screen (Figure 11.19), you get an overview of all messages that have been logged by the SAP system. For each entry, you see the time stamp, instance, client, user, and the priority of and information about the message. You can double-click any line item.

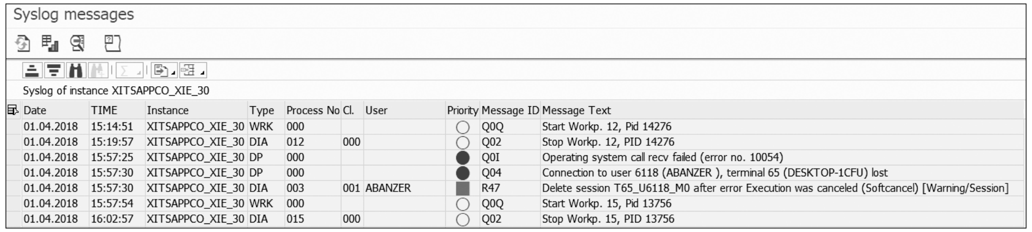


Figure 11.19 System Log Result Screen

After you double-click an item, you'll see to the details of the message (Figure 11.20) to dive further into the error. In addition to details about the message and the session, as well as technical and parameter details, you can also navigate to the trace from the menu bar.

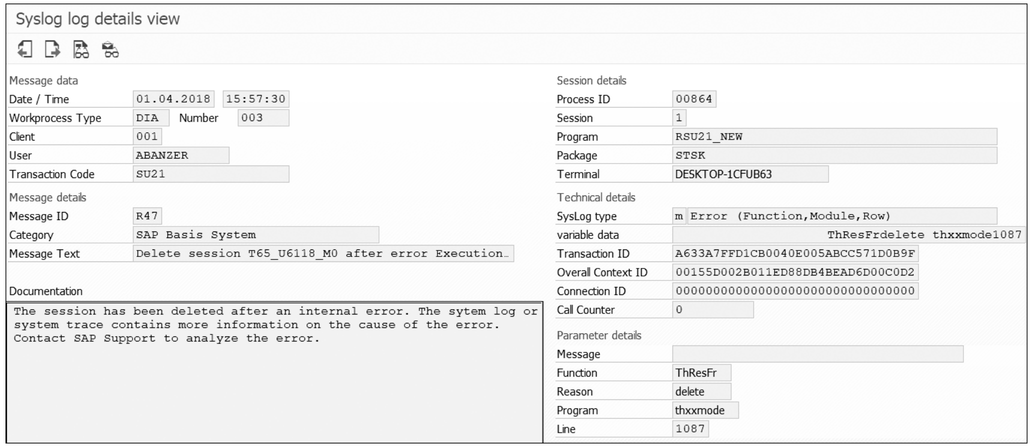


Figure 11.20 System Log Detail View

In the trace, you can see all the steps that the system performed for the message that you selected. Analyzing traces requires a deep understanding of the SAP system and is definitely an expert-tool only.

11.3.3 Table Logging

To enable table logging in general, you have to activate the table logging in the profile parameter `rec/client`. Once activated, you can define tables to be logged in the table properties. The profile parameter `rec/client` knows four different values:

- `OFF`
  Logging is deactivated.
- `nnn`
  Logging takes place for client-specific tables only in the client listed (001, 100, etc.).
- `nnn,nnn,nnn`
  Logging takes place for client-specific tables for the clients listed (a maximum of 10 clients possible, comma-separated).

- `ALL`
  Logging always takes place; for client-specific tables, it takes place for all clients. *Caution*: This setting makes sense only in special cases. Note that in the case of `ALL`, changes are recorded in the log file for all test clients (including SAP client 000).

Once the table logging has been activated, you can define which tables will be logged. To activate logging for a particular table, you have to define the properties in the table itself. You can do that from Transaction SE13. SAP predelivers customizing tables with the table change logging activated.

For the example shown in Figure 11.21 for table `RFCDES` (RFC Destinations), the table change log is activated.

