
# Microsoft SQL Server™ 2005 Database Engine Common Criteria Evaluation

Guidance Addendum / Installation / Startup

*Microsoft SQL Server Team*


| | |
|---|---|
| Author: | Roger French |
| Version: | 1.5 |
| Date: | 2008-09-26 |
| File Name: | MS_SQL_AGD_IGS_1.5 |



## Abstract

This document describes the User and Administrator Guidance and the procedures for Installation, Generation and Startup for the Common Criteria evaluation of the Database Engine of Microsoft SQL Server™ 2005, Enterprise Edition (English), Version 9.00.3068.00 (TOE)



## Keywords

CC, SQL, Common Criteria, AGD_USR, AGD_ADM, ADO_IGS


## Revision History

| Date | Version | Author | Edit |
|---|---|---|---|
| April, 6th 2006 | 0.1 | SQL Server Team | Initial version |
| April, 24th 2006 | 0.2 | SQL Server Team | First information for installation process |
| April, 28th 2006 | 0.3 | SQL Server Team | Added screenshots for installation process |
| May, 2nd 2006 | 0.4 | SQL Server Team | Added description for installation process |
| June, 2nd 2006 | 0.5 | SQL Server Team | Added chapter for SQL server trace |
| June, 8th 2006 | 0.6 | SQL Server Team | Added more information about trace |
| August, 10th 2006 | 0.7 | SQL Server Team | Added more information about trace |
| August, 26th 2006 | 0.75 | SQL Server Team | User event added for session establishment |
| October, 17th 2006 | 0.76 | SQL Server Team | Minor changes |
| May, 15th 2007 | 0.8 | SQL Server Team | General update after release of SP2 |
| July, 27th 2007 | 0.9 | SQL Server Team | Addressed comments from evaluation and AVA_MSU |
| Aug, 21st 2007 | 1.0 | SQL Server Team | Addressed minor comments. |
| Aug, 30th 2007 | 1.1 | SQL Server Team | Updates after review of test documentation |
| Dec, 11th 2007 | 1.2 | SQL Server Team | Minor updates |
| Mar, 28th 2007 | 1.33 | SQL Server Team | Minor updates |
| July, 14th 2008 | 1.35 | SQL Server Team | Minor updates |

| | | | |
|---|---|---|---|
| July, 16th 2008 | 1.4 | SQL Server Team | Final Version |
| Sep, 26th 2008 | 1.5 | SQL Server Team | Minor Updates |

This page intentionally left blank

# Table of Contents

## List of Tables

**Page**



## List of Figures

**Page**

# 1 Introduction

This is a mandatory input document required for the Common Criteria (CC) Evaluation of the Database Engine of Microsoft SQL Server 2005 Enterprise Edition (English) Version 9.00.3068.00 (comprising SP 2 and GDR 4)[1]. It addresses the assurance aspects AGD_USR.1, AGD_ADM.1 and ADO_IGS.1

**Chapter 2** of this document gives more details about the scope of the certification for and the assumptions, which have been made about the environment of the TOE.

**Chapter 3** of this document describes the steps for the installation process of the Database Engine of Microsoft SQL Server 2005 in its certified version.

**Chapter 4** introduces the concept of the SQL Server Books Online and provides the administrator and users with entry points for important aspects.

**Chapter 5** contains the important aspects of the guidance, which are specific to the certified version of the Database Engine of Microsoft SQL Server 2005.

Finally **chapter 6** introduces the concept and the important aspects of the trace mechanism of the Database Engine of Microsoft SQL Server 2005.

# 2 Scope of the evaluation

The Target of Evaluation (TOE), which has been addressed during this evaluation and certification process according to Common Criteria is one instance of the Database Engine of Microsoft SQL Server 2005. This database engine is the core component of the SQL Server Platform.

The TOE has been defined to be one instance of the database engine as it realizes the complete set of security functions as described in [ST, chapter 6.1] including:

- Security Management,
- Access Control,
- Identification and Authentication and
- Security Audit
- Session Handling

Note: In the context of the certified version a user of the TOE is only a user if he doesn't have any administrative permission. As soon as a user gets any administrative permission (e.g. because he becomes the owner of a database) he becomes an administrator.

Therefore, for the purposes of this guide, most of the users will be administrators (as they have administrative permissions for certain objects though their access to the TOE is still limited).

[1] Please note that the product will be referred to as Database Engine of Microsoft SQL Server 2005 or simply as TOE (= Target of Evaluation) for the rest of the document

Thus this document doesn't have a strong separation between user guidance and administrator guidance and the complete document should be read by all users of the TOE.

Additional information about the certification process and related documents can be obtained via [WEB].

The following chapter describes the assumptions, which have been made about the environment of the TOE during evaluation, and which therefore have to be addressed during the start-up and operation of the TOE. It further explains how these assumptions can be addressed.

## 2.1 Assumptions of Intended Environment

According to [ST] the following assumptions apply to the environment of use of the TOE.

| Assumption | Description |
|---|---|
| A.NO_EVIL | Administrators are non-hostile, appropriately trained, and follow all administrator guidance. |
| A.NO_GENERAL_PURPOSE | There are no general-purpose computing capabilities (e.g., compilers or user applications) available on DBMS servers, other than those services necessary for the operation, administration and support of the DBMS. |
| A.OS_PP_VALIDATED | The underlying OS has been validated against an NSA sponsored OS PP of at least Basic Robustness and the Operating System provides functionality for<br>• Identification and authentication of users,<br>• Access Control for Files,<br>• Domain Separation,<br>• Non Bypassability of TOE Security Functions,<br>• Time stamps and<br>• Cryptographic Functionality<br>• Residual Information Protection<br>• Audit Storage and Audit Review<br>• Hashing of passwords<br>The evaluation and certification of the underlying OS has been done on at least EAL 4 augmented by ALC_FLR.2. |
| A.PHYSICAL | It is assumed that appropriate physical security is provided for the value of the IT assets protected by the TOE and the value of the stored, processed, and transmitted information. |
| A.COMM | It is assumed that any communication path from and to |

| Assumption | Description |
|---|---|
| | the TOE is appropriately secured to avoid eavesdropping and manipulation. |

The following chapters provide more details about the requirements which result out of the several assumptions for the secure administration of the TOE.

### 2.1.1 Trained administrator

To address this assumption authorized administrators shall read and follow all guidance documentation.

Further this assumption requires appropriate training for the administrators.

It is assumed that the ‘sa’ has a commensurate level of knowledge to a MCDBA. Therefore, it is recommended that the 'sa' receive formal DBA training on the level of a MCDBA (or equivalent).

It is the responsibility of ‘sa’ to ensure that all other authorized administrators have sufficient knowledge and skills for the scope of their administrative permissions.

### 2.1.2 General purpose computing capabilities

The administrator of the TOE shall not install any general computing software on the machine where the TOE has been or will be installed other than those services necessary for the operation, administration and support of the DBMS. The installation of the TOE has to be performed on a virgin OS as described in chapter 3.2. Beside the installation of the TOE itself a SQL-client may be installed on the machine to be used for administration. Also the SQL Server Management Studio which ships together with SQL Sever can be used for administration (see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/toolref9/html/f289e978-14ca-46ef-9e61-e1fe5fd593be.htm]).

However it should be noted that Management Studio has not been within the scope of evaluation. Specifically all functionality of the Graphical User Interface has not been evaluated. Thus – within the context of the evaluation Management Studio should just be seen as any other T-SQL client and the administrator shall ensure that the version of the client he is using is up to date and does not introduce any potential vulnerabilities.

Further other parts of the SQL Server 2005 Platform may be installed as long as they are needed to support the administration and operation of the TOE.

For example the SQL Server Profiler may be used to review the audit logs (see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/3ad5f33d-559e-41a4-bde6-bb98792f7f1a.htm])

### 2.1.3 Validated Operating System

According to this assumption the underlying Operating System (which is Windows 2003 Server Enterprise Edition) has to be validated against an NSA sponsored OS PP of at least Basic Robustness.

As Windows 2003 Server Enterprise Edition has been evaluated and certified against an NSA sponsored PP (Controlled Access Protection Profile, V1.d, 8 October 1999) the resulting requirement is easily fulfilled as long as the administrator followed the installation guidance given in chapter 3 of this document. It has to be mentioned that the Common Criteria certificate of the underlying OS is only valid as long as all instructions from [WIN_CONF] and [WIN_ADMIN] are followed.

The fact that this certified Operating System is used further ensures that all requirements for the IT-environment as defined in [ST] are fulfilled as they all fall back to functionality, provided by the OS and defined in the corresponding ST for the OS. Only the requirement for Audit Review is not fulfilled by the Operating System. According to this requirement the environment of the TOE has to provide a means for the administrator to review the audit logs. To review the audit logs an additional tool has to be used (e.g. the SQL Server Profiler, which is part of the SQL Server Platform, see [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/3ad5f33d-559e-41a4-bde6-bb98792f7f1a.htm] for more details).

### 2.1.4 Physical Protection

It shall be ensured by the administrator that appropriate physical security is provided within the domain for the value of the IT assets protected by the TOE and the value of the stored, processed, and transmitted information.

It has to be mentioned that the maximum level of protection the TOE can provide for the user data which is stored in it depends on the physical security of the machine where the TOE is installed. With physical access to this machine an attacker could easily gain complete access to the user data which is stored in the database.

### 2.1.5 Protection of Communication

It should be mentioned that all connections from clients to the TOE are unencrypted per default. It falls into the responsibility of the administrator to provide an adequate level of security for the connections between the clients and the TOE. In principle this can be achieved in two different ways:

- By the use of a software which encrypts the connection
- By physical protection of the wire

The administrator should consider the value of the data which is stored in the TOE for the decision about the appropriate way to secure the connection.

# 3 Installation and Start-up Guide

This chapter provides instructions for a secure setup, installation, and configuration of the TOE. In addition, this chapter describes the prerequisites for the installation process.

## 3.1 Prerequisites

### 3.1.1 Hardware Prerequisites

According to [ST] a machine that meets at least following criteria has to be available:

- 600-megahertz (MHz) Pentium III-compatible or faster processor; 1-gigahertz (GHz) or faster processor recommended
- 512 megabytes (MB) of RAM or more; 1 gigabyte (GB) or more recommended
- Approximately 350 MB of available hard-disk space for the recommended installation
- Approximately 425 MB of additional available hard-disk space for SQL Server Books Online, SQL Server Mobile Books Online, and sample databases
- CD-ROM or DVD-ROM drive
- Super VGA (1,024x768) or higher-resolution video adapter and monitor
- Microsoft Mouse or compatible pointing device

Please note that additional disc space will be required for the recommended trace processes (Up to 10 GB in its default configuration).

## 3.2 Software Prerequisites

Before the installation of the TOE itself can start the Operating System has to be installed on the machine. The certified version of the Database Engine of Microsoft SQL Server 2005 shall be installed on the certified version of Windows 2003 Server Enterprise Edition.

Guidance on how to install Windows 2003 Server Enterprise Edition in its certified version and configuration can be found in [WIN_CONF] and [WIN_ADMIN].

Further it has to be noted that the certification according to Common Criteria is only valid for the Database Engine of Microsoft SQL Server 2005 Enterprise Edition (English) Version 9.00.3068.00.

## 3.3 SQL Server 2005 Installation

### 3.3.1 Downloading the additional information and SP2

The necessary guidance documentation for the TOE comprises:

- SQL Server Books Online Version as of February 2007 [AGD]

- SQL Server Guidance Addendum (this document)

All these guidance documents can be downloaded from [WEB].

Further scripts to put the database engine into the certified version and a script to install a trace process can be downloaded via this link.

### 3.3.2 Checking the integrity of the media

To verify the integrity of the installation media (and the related documentation) the administrator shall verify the following things:

- The box contains a so called Certificate of Authenticity (COA). The user should check the Security Characteristics of this COA to ensure that he received a genuine Microsoft product. More details about the security features can be found under http://www.microsoft.com/resources/howtotell/en/coa.mspx or in chapter 8.1.
- Additionally one should verify the integrity of the product by verifying the checksum. (see chapter 8.3)

### 3.3.3 Installing the product

This chapter introduces the installation process for SQL Server 2005 focused on the certified version. More general information about the installation process can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/instsql9/html/e5d061d0-d370-4140-9b19-b91b05da7271.htm]

The installation of the TOE starts after putting the first of two CD into the CD-ROM drive.


**Figure 1: Installation procedure (I)**

If the autorun mechanism of the Operating System has been disabled one has to execute the file "default.hta" in the root directory of the CD to start the installation process.

The welcome screen as seen in Figure 1 offers the following options:

- Review hardware and software requirements
- Read the release notes
- Install SQL Server Upgrade Advisor
- Install Server components, tools, Books Online and samples
- Run the SQL Native Client Installation Wizard
- Browse this CD
- Visit the SQL Server website
- Read the SQL Server license agreement

To start the installation of the TOE one has to choose the option Install Server components, tools, Books Online and samples

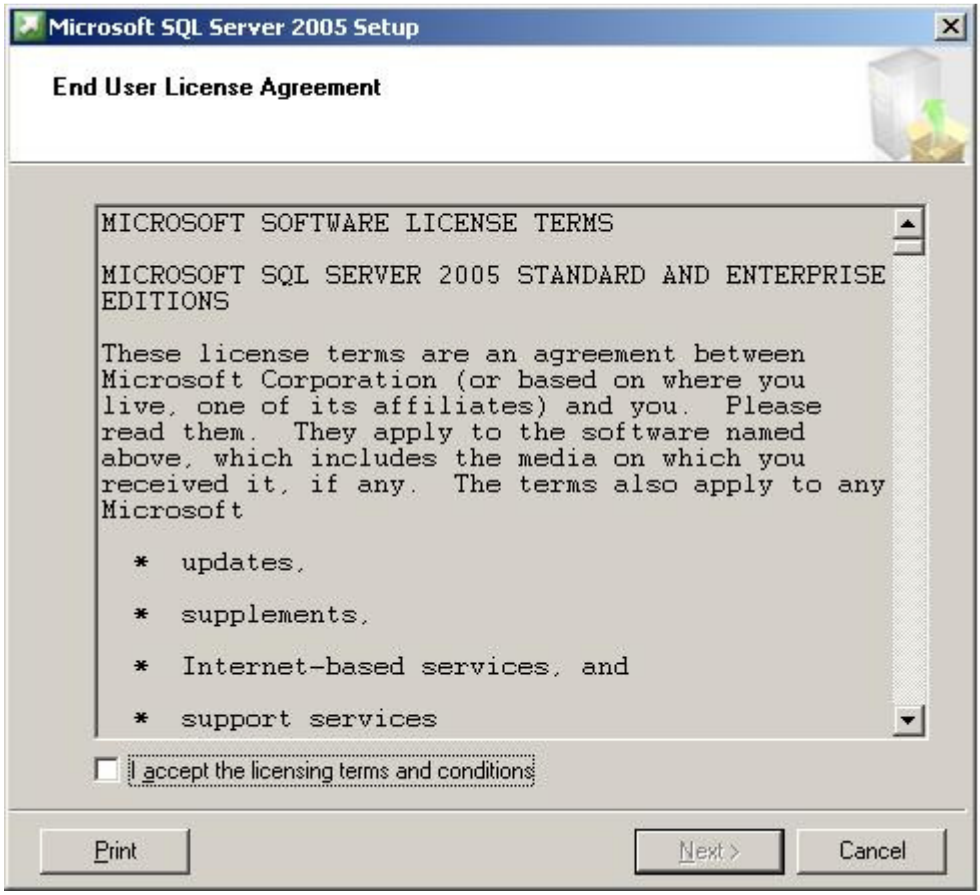


**Figure 2: Installation procedure (II)**

The End User License Agreement as seen in Figure 2 shows up. The user should carefully read this agreement. The tickbox “I accept the licensing terms and conditions” has to be ticked before it is possible to proceed with the installation process by clicking on the button “Next”.

After that a window as seen in the following figure will show up to inform the user that the setup process is going to install the .NET Framework 2.0, the Microsoft SQL Native Client and the Microsoft SQL Server 2005 setup support files.

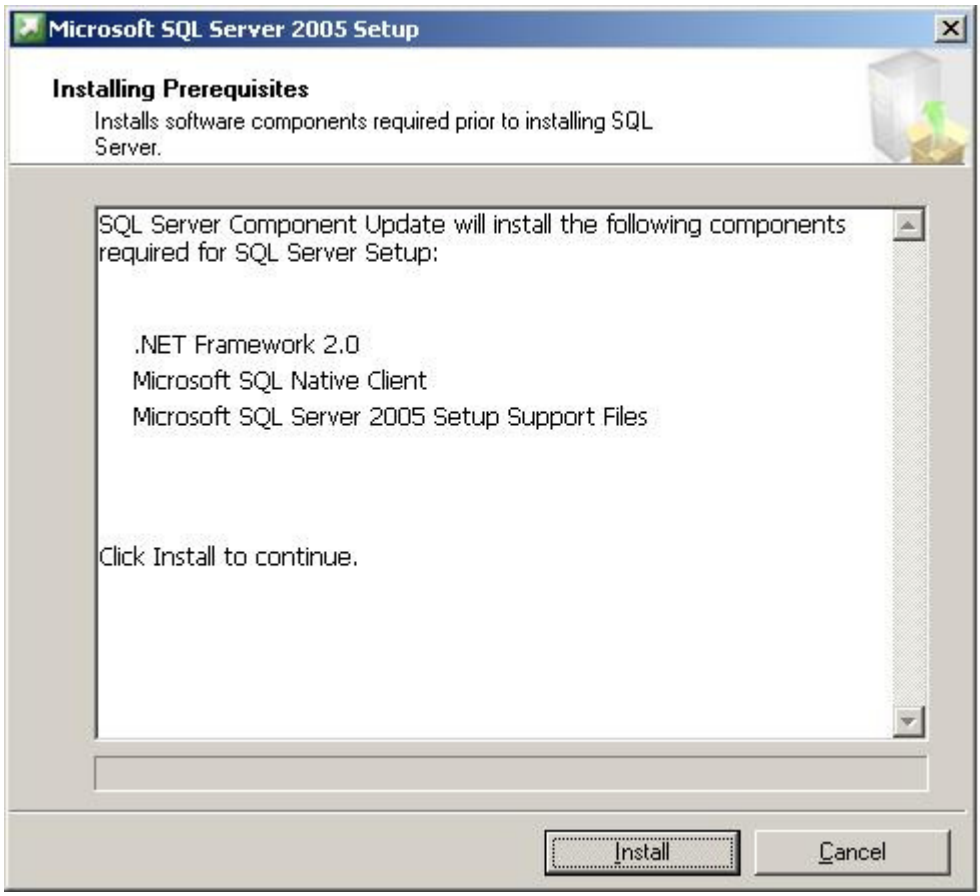


**Figure 3: Installation procedure (III)**

After the user clicked the button “Install” the first part of the installation procedure will start and the user will be informed about the progress as seen in the following two figures.

