SAP GUI Scripting Security Guide

SAP GUI for Windows
Release 7.20

Contents

Copyright

1. Introduction

The SAP GUI Scripting API is an automation interface that enhances the capabilities of SAP GUI for Windows. Using this interface, end users may automate repetitive tasks by recording and running macro-like scripts. Administrators and developers on the other hand may build tools for server-side application testing or client-side application integration.

From the SAP server's point of view there is no difference between SAP GUI communication generated by a script and SAP GUI communication generated by a user. For this reason a script has the same rights to run SAP transactions and enter data as the user starting it. In addition, the same data verification rules are applied to data entered by a user and data entered by a script.

However, just as a person might make mistakes that cannot be detected by a verification rule, an error in a script may cause bad data to be entered into the system without being detected immediately. A script runs significantly faster than manual interaction with a system, though, and it may also run unattended. It is therefore likely that a bad script can generate more bad data than a user before the mistake is detected.

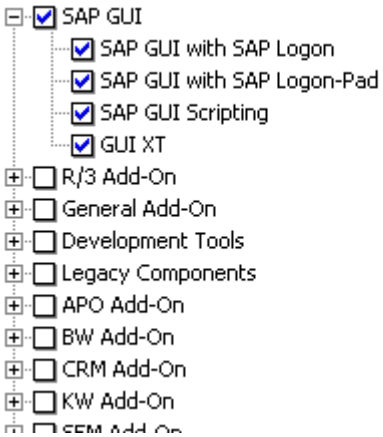
The main focus of the SAP GUI Scripting security considerations is to prevent scripts from being run accidentally or unbeknownst to the user. This also prevents user interaction from being recorded without the user's consent.

In the following chapters we will describe the available security measures that are implemented on different levels of the system architecture.

2. Installation

On any client PC, SAP GUI Scripting is only available if it has been installed. The SAP GUI Scripting support is included in the SAP GUI installation per default. However, an administrator can prevent SAP GUI Scripting from being installed. Using SAPAdmin, the administrator can create an installation package without Scripting and then prevent users from selecting components manually. Installation packages can be assigned to single users or to a group of users so that an administrator can easily define who will be able to use SAP GUI Scripting and who will not.

If a user has the right to select components himself he can exclude Scripting by not selecting the entry in the list of components.



Local administrator privileges are required to run the SAP GUI installation, unless a central installation server is used. A non-admin user can therefore not enable SAP GUI Scripting even if he has access to a SAP GUI installation medium.

It is possible for the administrator to disable SAP GUI Scripting on certain client machines even after it has been installed. All that needs to be done is to set the registry key HKLM\SOFTWARE\SAP\SAPGUI Front\SAP Frontend Server\Security\UserScripting to 0. This will disable SAP GUI Scripting and cannot be overridden by non-admin users.

SAP GUI Scripting makes use of an ActiveX object called “sapfewse.ocx”. Even though SAP applies strict security policies ActiveX objects may be vulnerable to attacks. Therefore SAP has decided to set the so-called “killbit” for the SAP GUI Scripting ActiveX object (see SAP Note 1261706 for more information). Setting the killbit does not have any effect on SAP GUI Scripting at all except for scenarios where sapfewse.ocx is called directly from a web page.