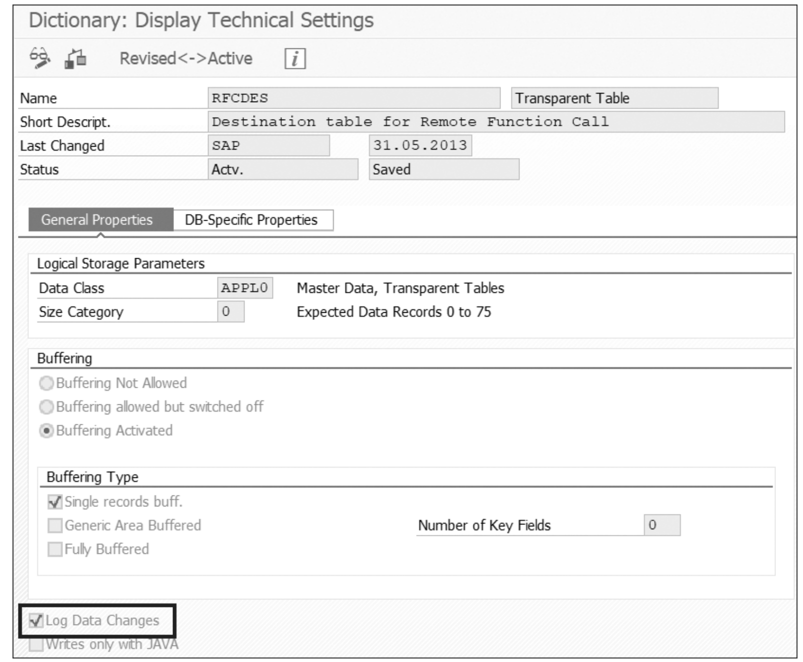


Figure 11.21 Log Data Changes in Technical Settings of Table in Transaction SE13

To check all tables that have the logging activated and to review the changes, you can use Transaction SCU3 (Figure 11.22). In Transaction SCU3, click **List of Logged Tables** to see an overview of all tables that have table change logging activated.

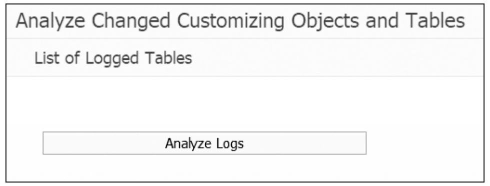


Figure 11.22 Initial Screen of Transaction SCU3

> **Warning**
> In an SAP NetWeaver 7.50 system with SAP ERP installed, SAP defined close to 30,000 tables with the table change log. Most of the tables are customizing tables and hence do not contain master data that changes regularly.

Analyze Logs

To analyze the changes that have been logged, you again can use Transaction SCU3. In Transaction SCU3, go to **Analyze Logs** and make your selections. In the selection screen (Figure 11.23), you must select one specific table or customizing object for analysis. This is enforced because the amount of data can be huge. For reporting purposes, we suggest using the ALV Grid Display, which lets you sort and filter the output.

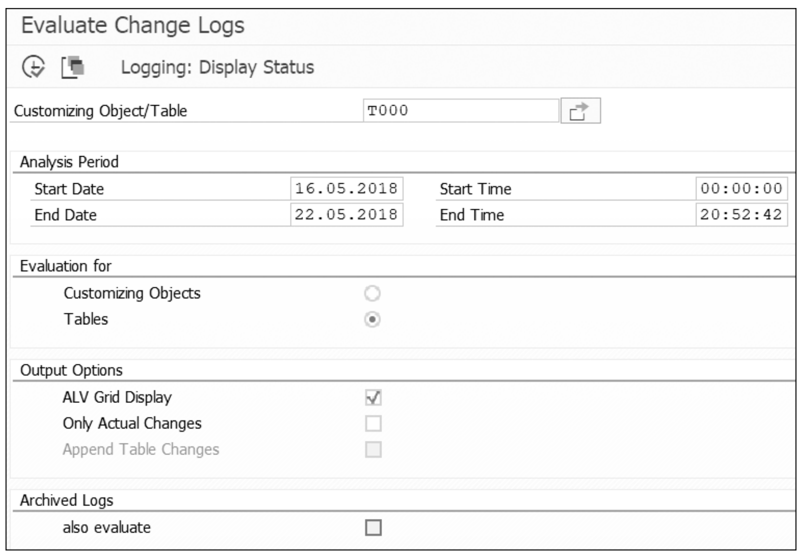
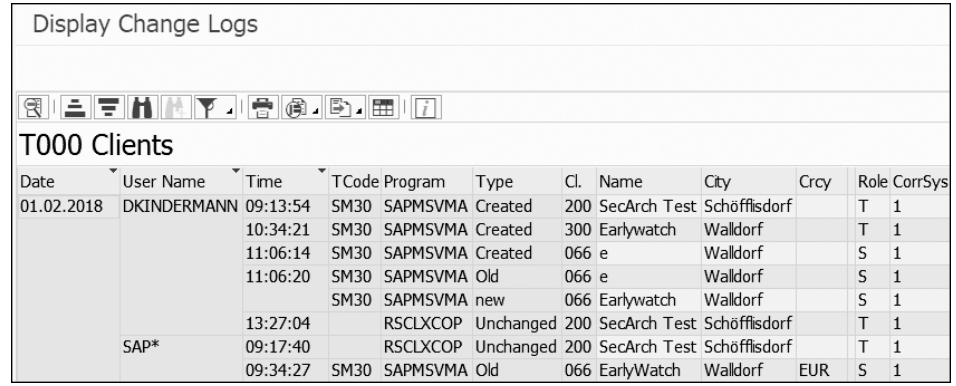


Figure 11.23 Evaluate Table Change Log in Transaction SCU3

In the output view, you can get the details of what's been changed in the table. In Figure 11.24, you can see changes to table T000 (clients). You can see the type of change, as well as which transaction and program were used to perform the change.