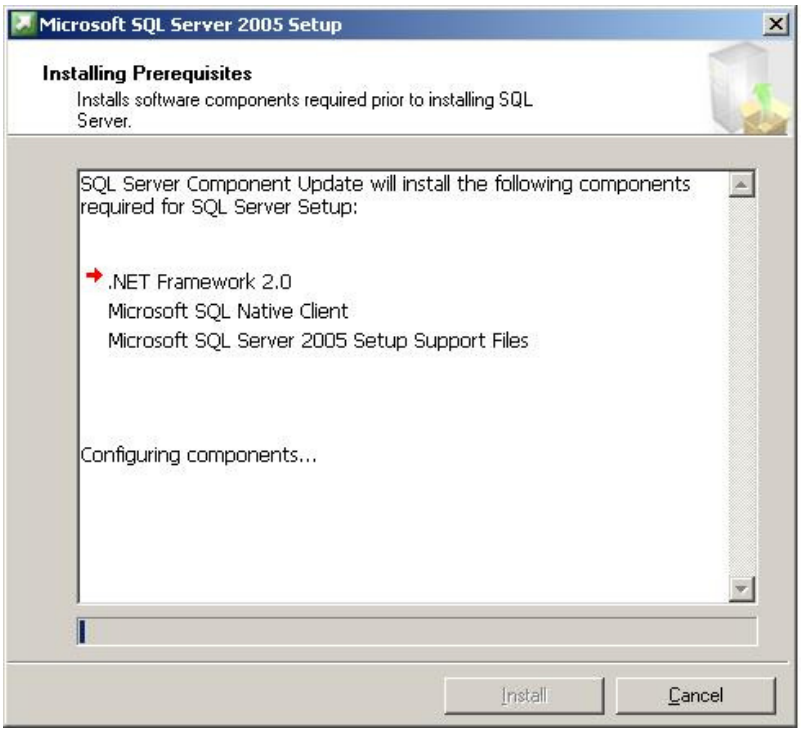


**Figure 4: Installation procedure (IV)**

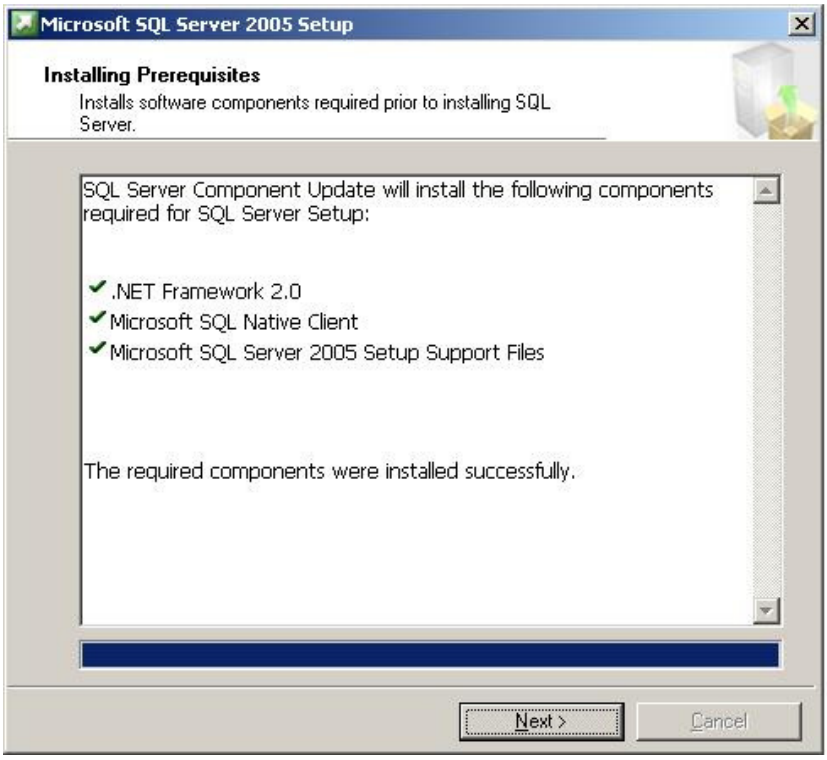


**Figure 5: Installation procedure (V)**

After the setup process shows the user that the required components have been successfully installed the user should click the button “Next” to start the SQL Server installation wizard which will guide the user thru the rest of the installation process.


**Figure 6: Installation procedure (VI)**

At the welcome screen of the SQL Server Installation Wizard the user only has the possibility to cancel the installation process or to proceed by clicking “Next”.

Then the SQL Server Installation Wizard will check a set of requirements to detect potential installation problems. If any non critical issues are detected the user will get a warning message as seen in the next figure. More information about the System Configuration Check can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/instsql9/html/8e712c15-6bfa-4d71-b303-9526101e5594.htm]. Only after the check has finished without any fatal errors the user has the possibility to proceed with the installation process by clicking the "Next" button.

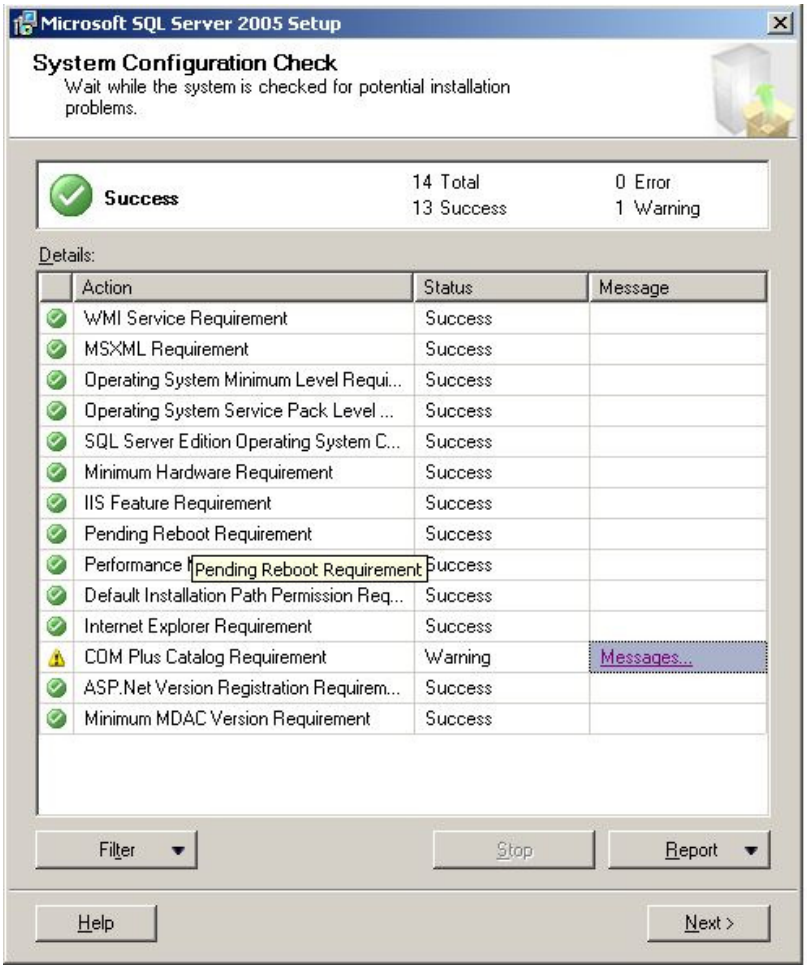


**Figure 7: Installation procedure (VII)**


**Figure 8: Installation procedure (IIX)**

In the next window as seen in Figure 8 the user has to type in his name, the name of the company (if any) and the 25 character Product Key. After that the user has the possibility to proceed with the installation process by clicking the "Next" button.

In the next window the user is asked, which components of the SQL Server 2005 platform should be installed. To install the Database Engine of Microsoft SQL Server 2005, which realizes all Security Functions as described in [ST] user shall choose to install the "SQL Server Database Services" as seen in Figure 9.

According to an assumption in [ST] no general-purpose computing capabilities must be available on DBMS servers other than those services necessary for the operation, administration and support of the DBMS.

However, the very strict configuration of the certified version of Windows may lead to errors when installing additional tools of the SQL Server platform. Thus it is recommended to install only the SQL Server Database Services.

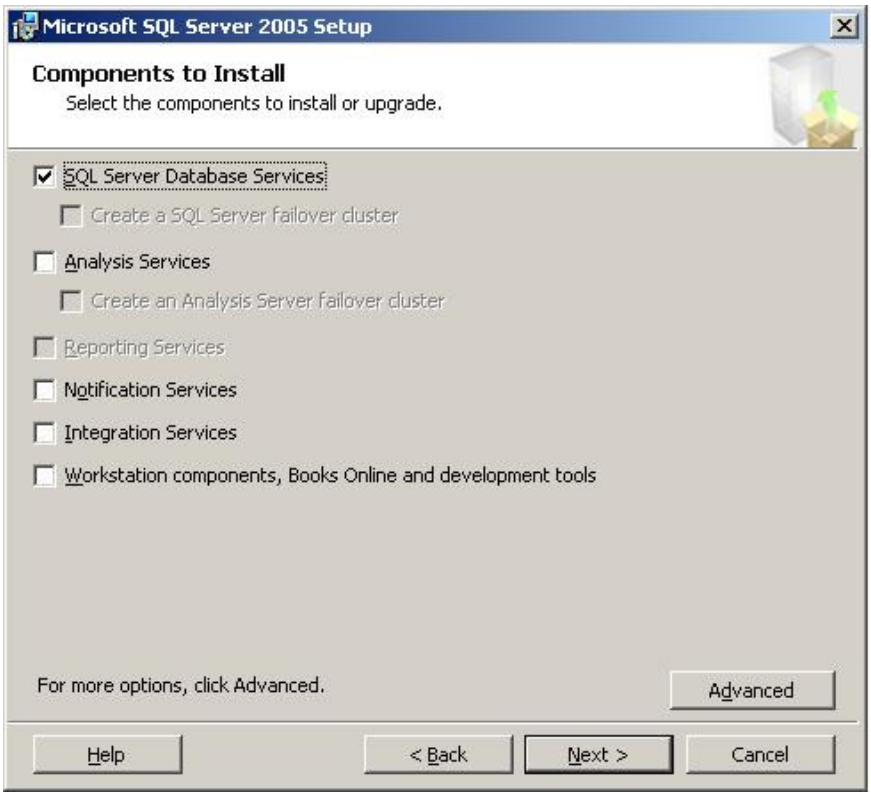


**Figure 9: Installation procedure (IX)**

Choosing only the "SQL Server Database Services" as seen in the previous figure will install one instance of the database engine of the TOE. This is the minimum installation.

The following table shows, what the different options in this dialog mean.

| Component group | To install these components and features |
|---|---|
| SQL Server Database Services | • SQL Server Database Engine includes the following technologies:<br>• The Database Engine is the core service for storing, processing, and securing data.<br>• Replication is a set of technologies for copying and distributing data and database objects from one database to another and then synchronizing between databases to maintain consistency.<br>• Full-Text Search provides functionality to issue full-text queries against plain character-based data in SQL Server tables.<br>• Tools for managing relational and XML data. |
| Analysis Services | Analysis Services includes the tools for creating and managing online analytical processing (OLAP) and data mining applications. |
| Reporting Services | Reporting Services includes server and client components for creating, managing, and deploying tabular, matrix, graphical, and free-form reports. |
| Notification Services | Notification Services is a platform for developing and deploying applications that send personalized, timely information to subscribers on a variety of devices.<br>Please note that the operation of Notification Services is not possible under the certified version of Windows 2003 as the Microsoft Distributed Transaction Coordinator (MSDTC) service is disabled under the evaluated configuration. |
| Integration Services | Integration Services is a set of graphical tools and programmable objects for moving, copying, and transforming data. |
| Workstation Components, Books Online, and Development Tools | Installs components for communication between clients and servers, including network libraries for DB-Library, OLEDB for OLAP, ODBC, ADODB, and ADOMD+.<br>**Management Tools**<br>• SQL Server Management Studio (SSMS), new in Microsoft SQL Server 2005, is an integrated environment for accessing, configuring, managing, administering, and developing all components of SQL Server. SSMS combines the features of Enterprise Manager, Query Analyzer, and Analysis Manager, included in previous releases of SQL Server, into a single environment that provides SQL Server access to developers and administrators of all skill levels.<br>• SQL Server Configuration Manager provides basic configuration management for SQL Server services, server protocols, client protocols, and client aliases.<br>• SQL Server Profiler provides a graphical user interface for monitoring an instance of the Database Engine or an instance of Analysis Services.<br>• Database Engine Tuning Advisor helps create optimal sets of indexes, indexed views, and partitions.<br>• Replication Monitor allows you to track the status and performance of publications and subscriptions across a replication topology.<br>• SQLXML Client Features<br>**Documentation**<br>• SQL Server Books Online is the core documentation for SQL Server 2005.<br>• Software development kits<br>**Development Tools**<br>The Business Intelligence Development Studio is an integrated development environment for Analysis Services, Reporting Services, and Integration Services solutions. |

**Table 1: Component groups during installation process**

Clicking the button “Advanced” will open another dialog, which allows selecting parts of the components, which should be installed in more detail as seen in the following figure.

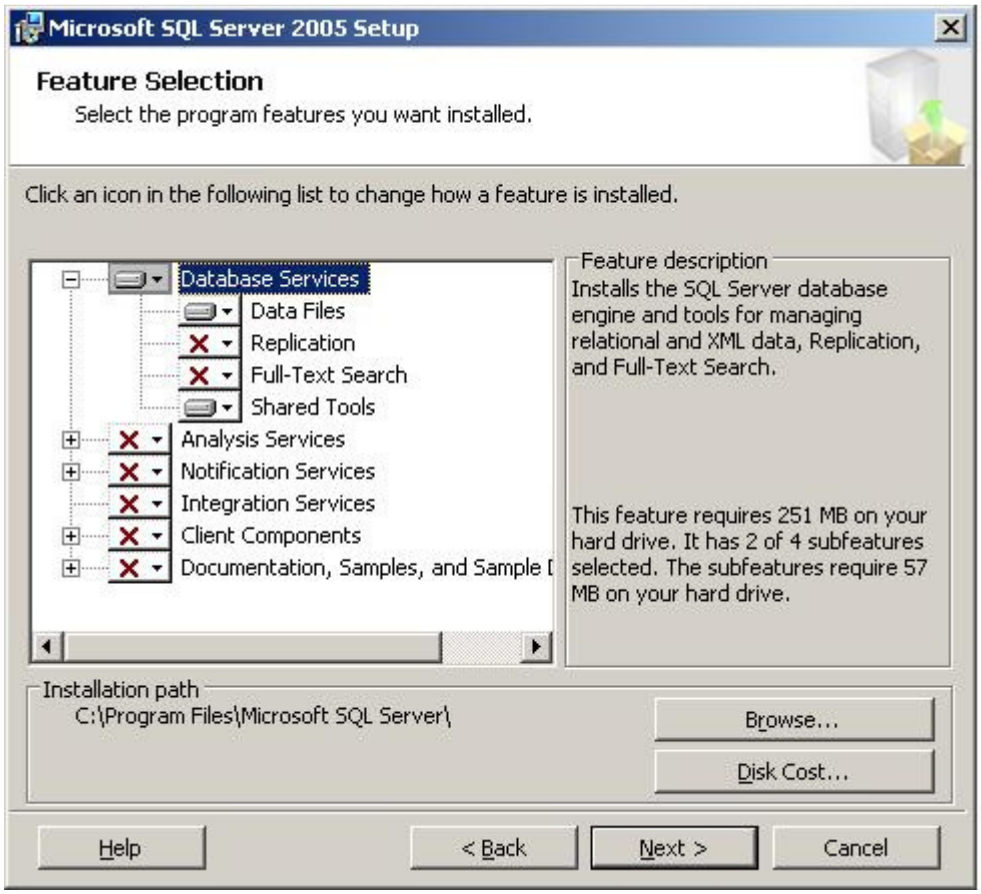


**Figure 10: Installation procedure (X)**

After clicking the "Next" button the user will be asked whether he would like to install a default instance of the TOE or a named instance.

To install a new default instance, there must not be a default instance on the computer. To install a new named instance, the user has to click Named Instance and then type a unique instance name in the space provided.

