Omega Core Audit ™

For MS SQL Server

August, 2025

DATAPLUS

Access control, audit monitoring and protection in real time

Omega Core Audit

For MS SQL Server

Evaluation Guide

1.0.0

www.dataplus-al.com

TABLE OF CONTENTS

1 Introduction

1.1 About this Evaluation Guide

This guide is intended to shortly explain the functionalities of the Omega Core Audit for MS SQL Server, once it has been deployed. This is performed using a database evaluation environment, a set of audit/protection policies created for the evaluation and also a benchmarking application.

Instructions for deploying Omega Core Audit are outlined into the Omega Core Audit Deployment Guide. Common instructions are found into the Omega Core Audit User's Guide.

1.2 Functionalities Evaluated

The Omega Core Auditing top functionalities (and evaluated in this guide) are:

Access Control

Evaluated is the server's entrance mandatory access control on all connections to the server. This is tested by the Logon test option of the Benchmark software.

Audit

Evaluated is the auditing of the system for user activity, user statements and operations on objects. This is tested by all testing options.

Real-Time Protection DDL

Evaluated is the Structural Change DDL (Data Definition Language - CREATE, ALTER, GRANT, DROP…) audit and protection. This is tested by the DDL test option.

1.3 General Prerequisites

Before the benchmark testing for evaluation, the following requirements must have been met:

- Omega Core Audit for MS SQL must have been installed successfully.
- The Omega Core Audit advised After Install actions must have been committed.
- The database evaluation environment must have been installed.
- The evaluation policies must have been installed.

Check 6.1 Appendix A. Evaluation Policies for a full description of Evaluation policies.

1.4 Omega Core Audit Evaluation Package

Omega Core Audit comes with a full Evaluation Package which is comprised of:

- A database evaluation environment, sample database users and objects.
- Policies created for the evaluation.
- A benchmark application that generates database activity for the evaluation.

Privilege Requirements!

The evaluation environment and the benchmark policies creation must be performed with **SYSADMIN** privileges!

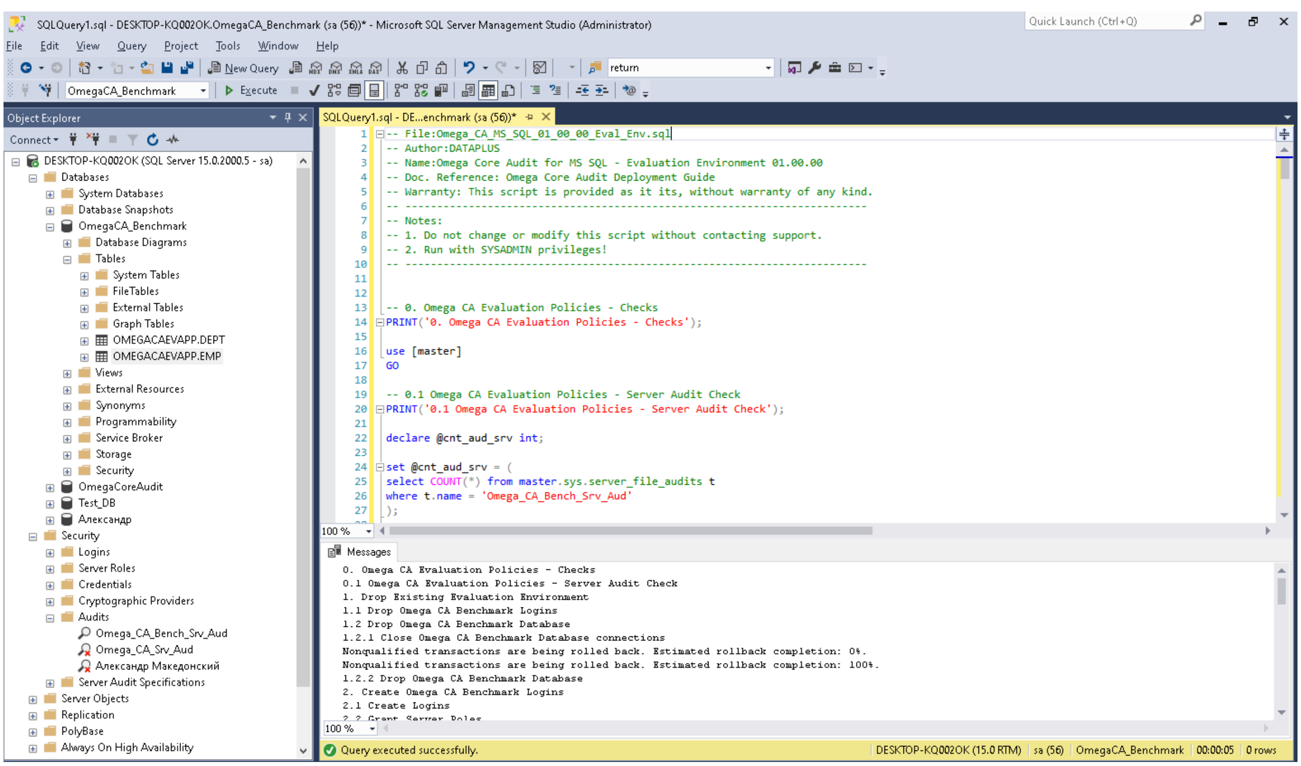
1.4.1 Database Evaluation Environment

The database evaluation environment consists of a MS SQL database, server logins and mapped DB users, and objects created, simulating in miniature the classical infrastructure: a common Application Schema Owner containing application data, code logic, and working in DB on behalf of all application users; two Developer (privileged) Users and one SYSADMIN User.

Install

To install the database evaluation environment, logon on the target server and execute the Evaluation Environment Install script in a SSMS query window:

Omega_CA_MS_SQL_[VS]_[MN]_[PT]_Eval_Env.sql



Press the Execute (F5) button in SSMS!