Date	User Name	Time	TCode	Program	Type	Cl.	Name	City	Crcy	Role	CorrSys
01.02.2018	DKINDERMANN	09:13:54	SM30	SAPMSVMA	Created	200	SecArch Test	Schöfflisdorf		T	1
		10:34:21	SM30	SAPMSVMA	Created	300	Earlywatch	Walldorf		T	1
		11:06:14	SM30	SAPMSVMA	Created	066	e	Walldorf		S	1
		11:06:20	SM30	SAPMSVMA	Old	066	e	Walldorf		S	1
			SM30	SAPMSVMA	new	066	Earlywatch	Walldorf		S	1
		13:27:04		RSCLXCOP	Unchanged	200	SecArch Test	Schöfflisdorf		T	1
	SAP*	09:17:40		RSCLXCOP	Unchanged	200	SecArch Test	Schöfflisdorf		T	1
		09:34:27	SM30	SAPMSVMA	Old	066	EarlyWatch	Walldorf	EUR	S	1

Figure 11.24 Display Table Change Logs

The data being analyzed in Transaction SCU3 is stored in table DBTABLOG. Transaction SCU3 offers a fully functional cockpit to analyze the data efficiently.

> **SAP Note 1916**
> For more information about table logging, see SAP Note 1916 (Logging of Table Changes in R/3).

Performance Impact

Table change logging shouldn't have a performance impact if you only log customizing tables. Although SAP delivers many tables with table logging activated, those tables usually contain little data that rarely changes. Avoid logging for master data and transaction data tables because those tables are subject to mass changes and hence would have a negative impact on the system performance. For custom tables, you can define whether you want to activate table logging or not.

If you experience negative performance after activating table logging, you can find out which tables log the most amount of data. In Transaction SCU3, you can validate the table logging via the menu path **Administration • Number of Logs (Selection)**. In the selection screen (Figure 11.25), leave the **Table Name** field empty and analyze the last month (or extend the time if required).

11

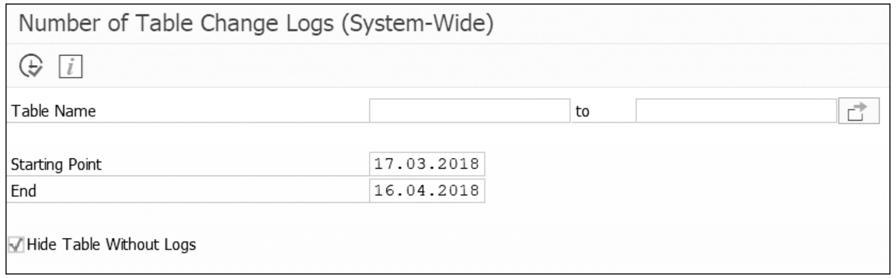


Figure 11.25 Number of Table Change Logs in Transaction SCU3

In the results screen, you can see the number of entries per table logged. For the example in Figure 11.25, table RFCDES logged 36 changes in the last 30 days, as shown in Figure 11.26. You can sort the number of logs in descending order to quickly get an indication of which tables might cause a performance issue.