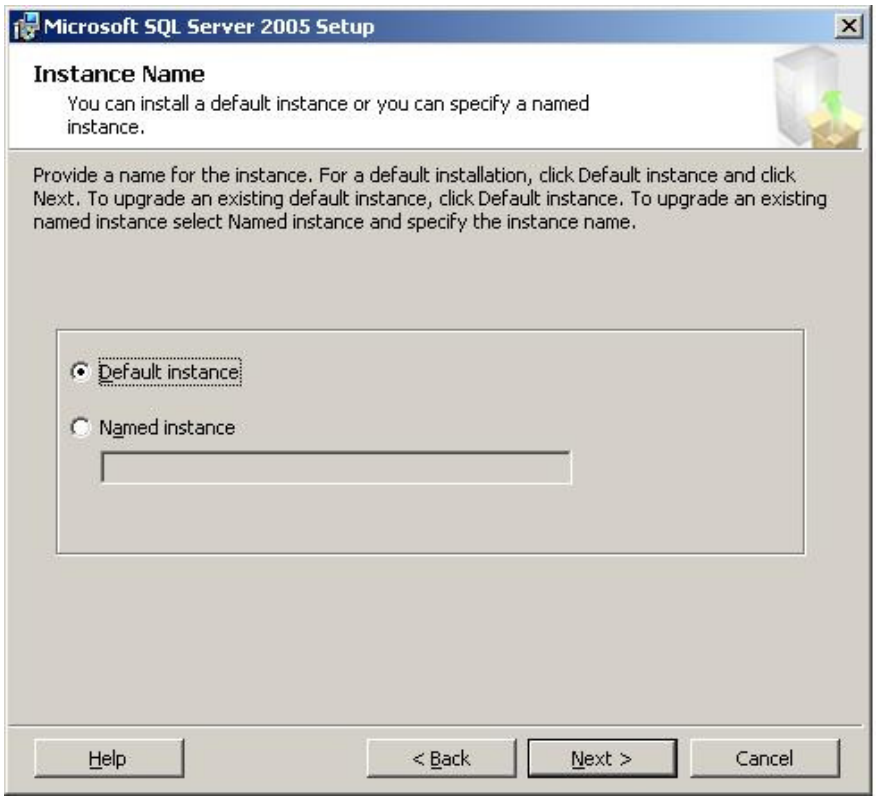


**Figure 11: Installation procedure (XI)**

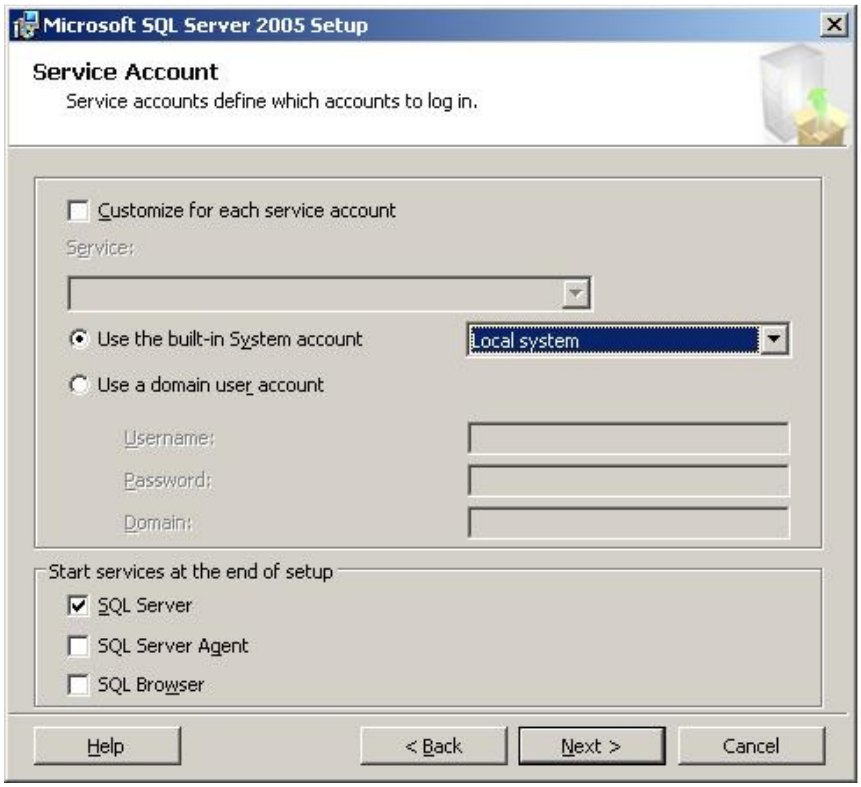


**Figure 12: Installation procedure (XII)**

In the following dialog as seen in the previous figure the user has to choose, which Windows account (either a local account or a Domain user account) shall be used to run the processes of SQL Server 2005.

A separate account shall be used to execute the TOE which has the least sufficient privileges. See [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/instsql9/html/309b9dac-0b3a-4617-85ef-c4519ce9d014.htm#Review_windows_permissions] for more information about the required minimum permissions for each service. Please note that the setup process of the database engine introduces dedicated groups for the services of the SQL Server platform. The account to run a service has to be a member of the corresponding group to ensure that all relevant permissions are granted (see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/instsql9/html/309b9dac-0b3a-4617-85ef-c4519ce9d014.htm]). This helps to limit the possibilities of an evil user who gets unauthorized access to the TOE (e.g. due to a misconfiguration) to misuse functionality of the Operating System. Please refer to [WIN_ADMIN] for a more detailed description of the management of user accounts of the Operating System.

Note: The TOE relies on functionality of the Windows Operating System to protect the running processes (domain separation). To support this functionality it is recommended to use a separate user account for every service, specifically for the cases, where more than one instance is installed on one machine.

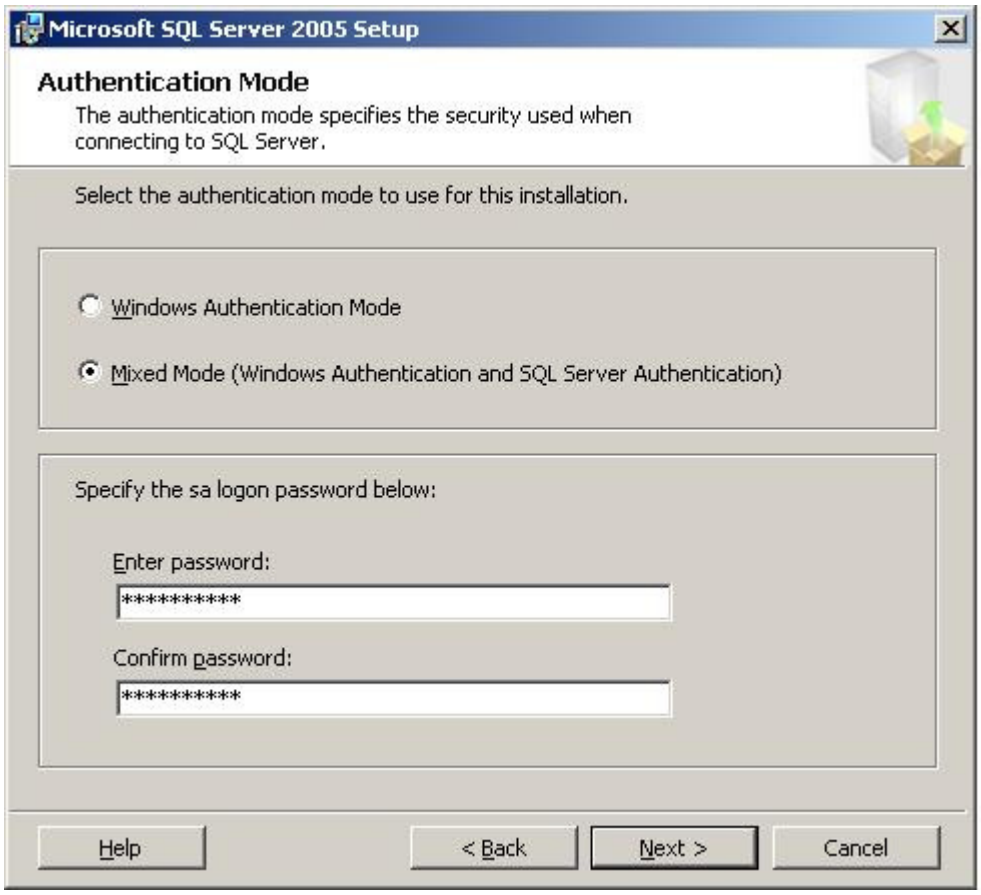


**Figure 13: Installation procedure (XIII)**

In the next dialog as seen in the previous figure the user has to decide whether the SQL authentication should be enabled or not.

If one selects Windows Authentication, Setup will create a ‘sa’ account, which is disabled by default. If one chooses the Mixed Mode Authentication one has to enter a password for ‘sa’. If Windows Authentication has been chosen it will not be possible to use the authentication mechanism of the TOE during operation.

Passwords are the first line of defense against intruders, so setting strong passwords is essential to the security of your system. Never set a blank or weak ‘sa’ password.

For the case that more than one instance of the TOE is installed on one machine it is recommended to use separate ‘sa’ passwords as multiple instances of the database engine represent multiple TOEs and shall have separate accounts for administration.

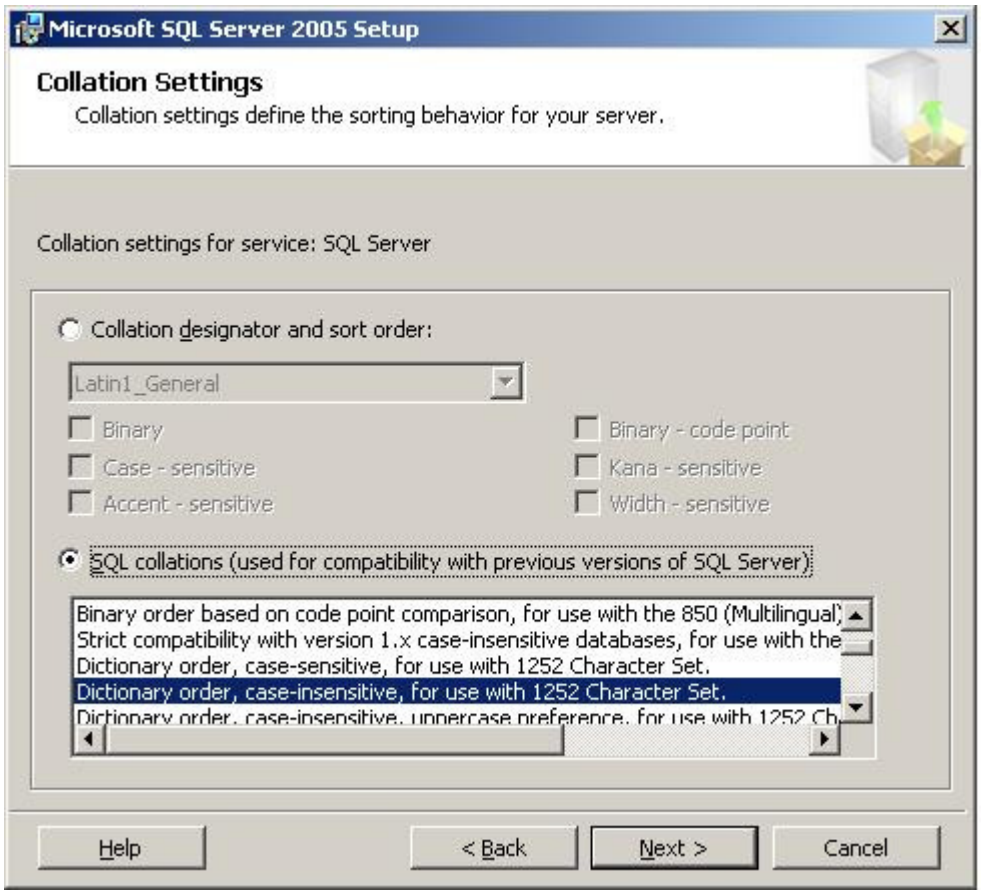


**Figure 14: Installation procedure (XIV)**

On the next dialog as seen in the previous figure the collation settings have to be set. Collation settings, which include character set, sort order, and other locale-specific settings, are fundamental to the structure and function of Microsoft SQL Server databases. More information about these settings can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/instsql9/html/11ce1a3d-8314-41a3-be5f-03db90bea61b.htm].

The next dialog as seen in the following figure asks the user whether he would like to send information about errors and the usage of the features of the TOE to Microsoft. After choosing these options and clicking the “Next” button the user will come to the dialog as seen in Figure 16. It shows a summary of the component which will be installed after clicking the button “Install”.

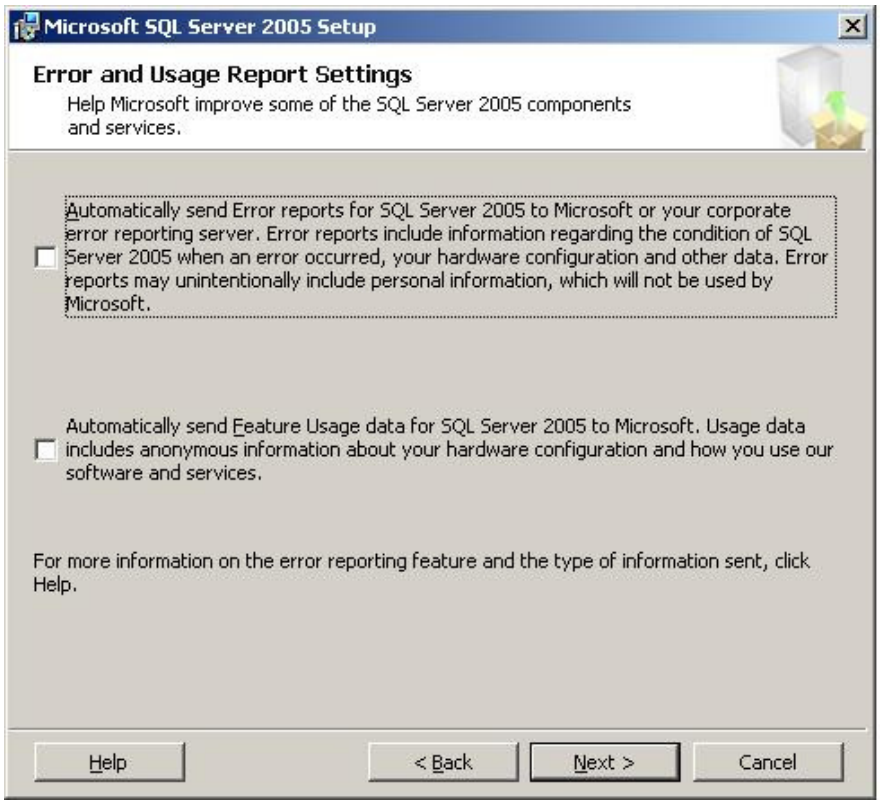


**Figure 15: Installation procedure (XV)**

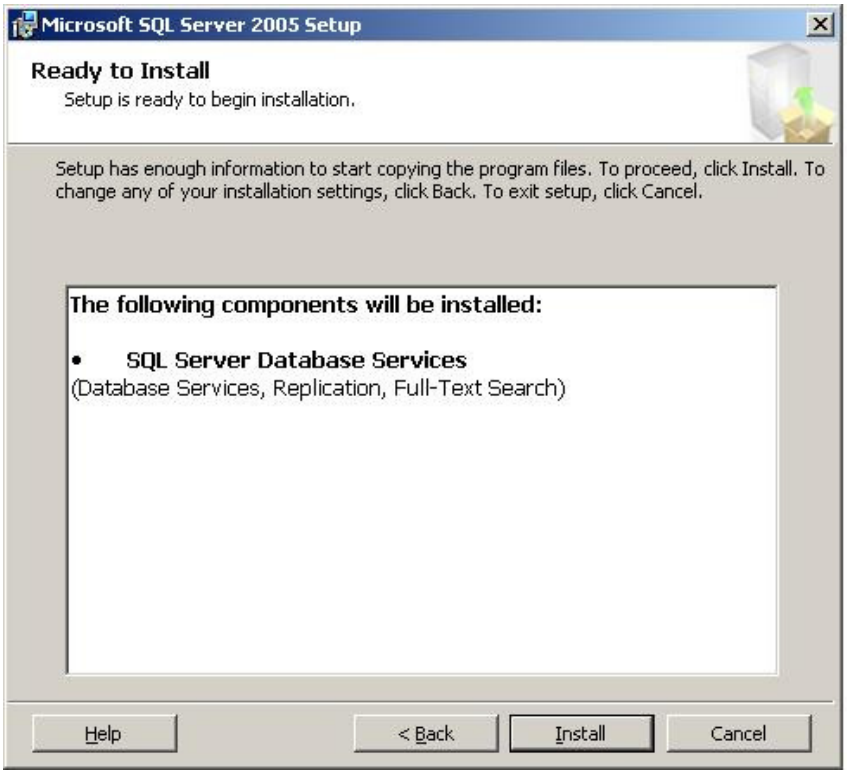


**Figure 16: Installation procedure (XVI)**

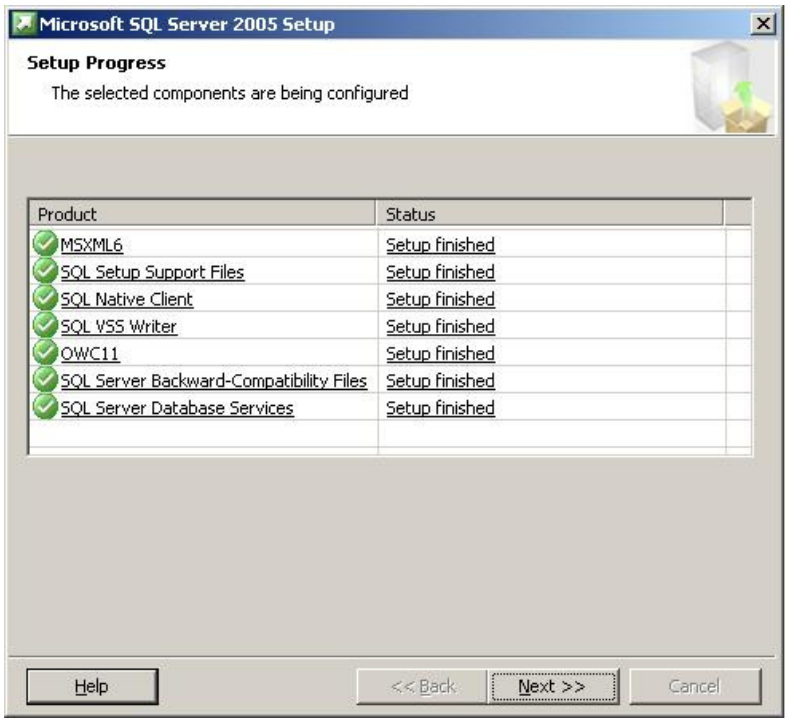


**Figure 17: Installation procedure (XVII)**

The installation process shows its progress in a dialog as seen in Figure 17. After all steps of the process have been finished and the user clicked the button “Next” the setup process shows a last summary of the installation process as seen in the following figure.

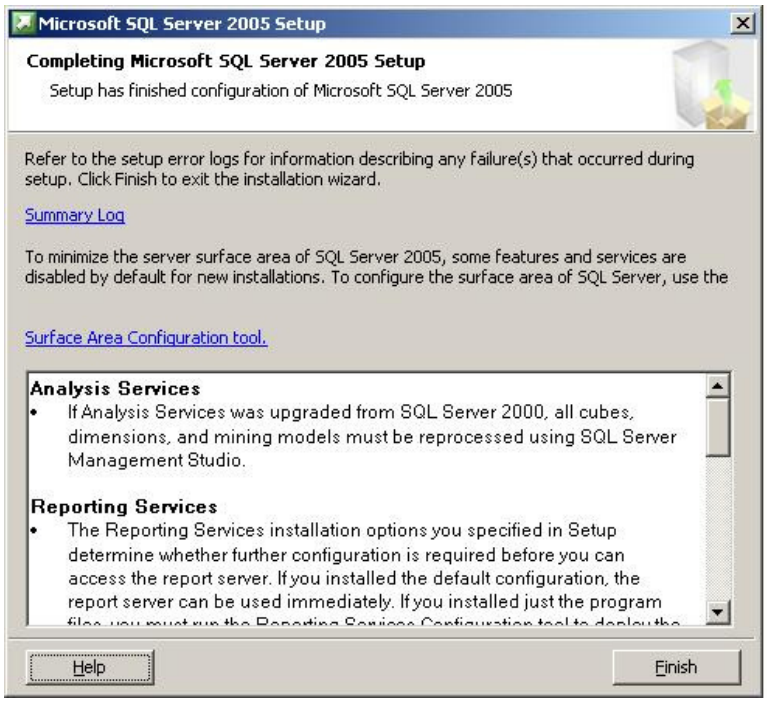


**Figure 18: Installation procedure (XIIX)**

### 3.3.4 Installing SP2

The Service Pack 2 (SP2) which is part of the evaluated version does not ship together with the product. The file "SQLServer2005SP2-KB921896-x86-ENU.exe" which contains the Service Pack can be obtained via http://www.microsoft.com/downloads/details.aspx?FamilyId=d07219b2-1e23-49c8-8f0c-63fa18f26d3a&DisplayLang=en.