3. Protecting critical SAP systems

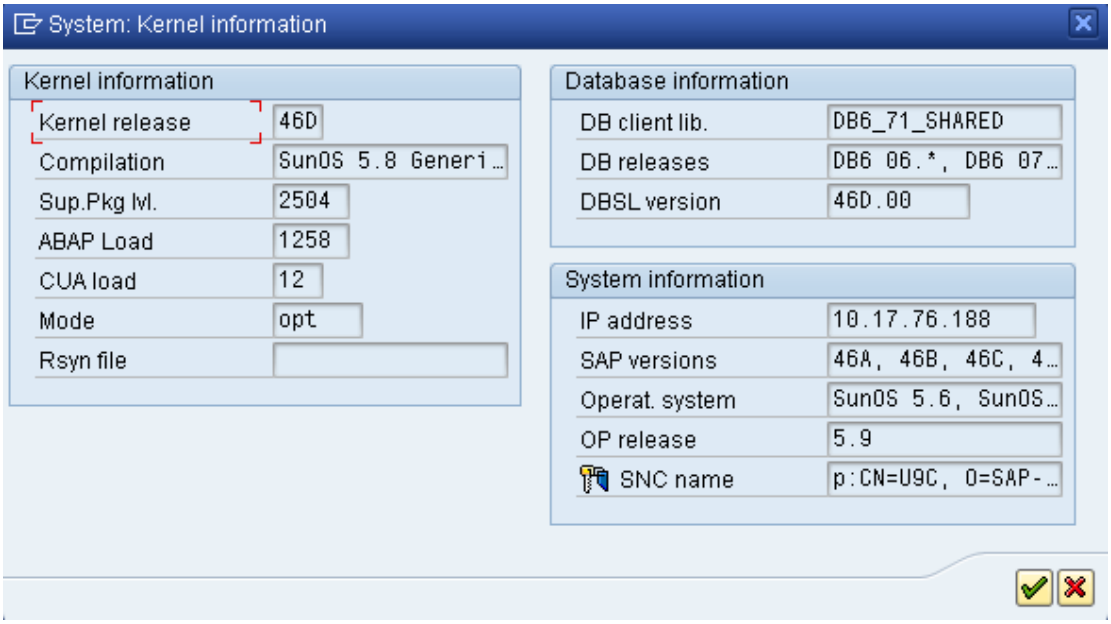
By default SAP GUI Scripting is disabled on any given SAP system. The administrator has to enable the support by setting the profile parameter *sapgui/user_scripting* to “TRUE” on the application server. That way an administrator can enable scripting either for all users of a given system by setting the parameter on all application servers or for a certain group of users by setting the parameter only on certain servers, which may have special access restrictions.

On the other hand it is possible to completely prevent scripts from being run against a specific SAP system. This might be desirable to protect mission critical data from being corrupted or downloaded.

The profile parameter requires the following kernel patch levels and SAP support packages:

- 6.20 and following: Standard
- 6.10: Kernel 6.10 patch level 360, support package SAPKB61012
- 4.6D: Kernel 4.6D patch level 948, support package SAPKB46D17
- 4.6C: Kernel 4.6D patch level 948, support package SAPKB46C29
- 4.6B: Kernel 4.6D patch level 948, support package SAPKB46B37
- 4.5B: Kernel 4.5B patch level 753, support package SAPKH45B49
- 4.0B: Kernel 4.0B patch level 903, support package SAPKH40B71.
- 3.1I: Kernel 3.1I patch level 650, support package SAPKH31I96.