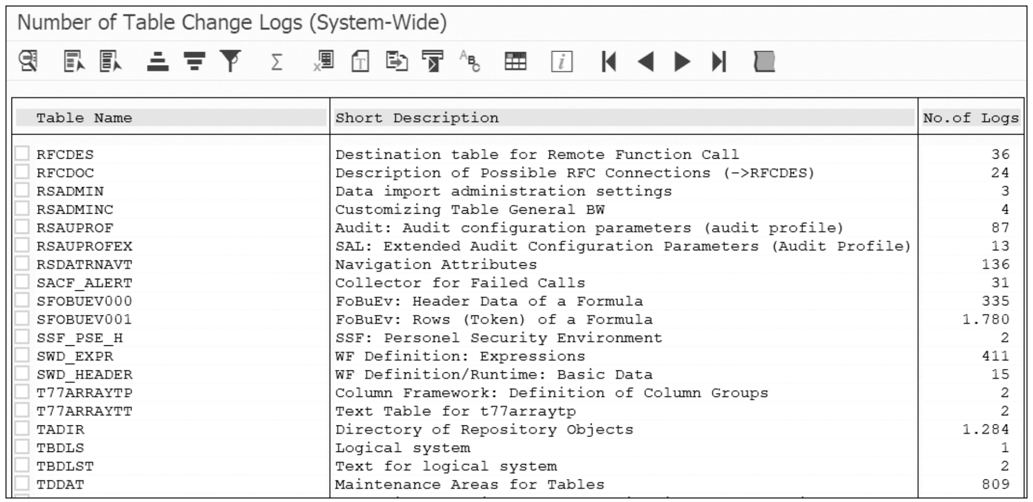


Table Name	Short Description	No.of Logs
RFCDES	Destination table for Remote Function Call	36
RFCDOC	Description of Possible RFC Connections (->RFCDES)	24
RSADMIN	Data import administration settings	3
RSADMINC	Customizing Table General BW	4
RSAUPROF	Audit: Audit configuration parameters (audit profile)	87
RSAUPROFEX	SAL: Extended Audit Configuration Parameters (Audit Profile)	13
RSDATRNAVT	Navigation Attributes	136
SACF_ALERT	Collector for Failed Calls	31
SFOBUEV000	FoBuEv: Header Data of a Formula	335
SFOBUEV001	FoBuEv: Rows (Token) of a Formula	1.780
SSF_PSE_H	SSF: Personel Security Environment	2
SWD_EXPR	WF Definition: Expressions	411
SWD_HEADER	WF Definition/Runtime: Basic Data	15
T77ARRAYTP	Column Framework: Definition of Column Groups	2
T77ARRAYTT	Text Table for t77arraytp	2
TADIR	Directory of Repository Objects	1.284
TBDLS	Logical system	1
TBDLST	Text for logical system	2
TDDAT	Maintenance Areas for Tables	809

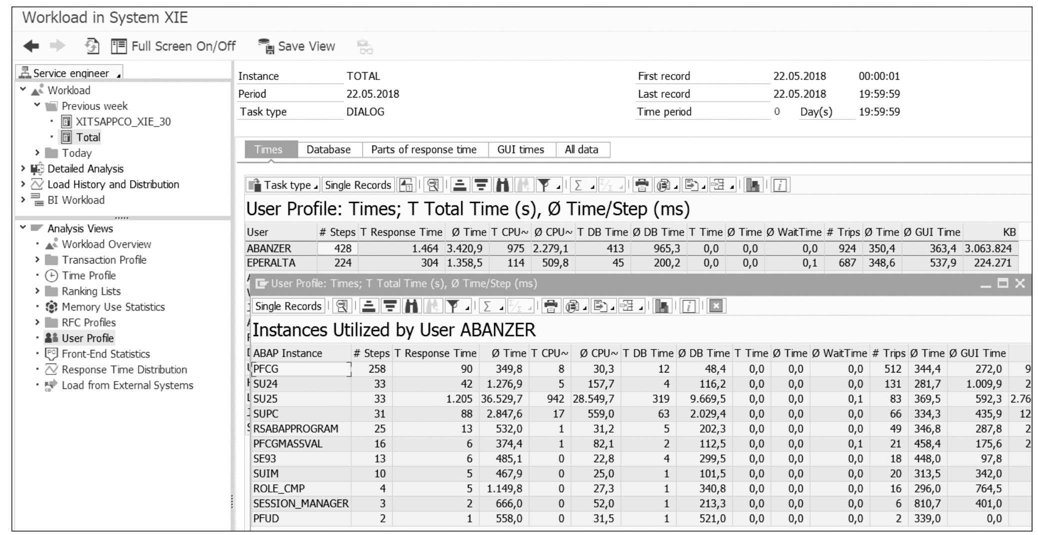
Figure 11.26 Result Screen of Number of Table Change Logs

Table logging shouldn't have an impact on your overall system performance and hence is a helpful feature to ensure the traceability of changes to customizing and other important tables in your SAP system.

11.3.4 Workload Monitor

The Workload Monitor lets you analyze system statistics in the SAP system. You can report on different task types like background processing, dialog processing, update processing, ALE, RFC, and so on. You will also see detailed information on CPU time, number of changes to the database, number of users that use the system, and so on. You can start the Workload Monitor in Transaction ST03N.

Apart from all the analysis capabilities to check the workload of your system, the Transaction ST03N trace contains information that might be helpful for auditing purposes. In Transaction ST03N, you can analyze the activity of a user and reproduce the actions a user has executed in the system. In the user profile, you can see all the users in a certain time frame and details of the actions they performed. In Figure 11.27, you can see that user ABANZER executed several transactions (e.g., Transactions RSAU_ADMIN, RSAU_READ_LOG, and so on). You can also see how many dialog steps were executed along with the details of average response times.



User	# Steps	T Response Time	Ø Time	T CPU~	Ø CPU~	T DB Time	Ø DB Time	T Time	Ø Time	Ø WaitTime	# Trips	Ø Time	Ø GUI Time	KB
ABANZER	428	1.464	3.420,9	975	2.279,1	413	965,3	0,0	0,0	0,0	924	350,4	363,4	3.063.824
EPERALTA	224	304	1.358,5	114	509,8	45	200,2	0,0	0,0	0,1	687	348,6	537,9	224.271