Important: Please note that SP2 of SQL Server 2005 was released two times (see also http://support.microsoft.com/kb/933508/en-us). All descriptions in this document assume that the later version of SP2 is used. This can be ensure by verifying the hash value of the downloadable file (see chapter 3.3.2)

The installation process for SP2 is self explaining and does not require any settings specific to the evaluated version of the Database Engine of Microsoft SQL Server 2005.

### 3.3.5 Installing the General Distribution Release (GDR) 4

The General Distribution Release (GDR) 4 which is part of the evaluated version does not ship together with the product. It can be obtained via : http://www.microsoft.com/technet/security/bulletin/ms08-040.mspx.

The integrity of the installation package can be ensured by verifying the hash value of the downloadable file (see also chapter 3.3.2)

The installation process for GDR 4 is self explaining and does not require any settings specific to the evaluated version of the Database Engine of Microsoft SQL Server 2005.

Please note that it is possible that after the evaluation and certification process of the Database Engine of Microsoft SQL Server 2005 as described in this document additional security patches are issued. Therefore the administrator shall visit the Microsoft technet website (http://www.microsoft.com/technet/security/current.aspx) to get informed about new security bulletins. For each new security patch the administrator shall carefully consider to install it (depending on the needs of the concrete installation). The authenticity of each downloadable package can be verified using the digital signature of the file as described in chapter 8.2.

The Microsoft technet also has a site that explains, how the development group of Microsoft products can be contacted for the case that an administrator finds a security bug (https://www.microsoft.com/technet/security/bulletin/alertus.aspx)

### 3.3.6 Checking the version of the product

After the installation process has been finished the admin shall finally determine whether the correct version of the Database Engine of Microsoft SQL Server 2005 is installed. To do

this he has to connect to the running database engine (using any T-SQL client) and execute the following command:

```
SELECT @@VERSION
go
```

Using this command the TOE will return the name of the product platform (of which the TOE is the central part), the version number of the TOE and information about the Operating System. If – as part of this answer - the command returns the version “9.00.3068.00” the correct version of the TOE has been installed.

### 3.3.7 Format of version numbers

Please note that due to reasons of backwards compatibility two different formats for the version numbers of SQL Server are available:

- The Product Version Number returned by the Database Engine of Microsoft SQL Server 2005 (see also chapter 3.3.6) (9.00.3068.00)
- The Version Number of the executable files (2005.90.3068.0)

These version numbers are the same, though they will be in a different format and in fact the same number may be displayed in different formats.

The version number that is returned by the database engine using the select @@version statement as described in chapter 3.3.6 is labeled 9.00.xxxx.yy. The “xxxx” is the build number. For every new version that number is incremented. The “yy” is the number of rebuilds of the same build. On a few occasions, late in the development process if ever, the “yy” represents builds when another product needs to hard code a SQL Server build number before the final build.

For example, the 2nd rebuild of the 127th build of SQL Server is 9.00.0127.02. The rebuild would be 9.00.0127.03 and the next build would be 9.00.0128.00.

The historical fact is that the File Version Number (that is displayed by the Operating System after doing a right-click and choosing “Properties” on that file) was once a date and it still has the year as it’s first part. To retain backward compatibility with the software that uses these, the File Version Number format was not changed.

For that reason, also the „.90.“ in the File Version Number is equivalent to ‚.9.00.’ in the Product Version Number. Note also that leading and trailing zeroes in the Product Version Number are sometimes displayed.

### 3.3.8 Checking the authenticity of the product

To allow a verification of the authenticity of the SQL Server 2005 SP 2 executables Microsoft signs *.cab , *.cat, *.ctl, *.dll, *.exe, *.ocx files with a digital signature.

The administrator shall at least verify the digital signature of the “sqlservr.exe” after the installation has been finished to ensure the authenticity of the product.

For more information on how to verify digital signatures please refer to chapter 8.2.

### 3.3.9 Installing the guidance

The primary guidance for the Database Engine of Microsoft SQL Server 2005 is the “SQL Server Books Online” as of February 2007. The installation package for this can be downloaded from [WEB] and installed using the automatic setup program.

The SQL Server Books Online uses a set of online sources to load help content per default. However as only the version of Books Online as of February 2007 has been used in the context of this evaluation, this version is the only valid version for the certified version. Thus the administrator shall disable the online functionality of Books Online after the installation has been completed as to see in the following figure (via Menu, Tools => Options).

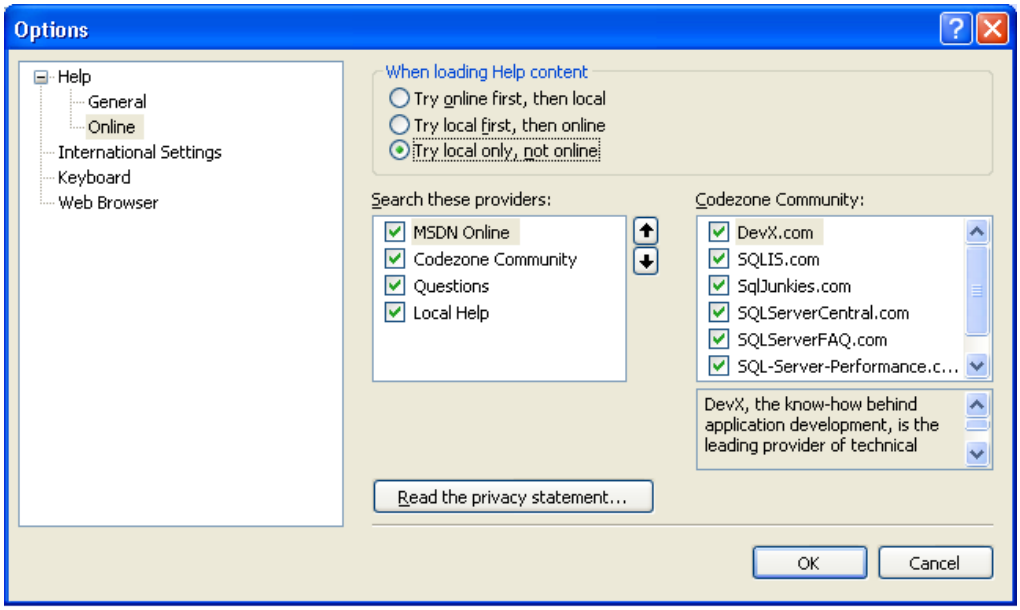


**Figure 19: Changing the settings for Books Online**

### 3.3.10 Enabling the certified version

In the default installation of the Database Engine of Microsoft SQL Server 2005 (including SP 2 and GDR 4) some of the Security Features that are important in the context of the evaluated version are not enabled.

Thus the administrator has to enable the Common Criteria Compliance option that enables:

- **Residual information protection**: This feature requires a memory allocation to be overwritten with a known pattern of bits before memory is reallocated to a new resource. Meeting the RIP standard can contribute to improved security; however, overwriting the memory allocation can slow performance. After the common criteria compliance enabled option is enabled, the overwriting occurs.

- **Login auditing** will be enabled. Each time a user successfully logs in to the Database Engine of Microsoft SQL Server 2005, information about the last successful login time, the last unsuccessful login time, and the number of attempts between the last successful and current login times is made available. These login statistics can be viewed by querying the sys.dm_exec_sessions dynamic management view.
- After the common criteria compliance enabled option is enabled, a table-level **DENY** takes **precedence** over a column-level GRANT. When the option is not enabled, a column-level GRANT takes precedence over a table-level DENY.

To enable this option the administrator shall connect to the database engine and issue the following commands:

```
sp_configure 'show advanced options', 1;
GO
RECONFIGURE;
GO
sp_configure 'common criteria compliance enabled', 1;
GO
RECONFIGURE
GO
```

These setting takes effect directly after the server has been restarted.

For more information please refer to [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/61766eea-c450-408d-af33-fbe7ef8c9ff2.htm]

### 3.3.11 Installing the logon triggers

The Security Function for Session Handling allows an administrator to restrict the ability of users to connect to the TOE based on

- The number of concurrent sessions per login
- The day of the week and time of the day

This functionality is implemented using the logon triggers of the TOE. (For more information about logon triggers please refer to [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/2f0ebb2f-de10-482d-9806-1a5de5b312b8.htm])

This means that a trigger is executed every time a user is attempting to connect to the TOE. This trigger determines whether the user is allowed to establish a session at this time and denies session establishment if necessary.

The tables that store the information for this Security Function, the triggers and the Stored Procedures to manage this functionality have to be installed as they do not ship together with the Database Engine of Microsoft SQL Server 2005.

The installation can easily be done by executing the script "Install_cc_triggers.sql" that can be obtained via [WEB]. This script shall be executed as ‘sa’.

This script will install/create:

The tables:

- dbo.denied_logins_A54E382458CA11DB8373B622A1EF5492
  This table contains the weekly intervals in which logins are not allowed to connect to the Database Engine of Microsoft SQL Server 2005. The table should not be modified directly. The following stored procedures should be used instead:
  - master.dbo.sp_deny_logon
  - master.dbo.sp_revoke_logon_denies
- dbo.maximum_number_of_connections_per_login_A54E382458CA11DB8373B622A1EF5492
  This table contains the value for the maximum number of connections per login. It should not be modified directly. Use the following stored procs instead:
  - master.dbo.sp_set_maximum_number_of_connections_per_login
  - master.dbo.sp_remove_maximum_number_of_connections_limit

The view:

- dbo.denied_logins
  This view dumps the contents of the table with the weekly intervals in human readable format.

The function

- dbo.fn_is_original_login_denied_A54E382458CA11DB8373B622A1EF5492
  This function checks whether the original login (the one who created the session) is allowed to logon at this time. EXECUTE permision for this function is granted to everyone.

The logon trigger

- trig_deny_access_A54E382458CA11DB8373B622A1EF5492
  This trigger is executed on every LOGON attempt. It checks whether the login is allows to logon at this time (based on the time of the day and the day of the week) and if NOT rejects the connection by raising an exception.
- trig_max_connections_A54E382458CA11DB8373B622A1EF5492
  This trigger is executed on every LOGON attempt. It checks whether the login is

allows to logon at this time (based on the maximum number of concurrent session per user) and if NOT rejects the connection by raising an exception.

The Stored Procedures

- dbo.sp_deny_logon_internal_A54E382458CA11DB8373B622A1EF5492
  This is an utility stored procedure and it is not supposed to be called directly
- dbo.sp_deny_logon (see chapter 5.4.1.8.1)
- dbo.sp_revoke_logon_denies (see chapter 0)
- dbo.sp_set_maximum_number_of_connections_per_login (see chapter 5.4.1.8.3)
- dbo.sp_remove_maximum_number_of_connections_limit (see chapter 5.4.1.8.4)
- sp_trace_setcategory (see chapter 5.4.1.6.1)
- sp_trace_setcategory_all (see chapter 5.4.1.6.2)

### 3.3.12 Setting up the trace process

According to [ST] the TOE has to be able to audit a minimum set of events. The TOE logs events in so called trace files. However this trace process is not automatically enabled but has to be created by the administrator.

This can be done by executing a T-SQL script named “EAL4_trace.sql", which can be downloaded from [WEB]. This script that shall be executed as ‘sa’ will install a trace process including all necessary events and ensure that this trace process is started every time the server starts.

More information about the T-SQL commands which are used by this script can be found in chapter 5.4. More details about the events in this trace process and about trace in general can be found in chapter 6.

### 3.3.13 Basic verification of Security Functions

According to [PP] the administrator of the Database Engine of Microsoft SQL Server 2005 shall be provided with a basic test to verify the correct operation of the Security Functions of the TOE.

This test is available in the file verification_script.zip that can be obtained via [WEB].

After unpacking the script locally (i.e. on the machine where the TOE is installed) it can be started by executing the file Verfication_Scripts.bat. This file will execute a set of easy test cases to verify the operation of the Security Functions and print the results to the screen. The following screenshot shows the output of the script for the case that all test cases passed.

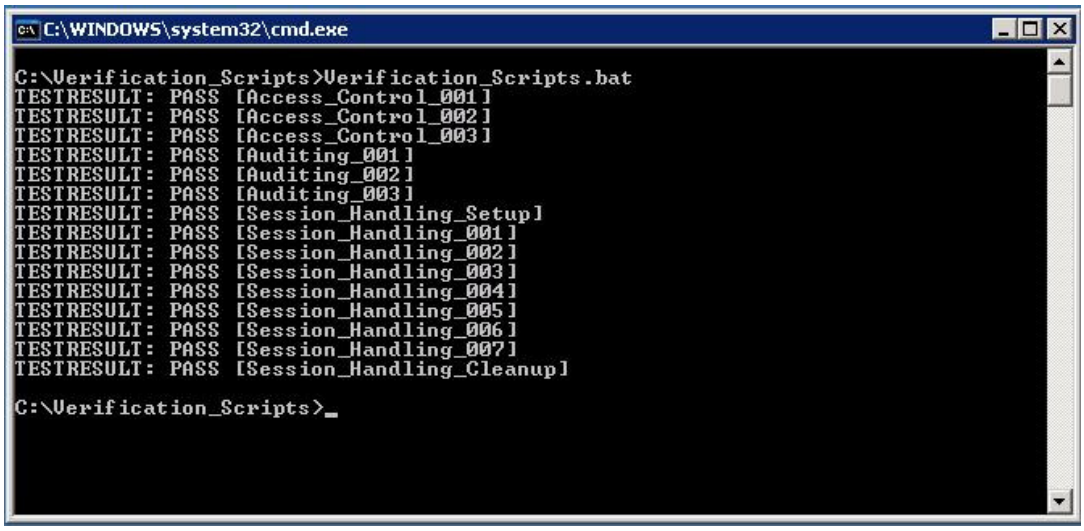


**Figure 20: Basic verification results**

More detailed results (e.g. in case of any error) can be found in the file Verification_Scripts_Result.txt that is created in the directory from which the test cases were started. However, it should be noted that the file Verification_Scripts_Result.txt will contain error messages also for the case that all test cases passed as some test cases produce and expect errors of the database engine. An analysis of the content of the Verification_Scripts_Result.txt should only be necessary if one or more test cases have failed. In this case the content of the file has to be read in the context of the structure of the test scripts.

Please note that the current user has to have administrative privileges and Mixed Mode Authentication has to be enabled in order to run the test scripts.

# 4 SQL SERVER BOOKS ONLINE

The TOE is the security relevant part of a database management system, which primary purpose is to store and retrieve user data in a secure way.

Thus it is impossible to define, who the user of the TOE will be in practice. Many scenarios for the use of a database management system are possible. E.g.

- A user, who uses a T-SQL client for interaction with the Database Engine of Microsoft SQL Server 2005
- An application using the Database Engine of Microsoft SQL Server 2005

Books Online ([AGD]) provides all kinds of users with the necessary information, how the Database Engine of Microsoft SQL Server 2005 can be used.

The following links can be used as entry points into Books Online

| Topic | Reference |
|---|---|
| General information about the database engine | [AGD, Ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en /udb9/html/65e2f424-1386-45a6-8912-bd053f434073.htm] |
| Designing and Creating Databases | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en /udb9/html/f7e79a8e-65e0-4fe3-8c01-252fcfe8631a.htm] |
| Accessing and Changing Database Data | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en /udb9/html/3e7adc44-d03a-4591-b3cd-2c4ce240854e.htm] |
| Administering the Database Engine | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en /udb9/html/f5f597d8-389c-4deb-85bd-5a4c805fe34a.htm] |
| Security Considerations for SQL Server | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en /scsql9/html/cc84bf70-b04a-47fb-b460-b3759d6481b8.htm] |
| SQL Server Management Studio Tutorial | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en /sqltut9/html/31f9e6dc-e946-4576-80bb-729f2bee7478.htm] |
| SQLCMD Utility Tutorial | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en /sqltut9/html/82f57875-d2df-4534-abb4-37eb4fb8c0e3.htm] |
| Writing Transact-SQL Statements Tutorial | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en /sqltut9/html/2addc9be-67d0-423d-a457-192fe9d7d058.htm] |

**Table 2: Entry Points into Books Online**

The following chapters are going to introduce the aspects for the secure administration and usage of SQL the Database Engine of Microsoft SQL Server 2005, which are specific to the certified version.

# 5 GUIDANCE ADDENDUM

This chapter contains the guidance addendum for the secure administration and usage of the TOE. It only covers the aspects of guidance, which are specific to the certified version of the Database Engine of Microsoft SQL Server 2005. It should be seen as a supplement to [AGD].

## 5.1 Modes of operation

In its default configuration the process of the Database Engine of Microsoft SQL Server 2005 is running as a service under Windows 2003 Server and automatically started after the start of the Operating System.

However in some situations it can be useful to start the engine using the "sqlservr.exe" directly and using certain modes of operation.

The following table lists the available options to be used with the "sqlservr.exe" that result in a certain mode of operation:

| Option | Description |
|---|---|
| -c | Shortens startup time when starting the Database Engine of Microsoft SQL Server 2005 from the command prompt. Typically, the SQL Server Database Engine starts as a service by calling the Service Control Manager. Because the Database Engine of Microsoft SQL Server 2005 does not start as a service when starting from the command prompt, use -c to skip this step. |
| -f | Starts an instance of the Database Engine of Microsoft SQL Server 2005 with minimal configuration. This is useful if the setting of a configuration value (for example, over-committing memory) has prevented the server from starting. |
| -g | Specifies an integer number of megabytes (MB) of memory that the Database Engine of Microsoft SQL Server 2005 will leave available for memory allocations within the SQL Server process, but outside the SQL Server memory pool. The memory outside of the memory pool is the area used by the Database Engine of Microsoft SQL Server 2005 for loading items such as extended procedure .dll files, the OLE DB providers referenced by distributed queries, and automation objects referenced in Transact-SQL statements. The default is 256 MB. |
| -h | Reserves virtual address space for Hot Add memory metadata when AWE (Address Windowing Extension) is enabled with 32-bit SQL Server 2005. Required for Hot-Add memory with 32-bit AWE, but consumes about 500 MB of virtual address space and makes memory tuning more difficult. Not required for 64-bit SQL Server. Hot Add Memory is only available for Windows Server 2003, Enterprise and Datacenter editions. It also requires |

| | |
|---|---|
| | special hardware support from the hardware vendor. |
| -m | Starts an instance of the Database Engine of Microsoft SQL Server 2005 in single-user mode. When you start an instance of the Database Engine of Microsoft SQL Server 2005 in single-user mode, only a single user can connect, and the CHECKPOINT process is not started. CHECKPOINT guarantees that completed transactions are regularly written from the disk cache to the database device. (Typically, this option is used if you experience problems with system databases that should be repaired.) Enables the sp_configure allow updates option. By default, allow updates is disabled. |
| -n | Does not use the Windows application log to record SQL Server events. If you start an instance of SQL Server with -n, we recommend that you also use the -e startup option. Otherwise, SQL Server events are not logged. |
| -s | Allows you to start a named instance of the Database Engine of Microsoft SQL Server 2005. Without the -s parameter set, the default instance will try to start. You must switch to the appropriate BINN directory for the instance at a command prompt before starting sqlservr.exe. For example, if Instance1 were to use \mssql$Instance1 for its binaries, the user must be in the \mssql$Instance1\binn directory to start sqlservr.exe -s instance1. |
| -T trace# | Indicates that an instance of the Database Engine of Microsoft SQL Server 2005 should be started with a specified trace flag (trace#) in effect. Trace flags are used to start the server with nonstandard behavior. For more information, see [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/b971b540-1ac2-435b-b191-24399eb88265.htm] |
| -x | Disables the keeping of CPU time and cache-hit ratio statistics. Allows maximum performance. |

**Table 3: Startup Options for "sqlservr.exe"**

The following modes shall not be used within the scope of the certified version as aspects of one or more Security Function as defined in [ST] may be affected.