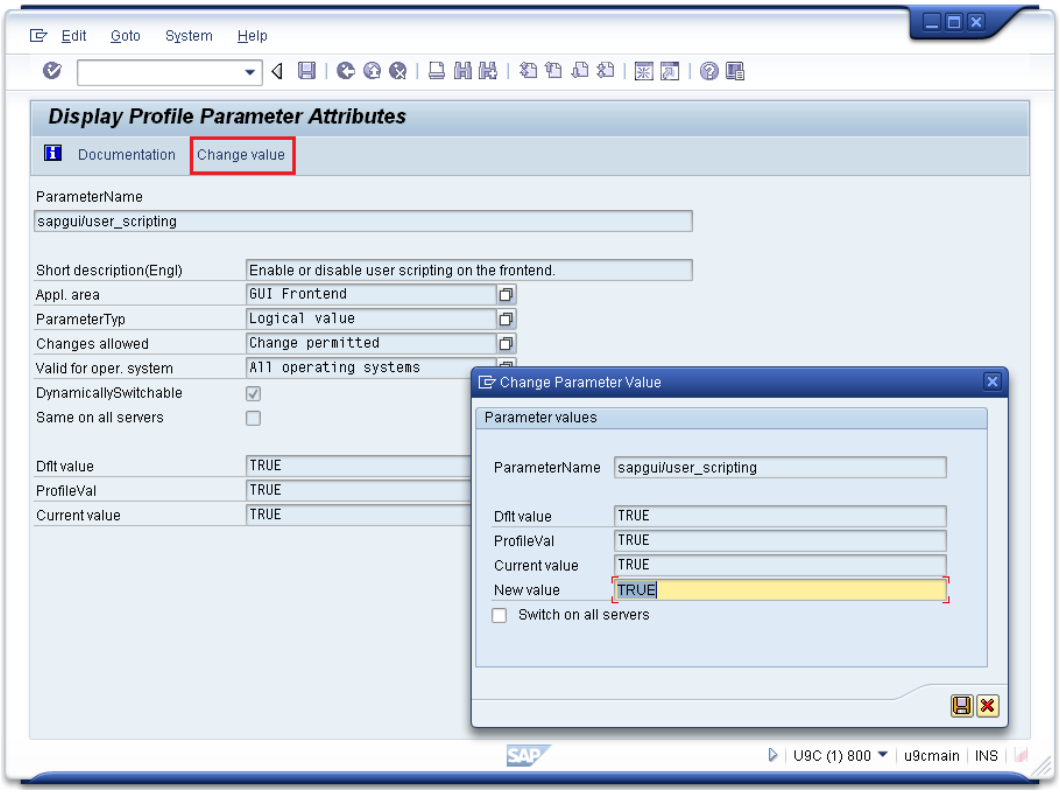
The currently installed kernel patch level can be checked on the status dialog. Select the dialog from the system menu and open the kernel information window by clicking on the *Other kernel info* button on the toolbar. The patch level is displayed in the *Sup. Pkg. Lvl.* field.



To check the support package level of an SAP system, go to transaction *SPAM* and press the *Package level* button. For SAP system releases 4.5B and below check the line *SAP_APPL*, for higher versions the *SAP_BASIS* line is relevant.

For the following systems the parameter can be set dynamically using transaction rz11 instead of changing the profile file and re-starting the application server:

- 6.20 and following
- 6.10: Kernel 6.10 patch level 391
- 4.6D: Kernel 4.6D patch level 972
- 4.6C: Kernel 4.6D patch level 972
- 4.6B: Kernel 4.6D patch level 972



To set the parameter run transaction rz11, enter the parameter name *sapgui/user_scripting* and press the *Display* button. If the parameter is not found then the support package level of the system does not suffice. On the following screen the *Current value* entry should be *TRUE*. If it is displayed as *FALSE*, press the *Change value* button on the toolbar, set the value to *TRUE* and save it. The value **must** be entered in uppercase; otherwise it will be interpreted as *FALSE*.

If the SAP system has several application servers and uses load balancing you may want to set the *Switch on all servers* check box. Otherwise the parameter is only set when you log into the current application server.

After saving the value the *Current value* should change to *TRUE*. If the value does not change then make sure that the appropriate kernel patch has been installed.

The scripting support will then be enabled **the next time** you log into the server.

See SAP Note 480149 for additional information.

4. Modes for server side protection

The profile parameter described in the previous chapter controls the availability of SAP GUI Scripting in an all-or-nothing kind of way. Some users have asked for a more fine grained approach. This would allow them to enable only those features of SAP GUI Scripting that are required for their specific application.

In response to these requests we have added two additional profile parameters that modify the behavior of the *sapgui/user_scripting* profile parameter.

sapgui/user_scripting_disable_recording

This parameter disables all SAP GUI Scripting events for the system on which it is set. It is still possible to run previously recorded or written scripts. However, it is not possible to record new scripts or log any other type of information in response to SAP GUI Scripting events.

sapgui/user_scripting_set_readonly

In SAP GUI Scripting's read only mode only a subset of the API can be used from a script. This comprises read access to properties and calling read only functions.

Please note that the read only restriction applies to the state of the SAP GUI session on the server. This implies that you may not execute any call which changes the data stream sent to the server, even if no actual database update is attempted.

Both of these parameters will only take effect after SAP GUI Scripting has been enabled using the parameter *sapgui/user_scripting*.