ABAP Instance	# Steps	T Response Time	Ø Time	T CPU~	Ø CPU~	T DB Time	Ø DB Time	T Time	Ø Time	Ø WaitTime	# Trips	Ø Time	Ø GUI Time
PFCG	258	90	349,8	8	30,3	12	48,4	0,0	0,0	0,0	512	344,4	272,0
SU24	33	42	1.276,9	5	157,7	4	116,2	0,0	0,0	0,0	131	281,7	1.009,9
SU25	33	1.205	36.529,7	942	28.549,7	319	9.669,5	0,0	0,0	0,1	83	369,5	592,3
SUPC	31	88	2.847,6	17	559,0	63	2.029,4	0,0	0,0	0,0	66	334,3	435,9
RSABAPPROGRAM	25	13	532,0	1	31,2	5	202,3	0,0	0,0	0,0	49	346,8	287,8
PFCGMASSVAL	16	6	374,4	1	82,1	2	112,5	0,0	0,0	0,1	21	458,4	175,6
SE93	13	6	485,1	0	22,8	4	299,5	0,0	0,0	0,0	18	448,0	97,8
SUIM	10	5	467,9	0	25,0	1	101,5	0,0	0,0	0,0	20	313,5	342,0
ROLE_CMP	4	5	1.149,8	0	27,3	1	340,8	0,0	0,0	0,0	16	296,0	764,5
SESSION_MANAGER	3	2	666,0	0	52,0	1	213,3	0,0	0,0	0,0	6	810,7	401,0
PFUD	2	1	558,0	0	31,5	1	521,0	0,0	0,0	0,0	2	339,0	0,0

Figure 11.27 Workload Monitor for Specific User Profile in Transaction ST03N

The workload is deactivated by default as it increases the chances for performance implications. Therefore, we recommend activating it temporarily for specific analysis. Before activation, make sure that you adhere to the laws and regulations in your territory.

> **Warning**
> Analyzing user activities may not be permitted based on your area of operation. Also, personal data protection regulations like GDPR may prohibit the use of such information.

11.3.5 Read Access Logging

Read access logging (RAL) is a tool to monitor and record the read access to sensitive and classified data in your SAP system. The type of data that you want to monitor can be categorized as sensitive by law or by internal or external company policies. In the context of the GDPR, companies must comply with the regulations and adhere to standards about data privacy.

With the RAL framework, you can comply with the regulations because you always know who accessed which data from where and when. Also, in case of a security breach or a leak of information, you can report not only who had access to the data from an authorization standpoint but also who accessed the data through the logging.

The RAL framework works with different types of channels when a user is accessing the data. *Channels* are the way the data leaves or enters the system (e.g., through SAP GUI). On the UI side, the RAL framework works with Dynpro (logging of Dynpro UI elements and ALV grids) and Web Dynpro (logging of context-bound UI elements).

It also works with APIs such as the following:

- **Remote Function Calls (sRFC, aRFC, tRFC, qRFC, bgRFC)**
  Logging of server- and client-side RFC-based communication
- **Web services**
  Logging of consumer- and provider-side web service communications
- **OData channels**
  Logging of data consumed by SAP Fiori applications through OData services

> **Further Information**
> For more information about the OData channels for SAP Fiori applications, you can check SAP Note 2182094 (Read Access Logging in SAP Gateway).

The configuration and monitoring of the RAL is done in Transaction SRALMANAGER (Figure 11.28).

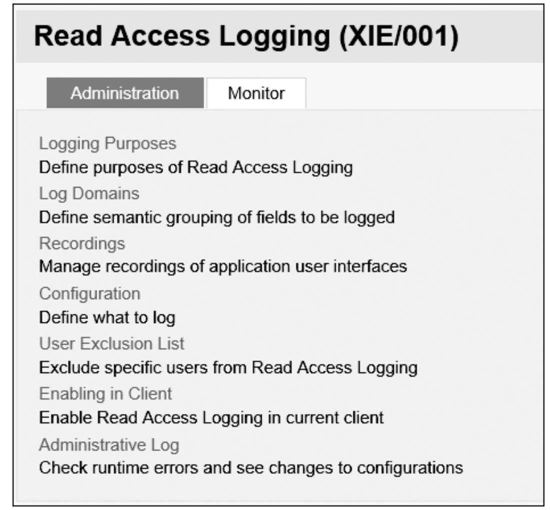


Figure 11.28 Read Access Logging Initial Screen

The configuration of RAL requires five steps, which are represented in the Web Dynpro application that starts with Transaction SRALMANAGER:

1. You have to identify and determine under what circumstances the RAL will log what type of data. For example, in view of GDPR, you have to protect personal information of your employees. Therefore, you have to monitor and protect transactions and tables that contain personal information, like Transaction SU01 (User Master Records), table `USR02` (User Master Records), and so on.
2. In the second step, you have to define the purpose of the logging, which allows you to group certain requirements. You can freely define a name for the logging purpose. The logging purpose is used to organize the data in the context of a specific use case, such as for GDPR.
3. In the third step, you have to define the channels that you want to monitor. Common channels are Web Dynpro, RFCs, and so on.
4. Once you have the channels defined, you define the log domains. The log domains group semantically similar or related fields. For example, in the Basis area, an "account" is different than the "account" in the banking application. Therefore, you want to classify similar content into log domains.
5. Finally, you define the conditions that must be met for the application to log the data—for example, which fields are being recorded and whether the access is recorded only or the content of the data is recorded as well.