- -f shall not be used within the scope of the certified version as aspects of one or more Security Function as defined in [ST] may be affected. Specifically the login trigger will be disabled in this mode.
- -m: It cannot be guaranteed that all Security Functions are working in single user mode. Thus this mode must not be used within the certified version.

The following modes will require special care of the administrator. It is highly recommended not to use these modes within a productive environment within the scope of the certified

configuration. However it can be necessary to use these modes for debugging or maintenance purposes within a specially observed environment:

- -n: Though the application log is not a direct part of any Security Function (Audit uses trace files) it is highly recommended not to use this mode within the certified configuration.
- -T Trace#: Indicates that an instance of the Database Engine of Microsoft SQL Server 2005 should be started with a specified trace flag (trace#) in effect. Trace flags are used to start the server with nonstandard behavior. For more information, see [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/b971b540-1ac2-435b-b191-24399eb88265.htm].

The following modes will not affect the behavior of the database engine with respect to the Security Functions and can therefore be used in the scope of the certified version:

- -c: will only shorten the startup process of the engine but not affect the behavior of any Security Function
- -g: is an option for tuning the way memory is handled. No Security Function is affected by this mode. However, to ensure the correct operation of the database engine, this parameter shall not be used with values less than 64 MB.
- -h: is an option to use larger amounts of physical memory. As this option is supported by the certified version of Windows 2003 Server Enterprise Edition and all Security Functions of the Database Engine of Microsoft SQL Server 2005 are working in this mode it can be used
- -s: Simply starts a further instance of the engine. The instances will work independently and enforce all Security Functions.
- -x: This mode can be used as the tuning which is done in this mode to allow maximum performance does not impact the Security Functions as defined in [ST].

Please note that the "sqlservr.exe" provides more options than listed in the previous table. However the other options do not represent a different mode of operation but would e.g. allow the administrator to specify another path for database files or error logs.

A complete overview of the options for "sqlservervr.exe" can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/d373298b-f6cf-458a-849d-7083ecb54ef5.htm]

## 5.2 Interfaces related to administrator roles

This chapter provides more detailed information about the interfaces of the TOE, which are relevant for administration.

The TOE provides two interfaces for administration:

1. Most of the administrative functions are available via the T-SQL interface of the TOE which can be accessed using any T-SQL client.

2. Some of the functions of the TOE rely on settings which are stored and changed in the Operating System. The SQL Server Configuration Manager (see also chapter 5.2.2) which is installed together with the TOE provides the administrator with the possibility to change these settings. Also the starting and stopping of the TOE can be done using this tool.

The next two chapters introduce these two interfaces in more detail.

### 5.2.1 SQL clients

Nearly the complete management functionality of the TOE (actually all management functions except the ones described in chapter 5.2.2) are available via the use of T-SQL commands or Stored Procedures which can be called using a T-SQL command. In this way any T-SQL conformant client can be used for administration.

The SQL Server Management Studio which ships together with the TOE comprises a T-SQL client in a comfortable GUI can be used for administration (see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/toolref9/html/f289e978-14ca-46ef-9e61-e1fe5fd593be.htm]).

However the functionality of the GUI has not been evaluated.

### 5.2.2 SQL Server Configuration Manager

The SQL Server Configuration Manager is a tool to manage the services associated with SQL Server, to configure the network protocols used by the Database Engine of Microsoft SQL Server 2005, and to manage the network connectivity configuration from SQL Server client computers.

SQL Server Configuration Manager is a Microsoft Management Console snap-in that is available from the Start menu, or can be added to any other Microsoft Management Console display.

- SQL Server Configuration Manager can be used to start, pause, resume, or stop the services of SQL Server 2005, to view service properties, or to change service properties.
- The Database Engine of Microsoft SQL Server 2005 supports Shared Memory, TCP/IP, Named Pipes, and VIA protocols for its communication. These protocols can be managed (e.g. disabled and enabled) using SQL Configuration Manager. For information about choosing a network protocols see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/6565fb7d-b076-4447-be90-e10d0dec359a.htm] However the VIA protocol shall not be used within the certified version of the product (see also chapter 7).

More detailed information about the functionality which is provided by the SQL Server Configuration Manager can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/toolref9/html/e6beaea4-164c-4078-95ae-b9e28b0aefe8.htm]

## 5.3 Interfaces related to users

A user without administrative permissions can only connect to the TOE via the T-SQL interface using any T-SQL client via the protocols, which have been enabled by the administrator (see chapter 5.2.2).

The SQL Server Management Studio which ships together with the TOE comprises a T-SQL client, which can be used. (see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/toolref9/html/f289e978-14ca-46ef-9e61-e1fe5fd593be.htm]). However the functionality of the GUI has not been evaluated.

For a complete overview over the T-SQL language please refer to [MSDN, ms-help://MS.MSDNQTR.v80.en/MS.MSDN.v80/MS.SQL.v2000.en/acdata/ac_oview_4pcx.htm ]

## 5.4 Security Functions relevant for administration and use of the TOE

The following four chapters list the Security Functions of the TOE as defined in [ST] and describe, which parts of these Security Functions are accessible for administrators and users.

### 5.4.1 Security Management

As the name implies, the Security Function “Security Management (SF.SM)” as defined in [ST] is the core function for the secure management of the TOE.

For users without any administrative permission this Security Function does not have any accessible part.

For administrators this Security Functions comprises the following aspects:

- Add and delete logins on an instance level
- Add and delete users on a database level
- Add and delete group memberships (for database groups and server groups)
- Create and destroy database groups
- Create, Start and Stop Security Audit
- Include and exclude Auditable events
- Define the mode of authentication for every login
- Modify the action to take in case the audit file is full
- Manage Attributes for Session Establishment

The Security Function SF.SM comprises all aspects, which are relevant for the administration of the Security Functions Identification & Authentication (SF.I&A) and Security Audit (SF.AU). Only the Security Function Access Control (SF.AC) contains an

additional aspect for administration: The possibility to grant and deny permissions to users (see chapter 5.4.2).

The following chapters introduce the commands which can be used via any T-SQL client to perform the operations mentioned before. More details about the commands can be found in [AGD]. Note: For some operations the following chapters list stored procedures to start the operation as well as T-SQL commands. For these cases the T-SQL commands shall be used primarily as the Stored Procedures are legacy commands and will be removed in a future release.

#### 5.4.1.1 Add and delete logins on an instance level

To add and delete logins on an instance level the following T-SQL commands can be used. These commands are also used to specify the type of the login (whether it is associated with a Windows user account or a SQL login) as one has to decide about the type of the login during creation time. Please note that SQL logins are only available if the Mixed Mode Authentication has been chosen during the installation process.

| Command | Purpose | Reference in [AGD] |
|---|---|---|
| sp_addlogin | Add a login | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/030f19c3-a5e3-4b53-bfc4-de4bfca0fddc.htm] |
| Create Login | Add a login | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/eb737149-7c92-4552-946b-91085d8b1b01.htm] |
| sp_droplogin | Delete a login | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/e58684d1-c394-48de-906e-da6ee91100c3.htm] |
| Drop Login | Delete a login | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/acb5c3dc-7aa2-49f6-9330-573227ba9b1a.htm] |

**Table 4: Commands to add and delete logins**

#### 5.4.1.2 Add and delete users on a database level

To add or delete users from/to a database the following commands can be used:

| Command | Purpose | Reference in [AGD] |
|---|---|---|
| Sp_adduser | Add user | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/61a40eb4-573f-460c-9164-bd1bbfaf8b25.htm] |
| Create user | Add user | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/01de7476- |

| | | |
|---|---|---|
| | | 4b25-4d58-85b7-1118fe64aa80.htm] |
| Sp_dropuser | Delete user | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/e28f18f9-7ecf-4568-89f4-fe5c520df386.htm] |
| Drop user | Delete user | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/d6e0e21a-7568-4321-b6d6-bcfba183a719.htm] |

**Table 5: Commands to add and delete users**

#### 5.4.1.3 Add and delete group memberships

To add or delete users from/to a database role/group or a server scoped group the following commands can be used:

| Command | Purpose | Reference in [AGD] |
|---|---|---|
| sp_addrolemember | Add a database user to a group | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/a583c087-bdb3-46d2-b9e5-3921b3e6d10b.htm] |
| sp_addsrvrolemember | Adds a login to a server scoped group | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/777f0e09-8ee5-4cb2-a3ac-939d02c3cd22.htm] |
| sp_droprolemember | Remove a database user from a group | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/c2f19ab1-e742-4d56-ba8e-8ffd40cf4925.htm] |
| sp_dropsrvrolemember | Remove a login from a server scoped role | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/7be99181-d221-49d0-9cb2-c930d8c044a0.htm] |

**Table 6: Commands to add and delete users from database and server groups**

An overview over the predefined server roles that ship together with the product and their permissions can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/13d47a53-1b5a-466f-8117-d060aa8d943e.htm]

#### 5.4.1.4 Create and delete database groups

The following commands can be used to create and delete database scoped groups.

| Command | Purpose | Reference in [AGD] |
|---|---|---|
| Sp_addgroup | Add a group | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/4fc5fa71-8596-4670-9261-e2799b045e0e.htm] |
| Sp_addrole | Add a group | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/e8a21642-8440-419a-8585-93d3d9d44f00.htm] |
| Create role | Add a group | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/b0cd54ad-e81d-4d71-acec-8a6d7261ca08.htm] |
| Sp_dropgroup | Delete a group | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/3b985ed0-2ace-4ce8-9e11-4a1f9ecda767.htm] |
| Sp_droprole | Delete a group | [ADG, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/889ee074-00f8-40a9-bddb-d7d3ef0cbc19.htm] |
| Drop Role | Delete a group | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/1f6f13ae-56a2-4ef1-93f5-8e6151b83e1d.htm] |

**Table 7: Commands to create and destroy database groups**

An overview over the predefined database roles that ship together with the product and their permissions can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/a08108a3-f1fb-43ac-a264-3f2f9749db5d.htm].

#### 5.4.1.5 Create, Start and Stop Security Audit

The following commands can be used to create, start and stop a trace process. When creating a new trace process one has to specify, what should happen in the case where the audit file is full.

| Command | Purpose | Reference in [AGD] |
|---|---|---|
| Sp_trace_create | Create a new trace process | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/f3a43597-4c5a-4520-bcab-becdbbf81d2e.htm] |
| Sp_trace_setstatus | Start and Stop a trace process | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/29e7a7d7-b9c1-414a-968a-fc247769750d.htm] |

**Table 8: Commands to create, start and stop audit**

Please note that a newly created trace will be in a stopped state until it is started using sp_trace_setstatus.

#### 5.4.1.6 Include and exclude Auditable events

The following commands can be used to include and exclude auditable events from/to a trace file and to apply a filter to a trace.

| Command | Purpose | Reference in [AGD] |
|---|---|---|
| Sp_trace_setevent | Include and exclude auditable events | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/7662d1d9-6d0f-443a-b011-c901a8b77a44.htm] |
| Sp_trace_setfilter | Apply a filter to a trace | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/11e7c7ac-a581-4a64-bb15-9272d5c1f7ac.htm] |

**Table 9: Commands to include and exclude auditable event**

Please note that a trace process has to be in a stopped state before a filter can be applied and has to be started over after the filter has been applied.

##### 5.4.1.6.1 Sp_trace_setcategory

This Stored Procedure allows the administrator to enable or disable a given data column for all events in a given trace category.

**Syntax**

sp_trace_setcategory [@traceid=] traceid

,[@categoryid=] categoryid

,[@columnid=]columnid

,[@on=] on

**Arguments**

[@traceid=] traceid

This is the ID of the trace process in which the events shall be enable/disabled. Traceid is of type INT.

[@categoryid=] categoryid

This is the ID of the category (sys.trace_categories) of which all events shall be enabled/disabled. Categoryid is of type INT.

[@columnid=]columnid

This is the ID of the column (sys.trace_columns) that shall be enabled/disabled for all events in the category. Columned is of type INT.

[@on=] on

This argument specifies whether the events shall be enable (1) or disabled (0). On is of type BIT.

**Return Values**

0 (Success) or >0 (Failure)

**Permissions**

Requires the EXECUTE permission on the Stored Procedure and ALTER TRACE permission.

**Remarks**

Please not that other than standard system Stored Procedures that do live in the sys. – schema this Stored Procedure is stored in the dbo-schema of the master database.

#### 5.4.1.6.2 sp_trace_setcategory_all

This Stored Procedure allows the administrator to enable or disable all valid data column for all events in a given trace category.

**Syntax**

sp_trace_setcategory [@traceid=] traceid

,[@categoryid=] categoryid

,[@on=] on

**Arguments**

[@traceid=] traceid

This is the ID of the trace process in which the events shall be enable/disabled. Traceid is of type INT.

[@categoryid=] categoryid

This is the ID of the category (sys.trace_categories) of which all events shall be enabled/disabled. Categoryid is of type INT.

[@on=] on

This argument specifies whether the events shall be enable (1) or disabled (0). On is of type BIT.

**Return Values**

0 (Success) or >0 (Failure)

**Permissions**

Requires the EXECUTE permission on the Stored Procedure and ALTER TRACE permission.

**Remarks**

Please note that other than standard system Stored Procedures that do live in the sys. – schema this Stored Procedure is stored in the dbo-schema of the master database.

### 5.4.1.7 Define the mode of authentication for every login

The mode of authentication for every login of the TOE has to be determined at creation time. The administrator, who creates a new login has to specify, whether a Windows login should be created or a SQL login. This is done via the parameter WINDOWS of the CREATE LOGIN command (See Chapter 5.4.1.1).

#### 5.4.1.8 Manage Attributes for Session Establishment

The following stored procedures can be used to manage the attributes for session establishment. After a default installation of the engine as described in chapter 3.3 of this document the maximum number of sessions per user is set to 5 and initially no further default deny rules are existing.

##### 5.4.1.8.1 Sp_deny_logon

This Stored Procedure allows the administrator to deny session establishment to a certain login based on the day of the week and the time of the day.

**Syntax**

sp_deny_logon [@login_name=] 'login'

,[@start_weekday=] start_weekday

, [@start_time =] 'start_time'

,[@end_weekday=] end_weekday

,[@end_time=] 'end_time'

**Arguments**

[@login_name=] 'login'

Is the name of the login. 'login is of data type **sysname**.

[@start_weekday=] start_weekday

Is the day of the week where the session deny should start. Start_weekday is **tinyint** according to the @@DATEFIRST setting (i.e. 1 means Sunday in the default setting for @@DATEFIRST).

[@start_time =] 'start_time'

Is the time of the day where the session deny should start. Start_time is of **nvarchar(12),** in format hh:mm:ss.000 (the last three digits represent milliseconds)

[@end_weekday=] end_weekday

Is the day of the week where the session deny should end. end_weekday is **tinyint** according to the @@DATEFIRST setting (i.e. 1 means Sunday in the default setting for @@DATEFIRST).

[@end_time=] 'end_time'

Is the time of the day where the session deny should end. end_time is of **nvarchar(12),** in format hh:mm:ss.000 (the last three digits represent milliseconds)

**Return Values**

0 (Success) or >0 (Failure)

**Remarks**

This Stored Procedure can be called with any @@datefirst setting and the start of the interval given can be > than the end of the interval. In this case it splits the passed interval into two intervals.

Please note that other than standard system Stored Procedures that do live in the sys. – schema this Stored Procedure is stored in the dbo-schema of the master database.

**Permissions**

Requires the CONTROL SERVER permission.

#### 5.4.1.8.2 Sp_revoke_logon_denies

This Stored Procedure allows an administrator to revoke all denies from a certain login.

**Syntax**

sp_revoke_logon_denies [@login_name=]'login'

**Arguments**

[@login_name=] 'login'

Is the name of the login for which all denies shall be revoked. 'login is of data type **sysname**.

**Return Values**

0 (Success) or >0 (Failure)

**Remarks**

Please note that other than standard system Stored Procedures that do live in the sys. – schema this Stored Procedure is stored in the dbo-schema of the master database.

**Permissions**

Requires the CONTROL SERVER permission.

#### 5.4.1.8.3 Sp_set_maximum_number_of_connections_per_login

This Stored Procedure allows the administrator to set the maximum number of connections that are allowed per login. This value is a global value that is valid for all logins.

**Syntax**

dbo.sp_set_maximum_number_of_connections_per_login
[@max_connections=] max_connections

**Arguments**

[@max_connections=] max_connections

New value for the maximum number of allowed connection per login. Max_connections is of data type INT.

**Return Values**

0 (Success) or >0 (Failure)

**Remarks**

Please note that other than standard system Stored Procedures that do live in the sys. – schema this Stored Procedure is stored in the dbo-schema of the master database.

**Permissions**

Requires the CONTROL SERVER permission.

#### 5.4.1.8.4 Sp_remove_maximum_number_of_connections_limit

This Stored Procedure allows the administrator to remove the setting for the maximum number of connections that are allowed per login. After successfully executing this Stored Procedure the TOE will not longer enforce any limitation on the number of concurrent sessions per login.

**Syntax**

dbo.sp_remove_maximum_number_of_connections_limit

**Arguments**

-

**Return Values**

0 (Success) or >0 (Failure)

**Remarks**

Please note that other than standard system Stored Procedures that do live in the sys. – schema this Stored Procedure is stored in the dbo-schema of the master database.