The installation requirements for these new parameters are as follows:

SAP GUI for Windows:

- 6.40 and following
- 6.20 Patch Level 42 and following

SAP System:

- 6.40 and following
- 6.20: Kernel 6.20 patch level 1223, support package SAPKB62037
- 4.6C: Kernel 4.6D patch level 1698, support package SAPKB46C47

5. Server side protection per user

In some cases it is not possible to enable SAP GUI Scripting for lack of a dedicated application server. This implies that users who are allowed to use SAP GUI Scripting work on the same server as others, so the support cannot be enabled for the server.

The problem can be solved by setting the rights to run SAP GUI Scripting per user. As in the previous chapter the profile parameter *sapgui/user_scripting* needs to be set to "TRUE". The new profile parameter *sapgui/user_scripting_per_user* allows the administrator then to enable SAP GUI Scripting support for specific users. Unless the administrator explicitly changes the value, this parameter is set to "FALSE". If the profile parameter is set to "TRUE" the following happens:

- On the login screen SAP GUI Scripting is available for every user.
- After login SAP GUI Scripting only remains available for those users that have the authorization for the *Execute(16)* action of the authorization object *S_SCR* in class *BC_A*.

The following software versions are required to use this new functionality:

SAP GUI for Windows

- 6.40: Patch 22 and following
- 7.10 and following

SAP System

- 6.40: Kernel 6.40 patch level 159, support package SAPKB64020
- 7.00: Kernel 7.00 patch level 87, support package SAPKB70011
- 7.10 and following.

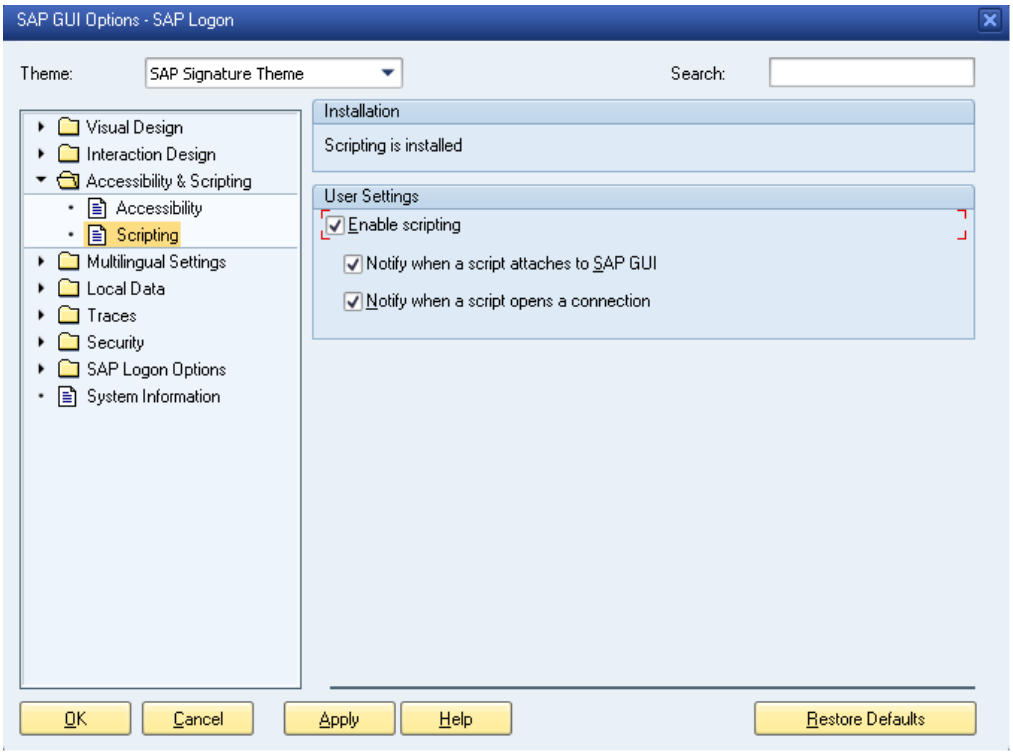
Please note that the update for SAP GUI is mandatory for all users working on the respective server. If the profile parameter *sapgui/user_scripting* is set to "TRUE" on the system, users with older versions of SAP GUI may in rare cases be able to run scripts on the system irrespective of the value of the profile parameter *sapgui/user_scripting_per_user* or the authority object.