For simplified operation of the RAL, you can define an exclusion list of users that won't be logged. A common scenario is to exclude batch job users that perform multiple reads, which would lead to a significant number of logs.

Once the configuration has been activated successfully, you can start to monitor the log entries in the Web Dynpro application. To review the logs, you can go to **Read Access Log** in the **Monitor** tab. You can search channel-specific, date-specific, or user name-specific logs.

11.3.6 User Information System

The User Information System is one of the main tools required for both internal and external audits. This tool is a directory for several programs that facilitate the retrieval of information required for an audit. Most of the tools focus on users and authorizations. However, the AIS also contains a powerful change document feature. Each function is organized by its type in the menu tree and can be launched by double-clicking the **Execute** button to the left of the function name.

As an example, let's look up users with critical authorization combinations. This is a common report used by auditors to satisfy audit controls. Proceed as follows:

1. Navigate to Transaction SUIM.

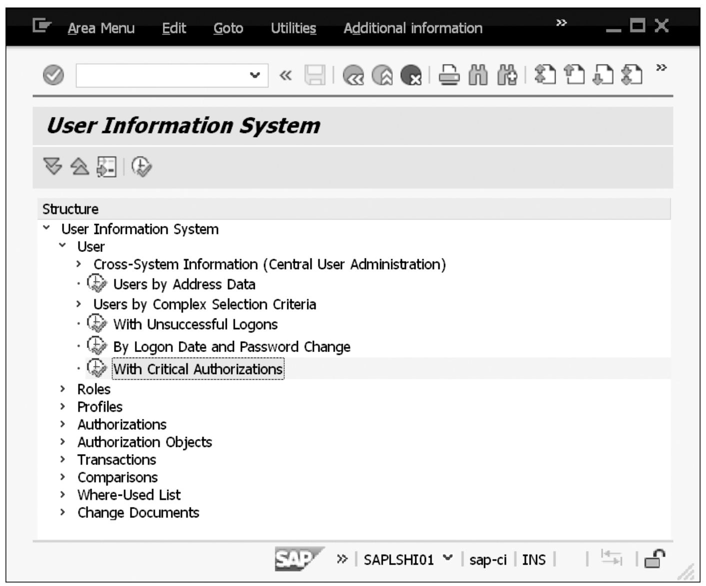


Figure 11.29 Transaction SUIM Main Screen

2. In the menu, click the **User** drop-down, then select **With Critical Authorizations** and click the **Execute** icon (Figure 11.29).

> **Direct Access**
>
> Alternatively, you can run report RSUSR008_009_NEW in Transaction SA38.

3. Next, choose the **For Critical Authorizations** radio button in the **Variant Name** box. For the variant name, choose the predelivered SAP_RSUSR009 variant (Figure 11.30).