**Permissions**

Requires the CONTROL SERVER permission.

### 5.4.2 Access Control

The Security Function Access Control ensures that only users, which have appropriate permissions, are able to perform operations on objects, under the control of the Security Function. The complete description of the Security Function can be found in [ST].

For users without any administrative permission the Security Function Access Control is only accessible in so far that for every command, which is issued to the TOE, the Security Function will check, whether the user has the appropriate permissions.

A part of the Security Function is that it is possible for administrators to grant, revoke or deny permissions to users using the following commands:

| Command | Purpose | Reference in [AGD] |
|---|---|---|
| Grant | Grant a permission | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/a760c16a-4d2d-43f2-be81-ae9315f38185.htm] |
| Revoke | Revoke a permission | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/9d31d3e7-0883-45cd-bf0e-f0361bbb0956.htm] |
| Deny | Deny a permission | [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/c32d1e01-9ee9-4665-a516-fcfece58078e.htm] |

**Table 10: Commands to grant, revoke and deny permissions**

Based on the identity of the user, the group membership of the user and the granted or denied permissions the database engine will decide based on the following rules whether an operation that is requested by a user is allowed:

1. If the requested mode of access is denied to the user, the access will be denied
2. If the requested mode of access is denied to any role of which the user is a member, the access will be denied
3. If the requested mode of access is permitted to that user, the access will be permitted
4. If the requested mode of access is permitted to any role of which the user is a member, the operation will be permitted
5. Else: The access will be denied

It should be noted that the permission check on an object includes the permissions of its parent objects. The permissions for the object itself and all its parent objects are accumulated together before the aforementioned rules are evaluated.

However, there are two cases for which the aforementioned rules are overridden:

1. A sysadmin, the owner of an object and owners of parent objects always have access
2. In the case of “Ownership Chaining” (see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/762249ee-881a-4c3e-b8c0-3a9475039aca.htm] the access is allowed.

### 5.4.3 Identification & authentication

The Security Function Identification & Authentication ensures that each user has been successfully authenticated before any other operations on behalf of that user are allowed. The complete description of the Security Function can be found in [ST].

The Security Function Identification & Authentication only has a small user-accessible part, which is that every user will be authenticated when connecting to the TOE. This applies to administrators and to users without any administrative permissions.

The complete administrative part of this Security Function is covered by the Security Function Security Management (see chapter 5.4.1).

### 5.4.4 Security Audit

This Security Function ensures that the TOE produces audit logs for a set of security relevant actions. These audit logs are stored into trace files in the environment of the TOE. The complete description of the Security Function including the complete list of events can be found in [ST].

For the user of the database engine without any administrative permission the Security Function Security Audit does not have any user-accessible functionality.

The complete administrative part of this Security Function is covered by the Security Function Security Management (see chapter 5.4.1).

For further information about the trace functionality of the TOE please refer to chapter 6.

### 5.4.5 Session Handling

The information about the

- last successful attempt to establish a session
- last unsuccessful attempt to establish a session
- number of unsuccesfull login attempts since the last succesfull login

can be obtained via the dynamic managmement view sys.dm_exec_sessions.

SELECT permission on this management view is granted to public by default so that every user is able to retrieve the information from this view. The user will retrieve information about their current session plus the date and time of the last unsuccessful and successful login attempt (before the current session was established) and the number of unsuccessful login attempts since the last successful login. A user who has the VIEW SERVER STATE permission will see this information for all active sessions.

For more information about this view please refer to [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/2b7e8e0c-eea0-431e-819f-8ccd12ec8cfa.htm]

# 6 SQL Server Trace

The audit functionality of the Database Engine of Microsoft SQL Server 2005 as defined in [ST] is realized by its trace functionality.

This chapter will provide information about the trace functionality, which are of specific relevance for the certification process of the Database Engine of Microsoft SQL Server 2005.

Detailed information about the trace functionality in general can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/deb81e26-d55b-4973-ab83-6de3ca20971c.htm]

The Database Engine of Microsoft SQL Server 2005 has the possibility to maintain several trace processes in parallel and allows the authorized administrator to include and exclude a wide range of events to the processes. For each event a set of data columns can be included, which contains the detailed information about the event. An overview over all events and all columns which can be included into a trace process can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/7662d1d9-6d0f-443a-b011-c901a8b77a44.htm].

The definition of each trace process comprises:

- The events which are captured in the trace process
- The definition of filters which are applied before the events are captured
- The maximum size for the trace file
- The action to take in the case that the trace file is full
- The maximum number of trace files (in case the rollover option has been specified)

The following chapters introduce the information, which have to be audited according to [ST] and the events from the Database Engine of Microsoft SQL Server 2005 trace functionality, which can be used to trace these information.

## 6.1 Information to be audited

The following table lists all the events which need to be audited according to [ST]:

| ID | Event | See Chapter |
|---|---|---|
| 1 | Start-up and shutdown of the audit functions | 6.3.2 |
| 2 | Start-up and shutdown of the DBMS | 6.3.1 |
| 3 | Use of special permissions (e.g., those often used by authorized administrators to circumvent access control policies) | 6.3.3 |
| 4 | All modifications to the audit configuration that occur while the audit collection functions are operating. | 6.3.4 |
| 5 | Successful requests to perform an operation on an object covered by the SFP | 6.3.5 |
| 6 | Unsuccessful revocation of security attributes for subjects and objects | 6.3.6 |
| 7 | Every use of the management functions:<br>• Add and delete logins<br>• Add and delete users<br>• Add and delete group memberships (DB scoped groups, Server scoped groups)<br>• Create and destroy database scoped groups<br>• Create, Start and Stop Audit<br>• Include and Exclude Auditable events<br>• Define the mode of authentication<br>• Define the action to take in case the audit file is full<br>• Manage attributes for Session Establishment | 6.3.7 |
| 8 | Modifications to the group of users that are part of a role. | 6.3.7 |
| 10 | FAU_STG(EXP).4 Every modification to the setting | 6.3.7.8 |
| 11 | Every use of the authentication mechanism including the final decision on authentication | 6.3.8 |
| 12 | Rejection of a new session based on the limitation of multiple concurrent sessions or due toe the session establishment mechanism | 6.3.9 |

**Table 11: Events to be audited**

For these events the following information need to be audited:

- Date and time of the event,
- subject identity (if applicable),
- and the outcome (success or failure)

The following two chapters introduce the trace processes, which are used to audit all necessary events.

## 6.2 Role of the default trace

Every instance of the TOE runs a so called default trace process.

This process

- Is automatically started together with the TOE
- Logs a predefined set of events
- Can only be started and stopped using “sp_configure”
- It uses the rollover option using a maximum number of 5 files and 20 MB per file
- Is stored in a trace file named “log_x.trc” in the default log directory (usually \MSSQL\LOG).

More detailed information about the trace functionality of the TOE can be found in “Introducing SQL Trace” ([AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/deb81e26-d55b-4973-ab83-6de3ca20971c.htm]).

The startup of the TOE (in form of the Audit Server Start and Stop event class) itself is usually only audited in this default trace of the TOE as no other trace processes are yet running in the early phase of startup. However, as the trace process as described in chapter 6.3 logs the start of the database engine in form of a user defined error message the default trace process is not mandatory for the certified version of the Database Engine of Microsoft SQL Server 2005.

Please note that the default trace process cannot be started and stopped using “sp_trace_setstatus” but only via the use of the Stored Procedure sp_configure. Please see [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/9f38eba6-39b1-4f1d-ba24-ee4f7e2bc969.htm] for more information.

For the information that has to be traced as described in chapter 6.1 it is recommended that the administrator of the TOE creates a separate trace process for the certified version of the Database Engine of Microsoft SQL Server 2005, which includes all events which shall be audited according to [ST]. This allows the admin to pay special attention to all events which have to be audited according to [ST].

Such a “CC” Trace process is described in the following chapter.

## 6.3 The “CC Trace”

To trace all the events in the TOE, which are important according to [ST] it is recommended to create a separate trace process, which includes all necessary events as listed in Table 12 and all necessary data columns.

Such a trace process can easily be created using the script “EAL4_trace.sql” as described in chapter 3.3.12.

This script will create and start a trace process with all necessary events. This trace process uses the rollover option (having 100 files, 100 MB each) and the TOE will stop operation if any error occurs in this trace process.

The administrator is free to define other values for the number of trace files or the size per file. He is also free not to use the rollover option but to only have one trace file. However the option to stop the server in case an error occurs in the context of this trace process shall always be used.

In this context it is important to mention that the administrator should ensure that sufficient disc space is available for the trace files as the engine in its default configuration will stop operation if the trace process has to be stopped due to insufficient disc space.

For the case that the TOE stops operation due to insufficient disc space for the trace file the administrator should either provide additional disc space or backup and delete the “old” trace files before starting the engine again.

Per default the trace files for this process are written into the default log directory (usually \MSSQL\LOG) and named “cc_trace_TIMESTAMP[2]_x.trc”. The “CC Trace” process will start automatically after the TOE has been stopped and started again. However as in every other trace process, which uses the rollover option, a rollover will happen (i.e. a new trace file will be started) every time the trace process is started again.

If the script succeeds it will return a message including the internal ID of the trace process and information about the trace files, which are in use.

The following table lists all events, which are included in the “CC trace” process.

[2] Please note that the timestamp, which is used as part of the filename for the trace files has a resolution of 1 second. Thus the execution of the script will abort with an error if the script is started twice in one second.

| event # | Name | Information audited from Table 11 |
|---|---|---|
| 14 | Audit Login | 5, 3, 11 |
| 15 | Audit Logout | 5 |
| 18 | Audit Server Starts and Stops | 1 |
| 20 | Audit Login Failed | 5,11 |
| 42 | Sp:starting | 7 |
| 43 | Sp:completed | 7 |
| 102 | Audit Database Scope GDR Event | 5 |
| 103 | Audit Schema Object GDR Event | 5, 6 |
| 106 | Audit Login Change Property Event | 5 |
| 107 | Audit Login Change Password Event | 5 |
| 108 | Audit Add Login to Server Role Event | 5, 6, 7, 8 |
| 110 | Audit Add Member to DB Role Event | 5, 6, 7, 8 |
| 112 | Audit App Role Change Password Event | 5 |
| 114 | Audit Schema Object Access Event | 5 |
| 115 | Audit Backup/Restore Event Audit DBCC Event | 5 |
| 116 | Audit DBCC Event | 5 |
| 117 | Audit Change Audit Event | 5, 2 |
| 128 | Audit Database Management Event | 5 |
| 129 | Audit Database Object Management Event | 5 |
| 130 | Audit Database Principal Management Event | 5, 6, 7 |
| 131 | Audit Schema Object Management Event | 5 |
| 132 | Audit Server Principal Impersonation Event | 5 |
| 133 | Audit Database Principal Impersonation Event | 5 |
| 134 | Audit Server Object Take Ownership Event | 5 |
| 135 | Audit Database Object Take Ownership Event | 5 |
| 152 | Audit Change Database Owner | 5 |
| 153 | Audit Schema Object Take Ownership Event | 5 |
| 162 | Audit User Error Message | 12, 7 |
| 170 | Audit Server Scope GDR Event | 5, 6 |
| 171 | Audit Server Object GDR Event | 5, 6 |
| 172 | Audit Database Object GDR Event | 5, 6 |
| 173 | Audit Server Operation Event | 5 |

| event # | Name | Information audited from Table 11 |
|---|---|---|
| 175 | Audit Server Alter Trace Event | 5 |
| 176 | Audit Server Object Management Event | 5 |
| 177 | Audit Server Principal Management Event | 5, 7, 8 |
| 178 | Audit Database Operation Event | 5 |
| 180 | Audit Database Object Access Event | 5 |

**Table 12: Necessary audit events**

The following chapters now introduce how these events of the trace mechanism of the Database Engine of Microsoft SQL Server 2005 can be used to audit all the information as required by [ST] and also provide information about relevant information, which are stored in every event.

### 6.3.1 Startup and shutdown of DBMS

The “Audit Server Starts” and “Stops” event class occurs when the Microsoft SQL Server service state is modified.

Please note that the startup of the TOE cannot be logged via this event as the trace process (other than the default trace) is not yet running when the TOE starts. The startup of the TOE is logged using the User Error Message as described in chapter 6.3.9 as early as possible (i.e. directly after the trace has been started). This event is fired by the script that also installs the CC trace process (see chapter 6.3).

This event contains the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| Success | 1 = success. 0 = failure. |
| EventSubClass | Type of event subclass.<br>1=Shutdown, 2=Started, 3=Paused, 4=Continue |

**Table 13: Important attributes of "Audit Server Starts and Stops" event**

Please note that the event which indicates that the server has been started will always show success as otherwise the server would not have been started.

A complete description of the event can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/8ddb55af-c77b-4d07-b803-a97320c0804e.htm]

### 6.3.2 Startup and shutdown of audit functions

The "Audit Change Audit event class" occurs whenever an audit trace modification is made. Modifications in the context of this event comprise specifically to stop and start a trace process.

This event contains the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| DBUserName | SQL Server database user name of the client. |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| Success | 1 = success. 0 = failure. For example, a value of 1 indicates success of a permissions check and a value of 0 indicates failure of that check. |
| EventSubClass | Type of event subclass.<br>1=Audit started, 2=Audit stopped, 3=C2 mode ON, 4=C2 mode OFF |
| TextData | Text containing additional information. This text also contains the information, which trace process is started or stopped. |

**Table 14: Important attributes of "Audit Change Audit" event**

More information about this event can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/8cfacc82-cee8-4199-a69e-acedecfc0b3b.htm]

Furthermore every trace file contains the "Trace Start" event as the first event and the "Trace Stop" event as the last event. However these events only show that the trace has been started or stopped and include no additional information beside the date and time of the event.

### 6.3.3 Use of special permissions

Authorized administrators do not have to take a specific action before they are allowed to perform administrative actions. Hence the only event which can be audited for the use of these special permissions is the fact that an authorized administrator has logged on to the TOE. This is covered by the event as described in chapter 6.3.8.

### 6.3.4 Modifications to the audit configuration

The "Audit Server Alter Trace event class" occurs for all statements that check for the ALTER TRACE permission. Statements that check for ALTER TRACE include those used to create or configure a trace, or to set a filter on a trace.

This event contains the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| DBUserName | SQL Server database user name of the client. |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| TextData | Text value dependent on the event class captured in the trace. |

**Table 15: Important attributes of "Audit Server Alter Trace" event**

It should be noted that it is not possible to modify the configuration of a trace process while this process is running. To apply a filter to a trace process the trace process has to be stopped first. Defining the setting, what should happen in case the audit files are full (ROLLOVER OPTION) is only possible at creation time. If this setting should be changed for a running trace process, the trace process would have to be stopped and a new trace process has to be created.

If it is necessary to stop one or both of the trace processes mentioned before while the TOE is still running (e.g. to change the configuration of the trace process) it should be considered to create a new trace process which contains all the relevant events and to start this new process before the CC trace process is stopped. In this way it can be ensured that the admin misses no important event.

### 6.3.5 Requests on operation

The [ST] requires that every successful request by a user to perform an operation on an object has to be audited by the TOE.

The sum of the events as listed in Table 12 is suitable to meet this requirement as all operations on objects, which can be performed, are covered by the set of these events.

It should be noted that most of the audit events in Table 12 result directly out of the access control functions of the TOE. This also means that if a user operation requires more than one permission the trace file will contain more than one audit event for this operation.

For example: If a user attempts to create a new table the CREATE TABLE permission and the ALTER SCHEMA permission on the corresponding schema are needed. Thus if a user attempts to create a table two events will show up in the trace file and the access control check for the operation can only be considered successful if both events show success.

### 6.3.6 Unsuccessful revocation

#### 6.3.6.1 For objects

For objects within the TOE the only (implicitly) defined attributes which can be revoked are the corresponding Access Control Entries (ACE) which are used for access control.

By the use of the T-SQL commands REVOKE, DENY and GRANT authorized users are able to modify these ACEs. The use of the commands REVOKE and DENY can be seen as revocation in terms of [CC].

The following events are fired for every REVOKE, DENY and GRANT statement for specific objects:

- Audit Database Scope GDR Event
- Audit Schema Object GDR Event
- Audit Server Scope GDR Event
- Audit Server Object GDR Event
- Audit Database Object GDR Event

The sum of these events covers all REVOKE, DENY and GRANT statements, which could happen in the TOE.

These events contain the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| DBUserName | SQL Server database user name of the client. |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| Success | 1 = success. 0 = failure. For example, a value of 1 indicates success of a permissions check and a value of 0 indicates failure of that check. |
| EventSubClass | Type of event subclass., 1=Grant, 2=Revoke, 3=Deny |
| DatabaseName | Name of the database in which the user statement is running. (if available) |
| ParentName | Name of the schema the object is within. (if available) |
| ObjectName | Name of the target object |
| TextData | Text value dependent on the event class captured in the trace. |

**Table 16: Important attributes of "Audit Object GDR" events**

More detailed information about these events can be found in:

**Audit Database Scope GDR:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/1641a38a-ef24-46ce-b2f4-bf732858c771.htm]

**Audit Schema Object GDR:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/a0187811-dc71-4792-a282-3bfe1ca90c21.htm]

**Audit Server Scope GDR Event:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/d3b1e47f-2ba2-49af-b404-1aa231d4e4a0.htm]

**Audit Server Object GDR:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/117fedca-c1c4-469a-929a-9ea332c83d25.htm]

**Audit Database Object GDR Event:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/2289aab5-e048-4288-bcae-aaf768ca014a.htm]

#### 6.3.6.2 For subjects

Unsuccessful revocation of security attributes of subjects in the context of the evaluated version of the Database Engine of Microsoft SQL Server 2005 could mean:

- Revoke the group membership of logins or database users (see 6.3.7.3)
- Delete a database user or login (see 6.3.7.1 and 6.3.7.2)

### 6.3.7 Use of Management functions/Modifications of groups

The following chapters introduce the events which can be used to trace the use of the management functions of the TOE.

#### 6.3.7.1 Add/delete logins

The “Audit Server Principal Management” event class occurs when server principals are created, altered, or dropped. Server principals include all logins and server scoped roles.

This event contains the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| DBUserName | SQL Server database user name of the client. |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| Success | 1 = success. 0 = failure. For example, a value of 1 indicates success of a permissions check and a value of 0 indicates failure of that check. |
| EventSubClass | Type of event subclass.<br>1=Create, 2=Alter, 3=Drop, 4=Dump, 5=Disable, 6=Enable, 11=Load |
| TextData | Additional information about the principal which is managed in form of a SQL string.<br>This text field also contains information of which type a login is (SQL or Windows) for the case that a login is created |
| ObjectName | Name of the object being referenced. |

**Table 17: Important attributes of "Audit Server Principal Management" event**

More information about this event can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/7894850c-91fe-47c0-a03c-baacbc10d29c.htm]

#### 6.3.7.2 Add/Delete users:

The "Audit Database Principal Management event" class occurs when database principals, such as users, are created, altered, or dropped from a database. Database principals comprise database users and database scoped groups.