6. Protection on the end user level

If the administrator has enabled scripting for end users, a given user can still disable it by setting the registry value *UserScripting* in key *HKCU\SOFTWARE\SAP\SAPGUI Front\SAP Frontend Server\Security* to 0. To enable scripting the user can set the key back to 1, which is also the default.

In addition a user may want to be notified whenever a script attempts to access the running SAP GUI or whenever a script attempts to open a connection to an SAP system. This can be done by setting the registry values *WarnOnAttach* and/or *WarnOnConnection* in *HKCU\SOFTWARE\SAP\SAPGUI Front\SAP Frontend Server\Security* to 1, which is the default. Setting these values to 0 will prevent the message popup.

These registry settings can also be set through the SAP GUI options dialog that can be called from the Control Panel, SAP Logon (Pad) or a running SAP GUI session.



If the administrator or the user has chosen not to install Scripting then the 'User Settings' are not available.

Since in Accessibility scenarios SAP GUI Scripting is used it is recommended that users disable the notification regarding attaching scripts to SAP GUI before starting SAP GUI.

As of SAP GUI for Windows 6.40 the user is notified about a script's execution through an animated icon on SAP GUI's status bar:

While a script is running, the icon on the right is animated. The icon's tooltip will also notify you if SAP GUI Scripting is disabled for some reason.

As off patch level 13 of SAP GUI for Windows 6.40, you can also disable the scripting functionality to write information to the hard disk. This is done via the registry value *DisableWriteToDisk* under the following registry path: *HKEY_LOCAL_MACHINE\software\sap\SAPGUI Front\SAP Frontend Server\Scripting*.

Please note that on 64bit operating systems the value mentioned above needs to be created under *HKEY_LOCAL_MACHINE\software\Wow6432Node\sap\SAPGUI Front\SAP Frontend Server\Scripting*.

7. Notes for users

7.1 Windows Script Host and SAP GUI for Windows

Visual Basic Scripts can only be executed from the Windows desktop if the Windows Script Host is installed. However tools for automatic testing or client analysis such as eCATT will generally not use VBS.

Visual Basic Script run within the scope of the Windows Script Host is a very powerful language. It is capable of accessing both the file system and the registry and of executing arbitrary commands.

In the past the WSH has on some occasions been used by computer viruses encoded as VBS to attack a machine. Therefore SAP does not encourage users to install WSH.

7.2 Do not write passwords into scripts

A user may decide to store scripts locally. As a script can be used to start a connection, a user might try to automate the login process by writing his login data into the script. We strongly discourage users from doing so. Scripts are not encrypted and anybody having access to the file can read the login data.

8. Security Q&A

1. Can a script corrupt the SAP system's data?
   → **No**. All changes done from a script are subject to the same data validation rules as end user interaction.
2. Can a script influence the system performance?
   → **Yes**. A script executes significantly faster than an end user, and may therefore put more load onto the system.
3. Can a script access data for which the end user does not have the necessary privileges?
   → **No**. The script has only access to the data to which the end user has access rights.
4. Can a script export data that the end user could otherwise not export?
   → **Yes**. Even if the download of a list is not allowed, an end user can extract the data from SAP GUI. Of course, the end user could also create a screen shot instead. SAP GUI Scripting can only export data that is displayed on the screen.
5. Can a script record end user interaction with SAP GUI?
   → **Yes**. However, the end user will be notified about this, unless he disabled the notification.

6. Can a script record passwords?
   → **No**. Therefore, a script cannot be played back if the user running it does not have an account on the SAP system.
7. Can a script run in the background without the end user's knowledge?
   → **No**. The end user will be notified when the script starts, unless she disabled the notification. In addition, SAP GUI Scripting needs to display SAP GUI for running a script.
8. Can SAP GUI Scripting be used to corrupt the client PC?
   → **No**. The functionality of SAP GUI Scripting is limited to driving SAP GUI. However, if you use Visual Basic Script and the Windows Script Host to access the SAP GUI Scripting interface, the functionality of the VBS language or the Windows Script Host object model might very well be used to perform arbitrary operations on the client PC.