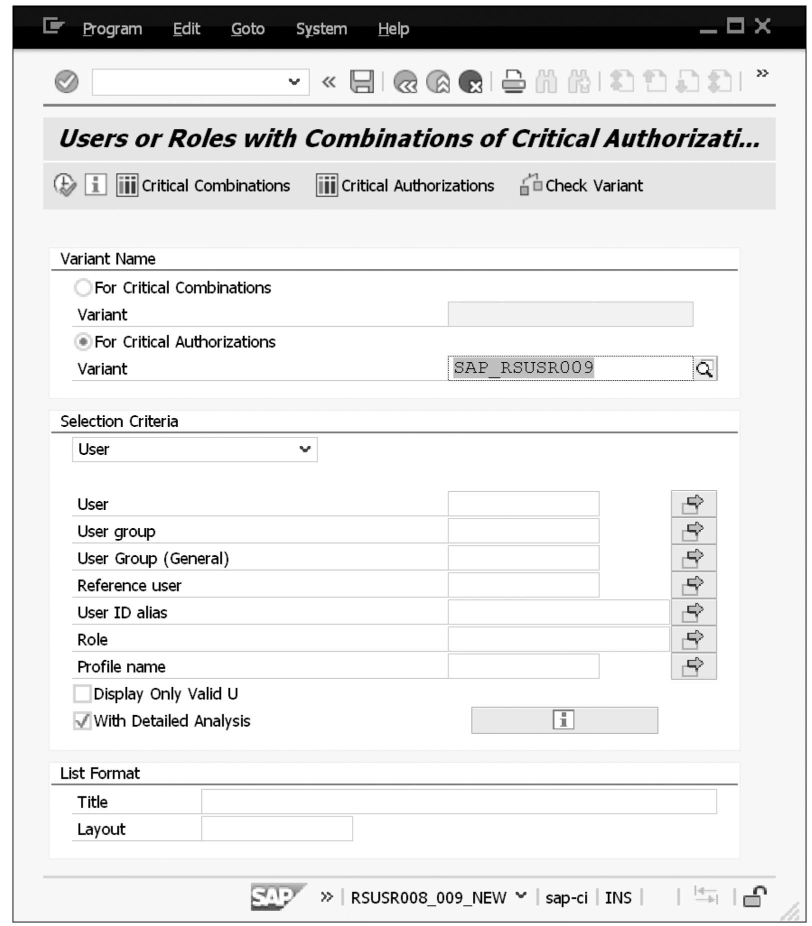


Figure 11.30 Critical Authorizations Selection Screen

> **Define Your Own Critical Authorizations**
>
> You can also define a list of critical authorizations. You may receive a list of critical authorizations or transaction codes from your internal auditor, external auditor, or functional business analysts. You may need to come up with this list on your own. A good starting point is to use the SAP delivered variant, SAP_RSUSR009, but be sure to adjust it for your auditing use.

4. Click the **Execute** button.

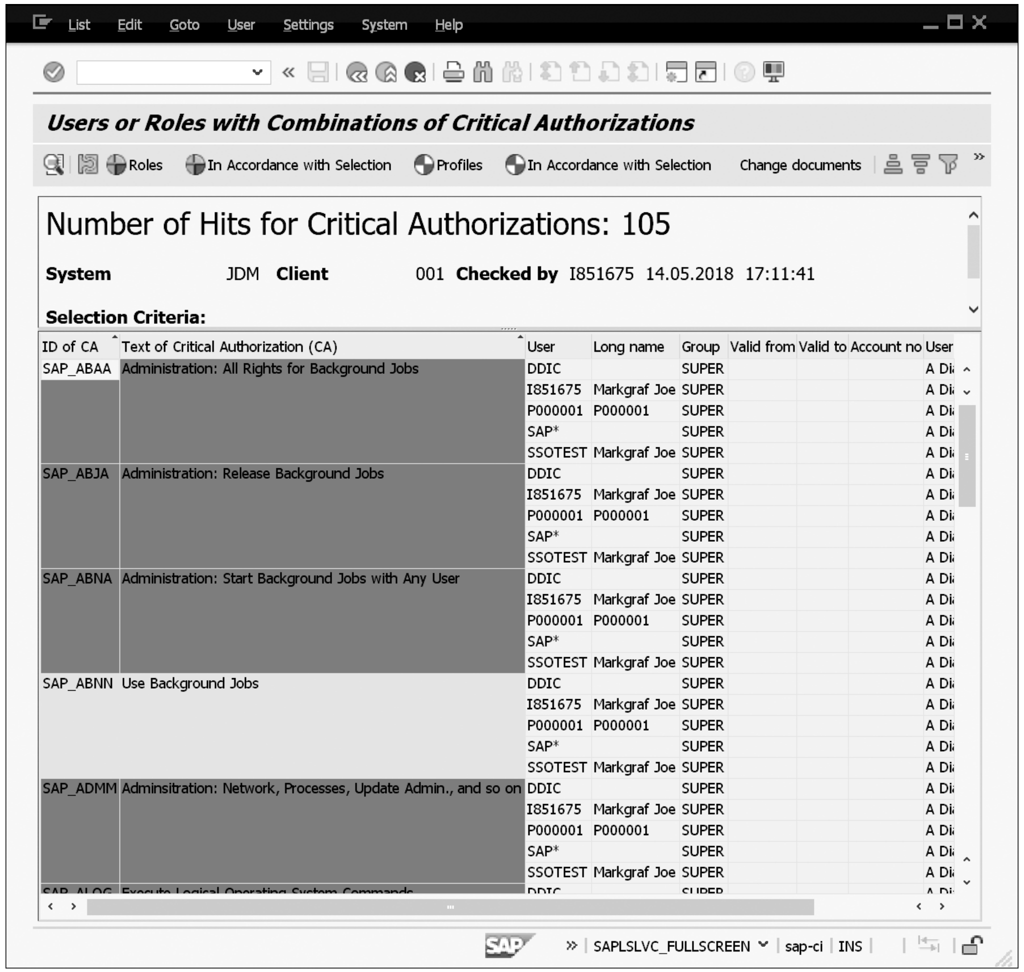


Figure 11.31 Report Generated with Critical Authorizations

The system will return a list of critical authorizations (Figure 11.31) that each user has in your system. If you have many super users, or administrators, this list could be in the thousands or tens of thousands. A review of this list and its users is done often, with the appropriateness of each user's access reviewed by either internal or external auditors.

11.4 Summary

In this chapter, you learned about internal and external audits and their purpose in an organization. You learned about auditing tools like security audit logging, the system log, table logging, the Workload Monitor, and Read Access Logging. Finally, you learned about the User Information System and how to use it to find users with critical authorizations.

In the next chapter, you'll learn about how to secure network communications to and from your SAP NetWeaver AS ABAP system. This is an important subject for a security administrator because most attacks against an SAP system use the network as an attack vector.