This event contains the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| DBUserName | SQL Server database user name of the client. |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| Success | 1 = success. 0 = failure. For example, a value of 1 indicates success of a permissions check and a value of 0 indicates failure of that check. |
| EventSubClass | Type of event subclass.<br>1=Create, 2=Alter, 3=Drop, 4=Dump, 11=Load |
| TextData | Additional information about the principal which is managed in form of a SQL string. |
| ObjectName | Name of the object being referenced. |

**Table 18: Important attributes of "Audit Database Principal Management" event**

More information about this event can be found in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/594eec78-677c-4500-ae9b-e400abf6f39c.htm]

#### 6.3.7.3 Add and delete group membership for database and server scoped groups

The following events cover the use of this management functionality:

- "Audit Add Login to Server Role" Event Class indicates that a login was added or removed from a fixed server role.
- "Audit Add Member to DB Role" Event Class indicates that a login has been added to or removed from a database role.

The "Audit Add Login to Server Role" event has the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| DBUserName | SQL Server database user name of the client. |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| Success | 1 = success. 0 = failure. For example, a value of 1 indicates success of a permissions check and a value of 0 indicates failure of that check. |
| EventSubClass | Type of event subclass.<br>1=Add, 2=Drop |
| RoleName | Name of the fixed server role whose membership is being modified. |

**Table 19: Important attributes of "Audit Add Login to Server Role" event**

The "Audit Add Member to DB Role" event has the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| DBUserName | SQL Server database user name of the client. |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| Success | 1 = success. 0 = failure. For example, a value of 1 indicates success of a permissions check and a value of 0 indicates failure of that check. |
| EventSubClass | Type of event subclass.<br>1=Add, 2=Drop, 3=Change group |
| RoleName | Name of an application role being enabled. |

**Table 20: Important attributes of "Audit Add Member to DB Role" event**

More detailed information about these events can be found in:

**"Audit Add Login to Server Role" event:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/7a8ed1c3-a98f-4f93-a6ba-e3901d941db9.htm]

**"Audit Add Member to DB Role" event:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/a5ac46b6-765b-4424-b6c7-4d5a1b898d65.htm]

#### 6.3.7.4 Create and destroy database groups

See chapter 6.3.7.2.

#### 6.3.7.5 Start and stop the audit process

See chapter 6.3.2.

#### 6.3.7.6 Include and exclude auditable events.

It is not possible to include or exclude auditable events from or to a trace process while this trace process is running. One has to stop the trace process, apply a filter and start the trace process again. The trace files will contain a Stop and a Start event to indicate that it has been stopped and started again.

See chapter 6.3.4 for more information.

#### 6.3.7.7 Define mode of authentication for every login

The type of a login is defined at creation time. It cannot be changed afterwards. See also chapter 6.3.7.1 as it describes the event which can be captured when a new login is created.

#### 6.3.7.8 Define the action to take in case the audit file is full.

See chapter 6.3.4.

#### 6.3.7.9 Manage Attributes for Session Establishment

The attributes that are used to determine whether a login is allowed to connect to the TOE based on the number of concurrent sessions, the day of the week and the time of the day can be managed via the Stored Procedures as described in chapter 5.4.1.8. These Stored Procedures do not ship together with the Database Engine of Microsoft SQL Server 2005 but are installed via a script.

The use of these Stored Procedures is traced via the User Defined Error Message. For important data columns of this event please refer to chapter 6.3.9.

### 6.3.8 Use of the authentication mechanism

Every use of the authentication mechanism is covered by the use of the following two events:

- “Audit Login” which indicates that a user has successfully logged into the Database Engine of Microsoft SQL Server 2005.
- “Audit Login Failed” which indicates that a user attempted to log in to the Database Engine of Microsoft SQL Server 2005

The Audit Login event contains the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| Success | 1 = success. 0 = failure. This event will always show success. |
| TextData | Semicolon-delimited list of all set options. |

**Table 21: Important attributes of "Audit Login" event**

The Audit Login Failed event contains the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| Success | 1 = success. 0 = failure. This event will always show failure. |
| TextData | Text value dependent on the event class captured in the trace. |

**Table 22: Important attributes of "Audit Login Failed" event**

More detailed information can be found in:

**Audit Login:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/ad0bdb48-7f9f-4335-805d-7769d6df89b2.htm]

**Audit Login Failed:**

[AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/6b83963b-b685-429d-92ba-5173f6f0000d.htm]

### 6.3.9 Rejection of Sessions

The rejection of a new session based on the limitation on number of concurrent sessions per user or due to a session deny is realized via logon triggers.

These logon triggers throw a user defined error message when they deny a session. This event is contained in the CC trace process.

This event contains the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| TextData | Text of the error message or exception |

**Table 23: Important attributes of "Audit User Error Message" event**

The TextData attribute of the event will contain more details about the reason why the login was denied. Assuming that the script as described in chapter 3.3.11 was use to install the trigger will say:

User Error Message: "Logon failed for login 'x'. The maximum number of Y connections per login was exceeded. (for deny based on max. number of sessions)

Or

User Error Message: "Logon failed. Login 'test' is temporarily disabled." (For deny based on day of the week/time of the day)

Further in both cases a User Error Message can be found with a generic message (that will also be forwarded to the client): "Logon failed for login 'X' due to trigger execution".

Please note that in addition to the User Error Messages the Audit Login Failed event class (see also chapter 6.3.8) is fired as the login process of the user failed.

For more information about the event please refer to [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/d7594261-ccd9-487c-9678-11875ba57fb7.htm]

### 6.3.10 Execution of Stored Procedures

Some stored procedures fire relevant events using the User Error Message event class. As every user is in principle able to fire such a User Error Message those messages can in principle not be considered to be safe against spoofing. For this reason the events "sp:starting" and "sp:completed" have been added to the CC trace file definition. This will allow the administrator to see whether a User Error Message has really been fired by a Stored Procedure (in which case the User Error Message will occur between the two corresponding "sp:starting" and "sp:completed" events and the SPID column of sp:starting and sp:completed will show the same ID as for User Error Message).

These events contain the following attributes which are important in the context of this evaluation:

| Attribute | Description |
|---|---|
| SPID | ID of the session on which the event occurred. |
| StartTime | Time at which the event started, if available. |
| LoginName | Name of the login of the user (either the SQL Server security login or the Microsoft Windows login credentials in the form of DOMAIN\username). |
| SessionLoginName | The login name of the user who originated the session. For example, if you connect to SQL Server using Login1 and execute a statement as Login2, SessionLoginName displays Login1, while LoginName displays Login2. This data column displays both SQL Server and Windows logins. |
| DatabaseName | Name of the database in which the user statement is running. (if available) |
| ObjectName | Name of the Stored Procedure |

**Table 24: Important attributes of sp:starting and sp:completed**

For more information about the events please refer to [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/ef55e579-080d-4650-a7fc-4dd03ed8e391.htm] and [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/7636a433-5d32-4562-8f5a-694f8e2beeca.htm].

## 6.4 Deeper audit

It should be noted that the trace functionality of Database Engine of Microsoft SQL Server 2005 offers many more event classes than the events listed and described in the previous chapters. Further for most of the events additional columns with more detailed information are available.

This allows an administrator to perform an even deeper audit than required in the context of this evaluation.

It would even be possible to log an audit event every time a SQL-Statement has been executed. (Audit SQL:StmtCompleted Event, see [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/a55f005d-e020-423c-8940-c24ea1b20104.htm]).

However the administrator should consider that a deeper audit will produce bigger trace files and that sufficient disc space for the trace files should be available.

## 6.5 Filtering of audit and prevention of audit loss

The TOE provides the authorized administrator with the possibility to include or exclude auditable events based on:

a) user identity and/or group identity,

b) object identity,

c) success or failure of auditable security events;

To include or exclude events based on these attributes one has to:

1. Stop the trace process
2. Apply a filter to the trace process
3. Start the trace process again

To apply the filter the Stored Procedure “sp_trace_setfilter” can be used. Please see [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/11e7c7ac-a581-4a64-bb15-9272d5c1f7ac.htm] for more details.

## 6.6 Security Relevant Events

The trace capabilities of the TOE are a powerful mechanism to detect potential security breaches. However the secure operation of the TOE needs the attention of the administrator. He shall review the trace files regularly and pay attention to any suspicious events.

As the definition of “suspicious” depends on the concrete installation and environment of the TOE it is not possible to provide a comprehensive definition of what suspicious events are. For example 1000 unsuccessful authentication attempts or failed read attempts per hour may not be suspicious in an installation that serves millions of users while it would be highly suspicious in installations with only a few users.

Classical suspicious events could e.g. be

- An unusual high amount of unsuccessful authentication attempts, which could point to a brute force attack.
- An unusual high amount of events recorded in the trace files could be an indication for an attacker, who is trying to flood the trace files in order to conceal an unauthorized operation.

# 7 Recommendations and requirements for secure administration, configuration and usage

The administrator of the TOE shall follow the following recommendations and requirements to ensure a secure operation of the TOE:

## 7.1 Recommendations/requirements about Security Audit

- It is recommended to use a separate trace process (also called “CC trace”) to audit all the events which have to be captured according to [ST]. See also chapter 3.3.12 for further guidance to create this trace process.
- The CC trace process should always be running. If it is necessary to stop the trace process while the TOE is still running (e.g. to change the configuration of the trace process) it should be considered to create a new trace process which contains all the relevant events as listed in chapter 6.3 and to start this new process before the CC trace process is stopped. In this way it can be ensured that the admin misses no important event.
- For the “CC trace” process it has to be ensured that the option “SHUTDOWN_ON_ERROR” is used, i.e. that the TOE will stop operation in case an error occurs. This option can be combined with the option “TRACE_FILE_ROLLOVER”.
- For the case that the TRACE_FILE_ROLLOVER option is used it is possible that an attacker floods the audit and intentionally causes an event to be overwritten. Thus the administrator has to ensure that sufficient disc space is available for the trace files and appropriate settings are used for the trace processes. Specifically – in cases where the audit of certain event is more important than the availability of the server – it should be considered not to use the TRACE_FILE_ROLLOVER option (i.e. to ensure that the server will shut down if the trace file is full) for all or certain trace processes.

## 7.2 Recommendations/requirements and further information about Access Control

- It should be mentioned that some permissions with the Database Engine of Microsoft SQL Server 2005 do imply other permissions. A good example of such a permission is the CONTROL SERVER permission that covers all other permissions. The complete hierarchy of permissions within the Database Engine of Microsoft SQL Server 2005 is contained in the file permission_hierarchy.zip that can be downloaded from [WEB]. This file contains 4 charts that show the permission hierarchy on the 4 levels: server, database, object and column.
- According to the concept for Access Control in the Database Engine of Microsoft SQL Server 2005 it is possible (if not likely) that two users/administrators have the

same permission for one object. This could lead into a situation, where administrators/users cause conflicting operation (e.g. that one administrator grants access to an object while a second administrator denies the same access). These situations can only be avoided by organizational mechanisms and the administrator should be well aware of this fact.

- In its default configuration the Database Engine of Microsoft SQL Server 2005 grants the EXECUTE permission on many Stored Procedures to public. This has been done to ensure a maximum level of compatibility to applications. However, some of the Stored Procedures do provide access to sensitive information or open channels for potential attacks. Therefore the administrator shall consider to revoke the EXECUTE permission on all Stored Procedures from public and grant those EXECUTE permissions to specific users or their corresponding groups if necessary.
- The internal access control functionality of the Stored Procedures 'sp_replsendtoqueue' and 'sp_replwritetovarbin' is not compliant to [PP]. Therefore these two procedures must not be accessible by any user within the scope of the certified version of the database engine. After a default installation however the execute permission on these Stored Procedures is granted to public. Therefore the administrator shall revoke the execute permissions from theses Stored Procedures from public.
- The two Stored Procedures sp_fetchLOBfromcookie and sp_fstr_getlocalnameforwinfs that ship together with the database engine have not been considered during the evaluation as they belong into the scope of WinFS. As these Stored Procedures are not part of the evaluated version of the database engine the administrator shall revoke the execute permissions on these Stored Procedures from public to ensure that they are not used.
- The two Stored Procedures sp_enum_dtspackages and sp_get_dtspackage may provide low privileged users with access to sensitive information (see also http://msdn2.microsoft.com/en-us/library/aa933557(SQL.80).aspx) as the EXECUTE permission on those is granted to public after a default installation. Therefore the administrator shall revoke the execute permissions from theses Stored Procedures from public to ensure that they cannot be executed by everybody. The EXECUTE permission on those Stored Procedures shall only be granted to users if absolutely necessary.
- The description of the CREATE FULLTEXT CATALOG statement in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/d7a8bd93-e2d7-4a40-82ef-39069e65523b.htm] describes that the CREATE FULLTEXT CATALOG permission is needed to execute this statement. However the ALTER ANY FULLTEXT CATALOG permission is needed to successfully execute this statement. This will be documented in future versions of the guidance documentation.
- The description of the sp_dropsrvrolemember in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/7be99181-d221-49d0-9cb2-

c930d8c044a0.htm] describes that the membership in the sysadmin fixed server role, or both ALTER ANY LOGIN permission on the server and membership in the role from which the member is being dropped. However to successfully execute this Stored Procedure the pure membership in the role from which a user should be removed is sufficient. The administrator should be aware of the fact that a login who is added to a server role does in this way implicitly inherit the permission to remove all other logins from that role.

- The description of the CREATE LOGIN statement in [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/eb737149-7c92-4552-946b-91085d8b1b01.htm] describes that the ALTER ANY LOGIN permission on the server is needed. However – as an exception – the CREATE LOGIN statement can also be executed by a user to create a login for his own Windows account (in this case the user would have access due to the membership in a Windows group).

## 7.3 Recommendations/requirements about Session Handling

- The TOE relies on functionality to enforce the separation of different user processes. However the database engine of SQL Server 2005 supports a so called lightweight pooling option (see [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/udb9/html/f5378318-6142-49f9-ba27-fce5a0483cbc.htm]) to reduce the overhead of context switching. In this mode the database engine would not comply to the requirements from [PP]. Thus the administrator shall not enable this mode (the mode is disabled in a standard installation).
- The security functionality to control session establishment based on the number of concurrent sessions per user or based on other attributes (such as day or time) is not enforced when using the Dedicated Admin Connection
- SQL Server endpoints do support session pooling (see [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/6405e7ec-0b5b-4afd-9792-1bfa5a2491f6.htm]) However, the behaviour of endpoint is not compliant to [PP]. Therefore the functionality for session pooling (that is disabled by default) must not be used for the certified version. Specifically the option "SESSIONS" must not be set to enabled for any endpoint.
- It should be noted that any changes to logins that occur while a user is connected to the database engine may require the user to log off and log on again before the updated settings take effect. The administrator should therefore consider to terminate a user session (using the KILL command, see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/071cf260-c794-4b45-adc0-0e64097938c0.htm]) in case of important changes to the login of that user (e.g. the change of group memberships of a user). Further it is possible that sessions are

cached after a user disconnected and that a cached session may be reused in case a user logs in again. Changes to login may not be applied to cached sessions under certain circumstances. To avoid this behaviour the administrator shall consider to run the command "DBCC FREESYSTEMCACHE 'ALL'" after important changes to one or more logins. If the server is involved in scenarios of distributed queries the administrator shall further consider to run the “DBCC FREESESSIONCACHE” command in those cases.

## 7.4 Recommendations/requirements about Identification and Authentication (Secure Passwords)

- The administrator(s) and users shall ensure that their passwords for SQL Server logins are of sufficient quality. General guidance, how to create strong passwords can be found under http://www.microsoft.com/athome/security/privacy/password.mspx.
- The concrete settings for the enforcement of minimum password requirements on the underlying Operating System depend on the concrete installation ([WIN_CONF]) provides different templates). To allow the secure operation of the TOE the administrator shall ensure that the OS enforces strong password using not less than the following settings (for guidance on this functionality of the OS please refer to [WIN_CONF]):
    - Password must be at least 8 characters in length
    - "password must meet complexity requirements" setting of the OS is enabled. This will ensure that passwords:
        - Do not contain all or part of the user's account name
        - Contain characters from three of the following four categories:
            - English uppercase characters (A through Z)
            - English lowercase characters (a through z)
            - Base 10 digits (0 through 9)
            - Non-alphabetic characters (for example, !, $, #, %)
- The Database Engine of Microsoft SQL Server 2005 supports the enforcement of password policies for SQL Server logins based on the policies of the underlying Operating System. This option can be configured using the ALTER or CREATE LOGIN command for each login (parameter CHECK_POLICY=on). Though this feature has not been considered as a Security Function during the evaluation it shall be used to enforce the quality of passwords as it was considered during the evaluation of Windows 2003 Server (Please see [WIN_ADMIN] for further details).

## 7.5 Other Recommendations and requirements

- The TOE supports connections via the VIA protocol. However this connection protocol has not been considered during the evaluation. Thus this protocol (which is disabled by default) should not be enabled.
- The TOE has an integration of the common language runtime (CLR) component of the .NET Framework for Microsoft Windows. (see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/denet9/html/7be9e644-36a2-48fc-9206-faf59fdff4d7.htm]) This means that one can write stored procedures, triggers, user-defined types, user-defined functions (scalar and table-valued), and user-defined aggregate functions using any .NET Framework language, including Microsoft Visual Basic .NET and Microsoft Visual C#. This feature is disabled by default. As it has not been within the scope of the evaluation the administrator shall not enable this feature.
- Per default the connections to the database engine are not encrypted and the encryption features of the Database Engine of Microsoft SQL Server 2005 have not been considered during the evaluation. Thus the administrator and the users have to ensure that their connections to the database engine are appropriately protected, e.g. by using an encrypted connection or by using a physically secured connection.
- The use of the column data types text/ntext and image is a deprecated feature (see also [AGD, ms-help://MS.SQLCC.v9/MS.SQLSVR.v9.en/tsqlref9/html/b0d8769c-7598-4f97-8162-ace5f182b5bc.htm]) and has not been considered during the evaluation and certification process with respect to the access control functionality. Therefore the administrator shall ensure that user defined objects do not use this data type. The following SQL query can be used to show all columns that use this data type within the current database.
  *select b.name, a.name from sys.columns a inner join sys.objects b on a.object_id = b.object_id where b.is_ms_shipped=0 and (a.user_type_id=35 or a.user_type_id=99 or a.user_type_id=34)*

# 8 Appendix

## 8.1 Security Characteristics of the COA


**Figure 21: Generic COA**

To confirm that you have genuine Microsoft software, look for the Certificate of Authenticity (COA) affixed to the top of the retail packaging. A COA should always accompany the product with which it is associated. COAs cannot be purchased separately.

Each COA features the product name printed on the label, as well as a background image created by the repeated words "MICROSOFT CERTIFICATE OF AUTHENTICITY." These words are blue letters on a white background in a distorted curve.

The right side of your Certificate of Authenticity contains a transparent porthole with individual paper fibers visible around the inside edge. Interwoven vertically through this porthole is a metallic thread with well defined letters that read "GENUINE." One way to check if your COA is genuine is to gently rip the edge of the COA to determine that the thread is actually interwoven into the fibers of the label rather than printed on top. The background surrounding the transparent porthole contains the words "RETAIL COA" as blue letters on a white background in a distorted curve.

## 8.2 Checking digital signature

Microsoft signs all files of products which are of one of the following types: *.cab , *.cat, *.ctl, *.dll, *.exe, *.ocx. Having said that the end user is easily able to verify whether a file which has been downloaded from the internet or which has been installed on the machine of the user as part of an installation process is an authentic Microsoft file.

The following paragraphs and figures show the process to check a digital signature using the sqlservr.exe as an example. One has to open the properties dialog of the file one wants to check. This can be easily done by clicking on the file with the right mouse button as to see in the following figure.

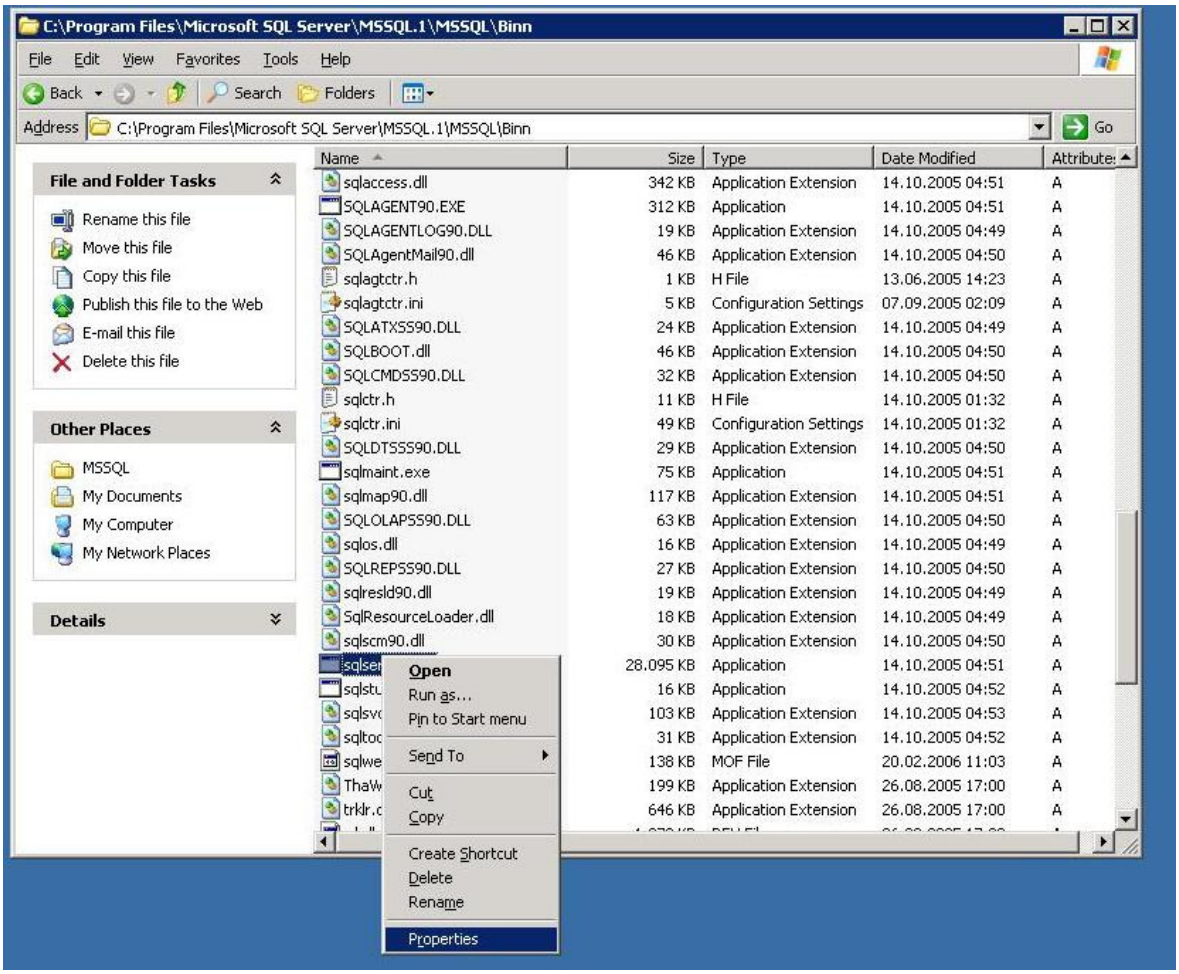


**Figure 22: Verifying the digital signature of “sqlservr.exe” (I)**

The properties dialog box which shows up then can be found in the following figure.

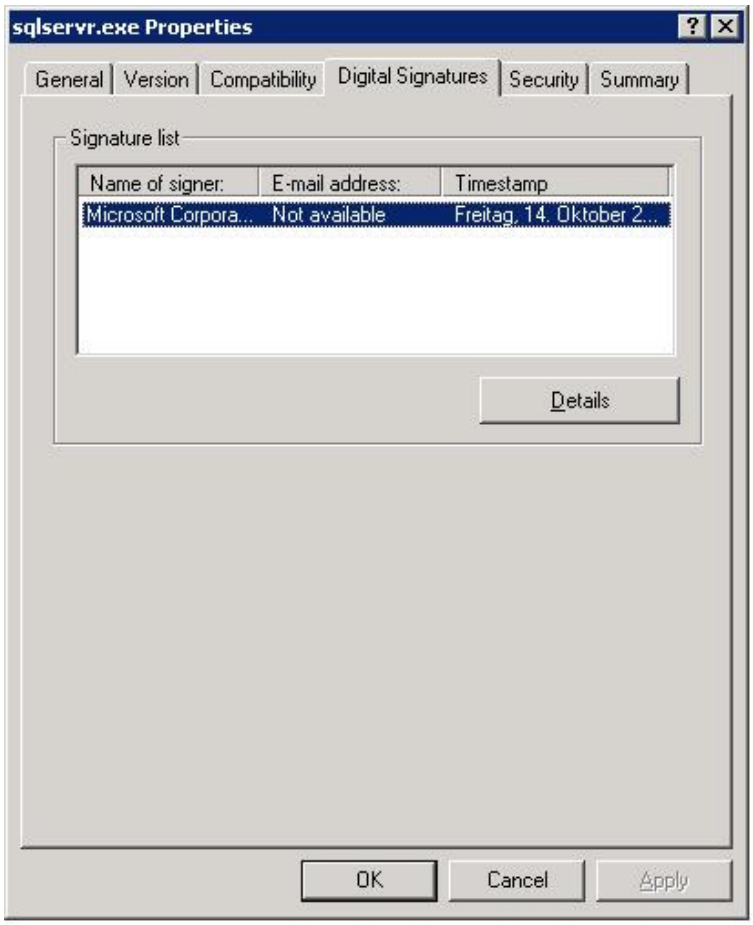


**Figure 23: Verifying the digital signature of “sqlservr.exe” (II)**

The existence of the tab “Digital Signatures” shows that the file contains at least one digital signature. For a file without a digital signature this tab would not exist. By clicking the button “Details” the following dialog with more details about the signature shows up.

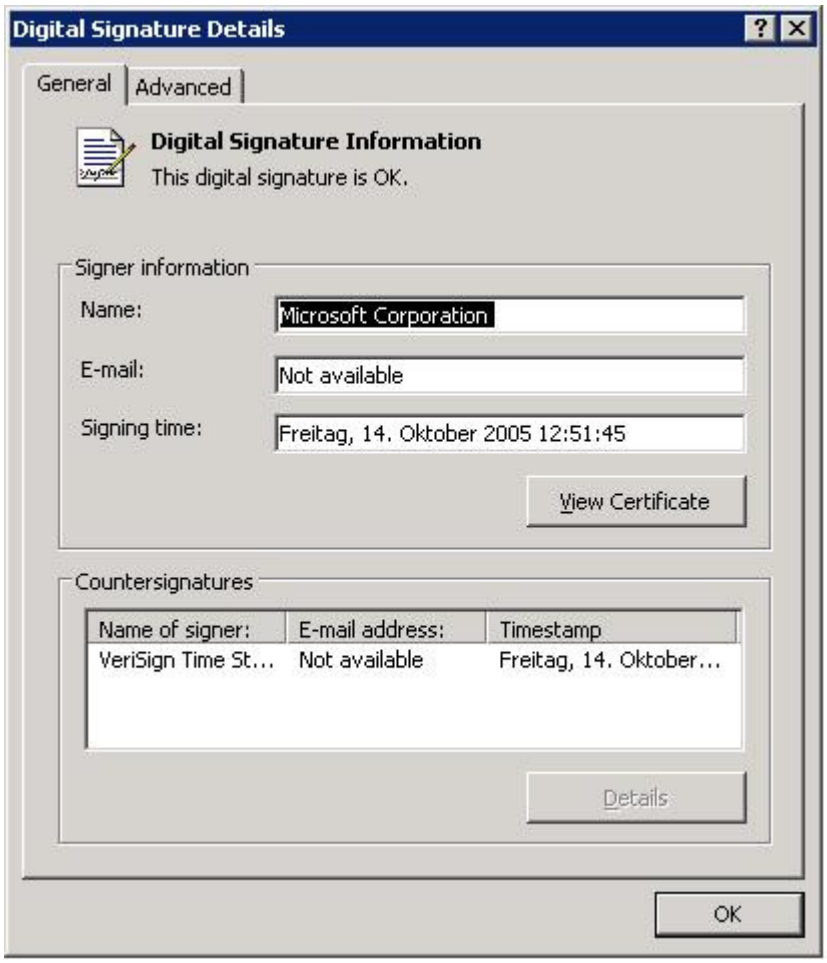


**Figure 24: Verifying the digital signature of “sqlservr.exe” (III)**

This dialog now shows that the digital signature is a valid signature “This signature is OK”. Only if this statement is shown here the user can be sure that the digital signature of the file is valid and that the file has not been changed since it has been signed.

It also shows the signature of the Trustcenter (Verisign) which guarantees for the identity of the signing party (Microsoft).

## 8.3 Checking the hash value

The integrity of the following deliverables can be checked using the FCIV tool:

- The installation media for SQL Server 2005 Enterprise Edition
- The installation package for SP 2
- The installation package for GDR4
- The guidance [AGD]
- The guidance addendum [AGD_ADD] (which is this document)

- The permission_hierarchy.zip (see also chapter 7.2)
- EAL4_trace.sql
- Install_cc_triggers.sql

Via [WEB] the administrator is advised to check the hash values of the guidance addendum (this document) and of the archive that contains the hash files (SQL2005SP2_EAL4_Hashes.zip). The further description in this chapter assumes that this has been successfully done.

The file SQL2005SP2_EAL4_Hashes.zip [WEB] contains the hash values for the deliverables mentioned before. For reasons of convenience all hash value files have been packed into a single ZIP-file and have to be extracted before they can be used.

To verify the integrity of the deliverables the end user should use the following commands:

**For the CD 1, 2 and the DVD of the SQL Server 2005 Enterprise Edition**

- fciv.exe -v -bp x:\ -sha1 -xml sql_2005_dvd.xml

where x:\ has to be substituted by the drive letter of the local CD-Rom/DVD drive and sql_2005_dvd.xml by sql_2005_disk1.xml or sql_2005_disk2.xml for the case that the CDs shall be verified.

**For all other files:**

- fciv.exe -v x:\file.exe -sha1 -xml file.xml

where x:\file.exe has to be substituted by the name and complete path of file to be verified and file.xml has to be substituted by the corresponding XML file. The correct syntax and the correct names are very important in this context as the verification of the hash values may lead to wrong results. Specifically the fciv tool does not warn the user if the file that he is trying to verify does not exist.

All these commands can be used from the command prompt of Windows. It has to be ensured that the file “fciv.exe” and the xml-databases are in the current working directory or the directories with these files have been added to the PATH variable of the Windows OS.

## 8.4 Stored Procedures

The following chapters contain information on Stored Procedures that are contained in the Database Engine of Microsoft SQL Server 2005 but not documented in [AGD].

All these Stored Procedures have been developed for internal use only and are documented for information purposes only. These Stored Procedures are not officially supported by Microsoft and no future compatibility is guaranteed.

### 8.4.1 sp_MSgetversion

This Stored Procedure can be used to get the current version of the Database Engine of Microsoft SQL Server 2005.

Input: no input parameters

Returns: 0 / Error number

Output: row(s) with the Version Number in Character_value

Syntax: exec sp_MSgetversion

### 8.4.2 xp_dirtree

Returns a complete listing of all subdirectories on the server; for each subdirectory listed its depth in the directory tree is also returned. If a *depth* is specified then only subdirectories up to and including the specified depth will be returned. If *IncludeFiles* is specified (as a 1) then files will also be returned and the result set will include an additional column to indicate if a row is a file or a directory.

Input: @filepath, @depth, @IncludeFiles

Output: subdirectory, depth, file

Note: file is only displayed if @IncludeFiles = 1

Permission If the calling user is 'sa' this Stored Procedure is executed in the context of the SQL Server system account. In all other cases the Stored Procedure will be executed in the context of the calling user (i.e. the Stored Procedure will impersonate the user). This impersonation will fail for the case that a SQL login is used and an empty set will be returned.

Syntax: xp_dirtree <filepath>, <depth>, <IncludeFiles>

Examples: exec xp_dirtree 'c:' - Lists all dirs and sub-dirs on C:

exec xp_dirtree 'c:', 1 - Lists all dirs at the root level of C:

exec xp_dirtree 'c:', 1, 1 - Lists all dirs and files at the root level of C:

### 8.4.3 xp_fileexist

This Stored Procedure can be used to determine whether a particular file exists on disk or not.

Input: <filename>

Result: 0 / Error number

Permission If the calling user is 'sa' this Stored Procedure is executed in the context of the SQL Server system account. In all other cases the Stored Procedure will be executed in the context of the calling user (i.e. the Stored Procedure will impersonate the user). This impersonation will fail for the case that a SQL login is used and an empty set will be returned.

Syntax: EXECUTE xp_fileexist <filename> [, <file_exists INT> OUTPUT]

Example: For example, to check whether the file boot.ini exists on disk c: or not, run:

EXEC master..xp_fileexist 'c:\boot.ini'

### 8.4.4 xp_fixeddrives

Returns a row for each fixed drive containing the drive name and the amount of disk space available in MB.

Input: no input parameters

Output: (two columns – drive, MB free)

Permission If the calling user is 'sa' this Stored Procedure is executed in the context of the SQL Server system account. In all other cases the Stored Procedure will be executed in the context of the calling user (i.e. the Stored Procedure will impersonate the user). This impersonation will fail for the case that a SQL login is used and an empty set will be returned.

Syntax: exec @retval=xp_fileexist

Example: To see the list of drives, run:

EXEC master.xp_fixeddrives

### 8.4.5 xp_getnetname

This extended stored procedure returns the WINS name of the SQL Server that you're connected to.

Input: no input parameters

Output: (optional) one column (Server Net Name)

Else single-row, single-column result set is returned

Syntax: exec @retval=xp_getnetname

### 8.4.6 xp_MSADEnabled

This Stored Procedure can be used to determine whether the server is on Win NT4 SP5 or later with AD enabled

Input: no input parameters

Result: 0 / Error number

Output: if platform = win32_nt then

if version > 4 then

if service pack version > 4 then

return TRUE;

### 8.4.7 xp_qv

This Stored Procedure wraps SQLBOOT's QueryProductValue function.

USAGE: xp_qv '<setting>' [, '<instancename>']

If the optional instance name is not provided, then the default instance ('MSSQLSERVER') is assumed.

RETURNS: A signed int return value from QueryProductValue or VALUE_ERROR (-1), if an error occurred. VALUE_NOT_FOUND (-2) is returned if the input value is not a valid VALUE_* const.

Example: declare @sqlbootvalue int

exec @sqlbootvalue = xp_qv '2745196162'

select @sqlbootvalue 'VALUE_REPLICATION'

### 8.4.8 xp_instance_regread

See xp_regread for details

### 8.4.9 xp_regread

Functionality: This Stored Procedure is used to read from the registry.

Input: @rootkey, @key, @value_name, [,@value] (can have 5 input parameters)

Comments: Error if <2 input parameters

5th param – “no_output” then no output is displayed

No error check if >5 params are given

Permission If the calling user is ‘sa’ this Stored Procedure is executed in the context of the SQL Server system account. The Stored Procedure ensures that other users are only granted access to a limited set of registry values.

Return: 0/ Error number

Syntax: EXECUTE xp_regread [@rootkey=]'rootkey', [@key=]'key' [, @value_name=]'value_name'] [, [@value=]@value OUTPUT]

Example: To read into the variable @test from the value 'TestValue' from the key 'SOFTWARE\Test' from the 'HKEY_LOCAL_MACHINE', run:

DECLARE @test varchar(20)

EXEC master..xp_regread @rootkey='HKEY_LOCAL_MACHINE', @key='SOFTWARE\Test', @value_name='TestValue', @value=@test OUTPUT

SELECT @test

## 8.5 References

| Reference | Title | Version | Date |
|---|---|---|---|
| [ST] | Security Target<br>Microsoft SQL Server 2005 Database Engine Common Criteria Evaluation | 1.27 | 2008-07-23 |
| [PP] | U.S. Government Protection Profile for Database Management Systems in Basic Robustness Environments | 1.1 | 2006-06-07 |
| [AGD] | SQL Server Books Online | | February 2007 |
| [MSDN] | Microsoft Developer Network, CD/DVD Version, , Microsoft Corp. | | May 2006 |
| [WIN_CONF] | Windows Server 2003 Security Configuration Guide, Microsoft | 1.0 | September, 22nd 2005, |
| [WIN_ADMIN] | Windows Server 2003 Evaluated Configuration Administrator's Guide, Microsoft | 1.0 | September, 21st 2005 |
| [WEB] | https://www.microsoft.com/sql/commoncriteria/2005/sp2/default.mspx | | |