Contents

Index

D

E

F

G

H

I

K

L

M

N

O

P

R

S

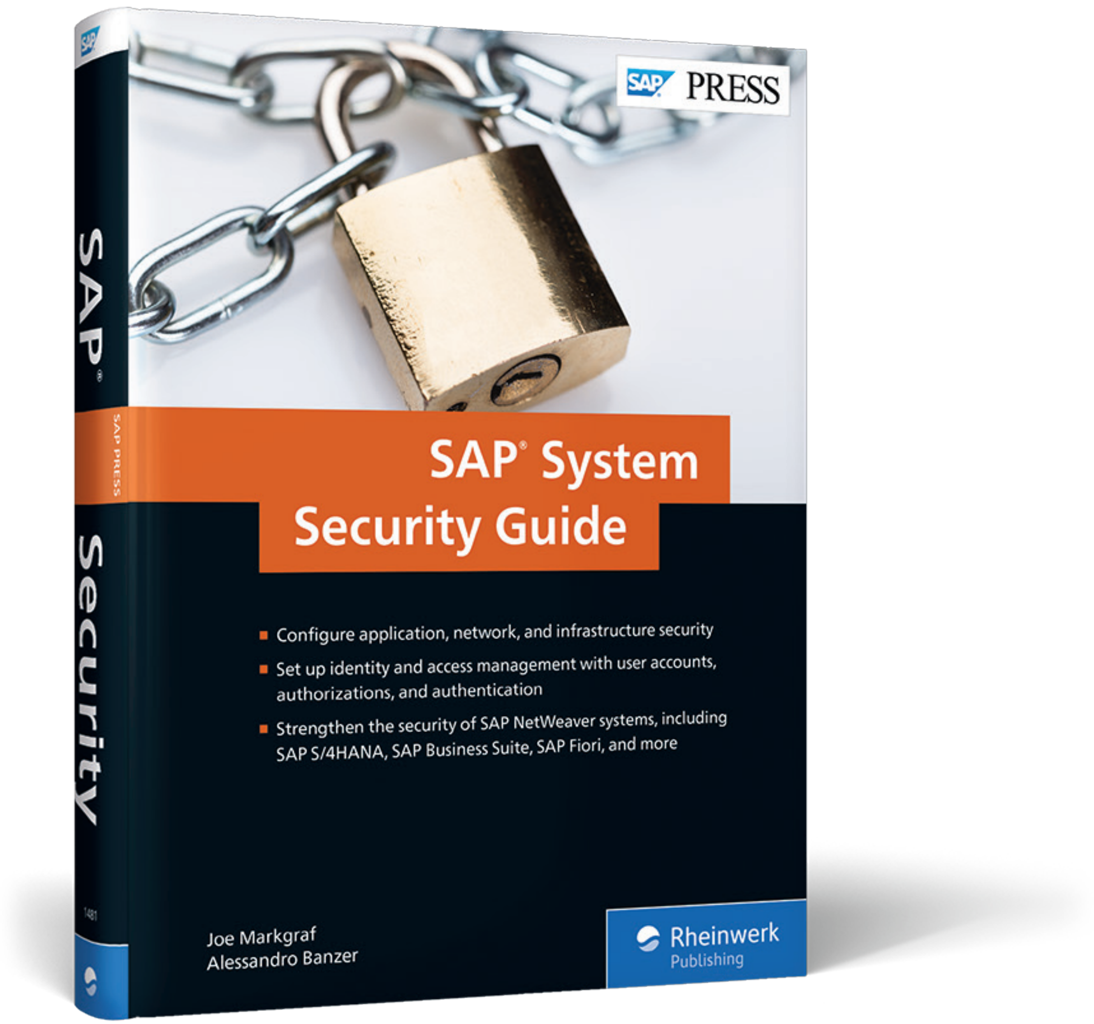
T

U

V

SAP PRESS **First-hand knowledge.**

Rheinwerk Publishing



Joe Markgraf is a senior cloud architect and advisor for SAP HANA Enterprise Cloud at SAP. Before joining SAP he worked as a Basis and security administrator, contributing to both small- and large-scale SAP system implementations. He holds a business degree with a focus on information system management from Oregon State University. He enjoys playing vintage video games and shooting sports with his family in Washington State.

Alessandro Banzer is the Chief Executive Officer of Xiting, LLC. He has worked in information technology since 2004, specializing in SAP in 2009. Since then, Alessandro has been involved with global SAP projects in various roles. Alessandro is an active contributor and moderator in the Governance, Risk, and Compliance space on SAP Community, as well as a speaker at SAPPHIRE, ASUG, SAPInsider, and other SAP-related events. He holds a degree in business information technology, as well as an executive master of business administration from Hult International Business School in London, UK.

Joe Markgraf and Alessandro Banzer

SAP System Security Guide

574 Pages, 2018, $79.95

ISBN 978-1-4932-1481-5

www.sap-press.com/4307