Check the content of the Messages tab after the install. Normally there should be no errors!

This script will install create the evaluation database OmegaCA_Benchmark, and the following:

OMEGACAEVAPP	Application Schema Owner
OMEGACAEVDEV1	First Developer User
OMEGACAEVDEV2	Second Developer User
OMEGACAEVDBA	SYSADMIN User

Objects will be created for OMEGACAEVAPP Application Owner and sample data will be inserted. These objects you will notice on the Benchmark Application.

Privileges on the OMEGACAEVAPP's objects are granted to the two Developers, OMEGACAEVDEV1 and OMEGACAEVDEV2. OMEGACAEVDBA is granted the SYSADMIN role.

Uninstall

An implicit Uninstall of any currently existing evaluation environment is performed at the start of the Install. In a first Install, these Uninstall actions will result in errors!

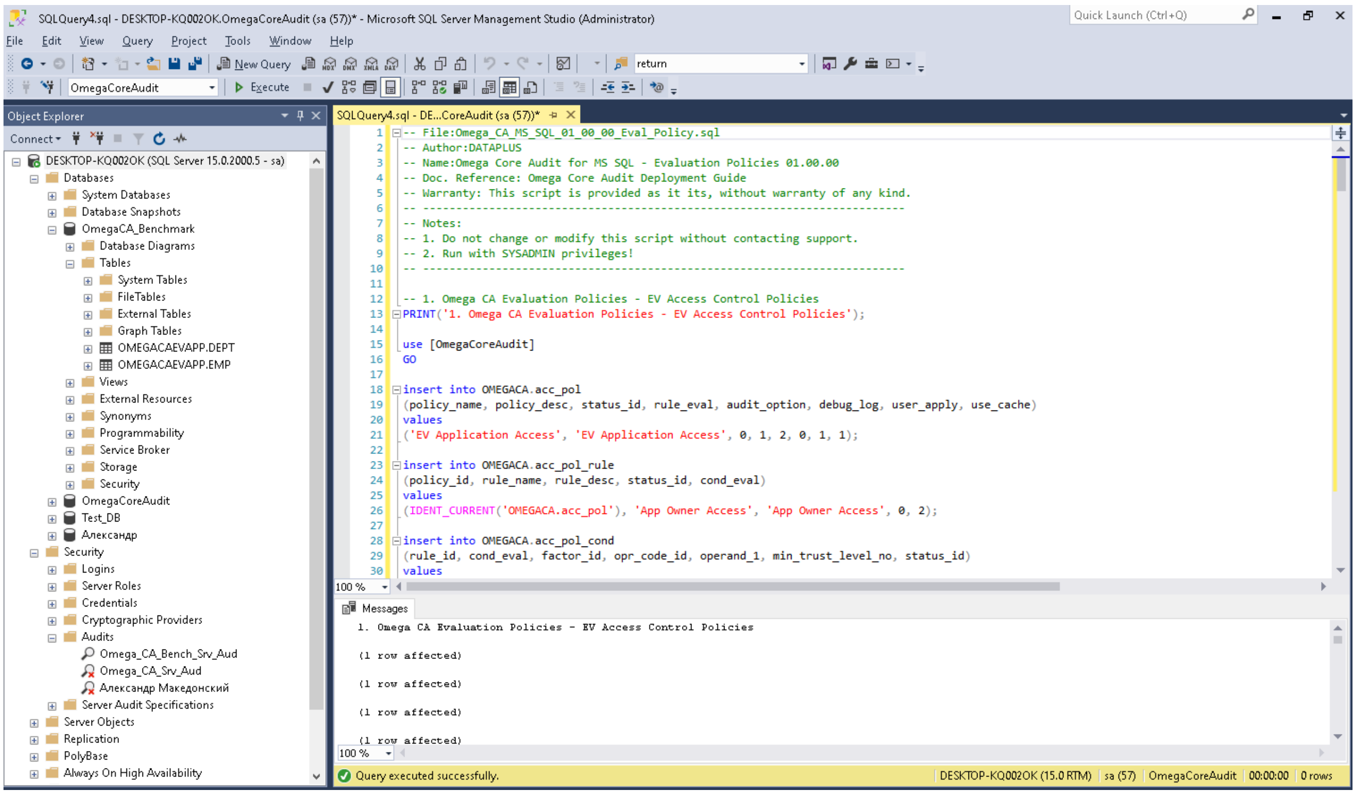
1.4.2 Evaluation Policies

Policies created purposely for the evaluation are made available as an install routine, rather than entering them manually from the Omega Core Audit Application.

Install

To install the evaluation policies into the Omega Core Audit Repository, logon on the target server and execute the Evaluation Policies script in a SSMS query window:

Omega_CA_MS_SQL_[VS]_[MN]_[PT]_Eval_Policy.sql.



Press the Execute (F5) button in SSMS!

Check the content of the Messages tab after the install. Normally there should be no errors!

The evaluation environment install script will create the following evaluation policies into the Omega Core Audit Repository:

Access Control:	"EV Application Access", "EV Developer Access" and "EV DBA Access"
Audit:	"Omega_CA_Bench_DB_Aud_Spc" – a DB Audit Specification (and details)
RTP DDL:	"EV Application DDLs", "EV Security and Audit" and "EV Logins and Roles"

Important Notes:

1. All evaluation policies are created as Inactive, so are the Rules and Conditions. You will manually activate them during this evaluation guide.
2. All evaluation policies are created with a prefix of "EV" (for evaluation) in their name and description.

3. Where Client IP Factor is used, the provided value of 192.168.x.x should be set as the application Benchmark's machine IP address.
4. You can easily convert these evaluation policies to real-life cases.

Uninstall

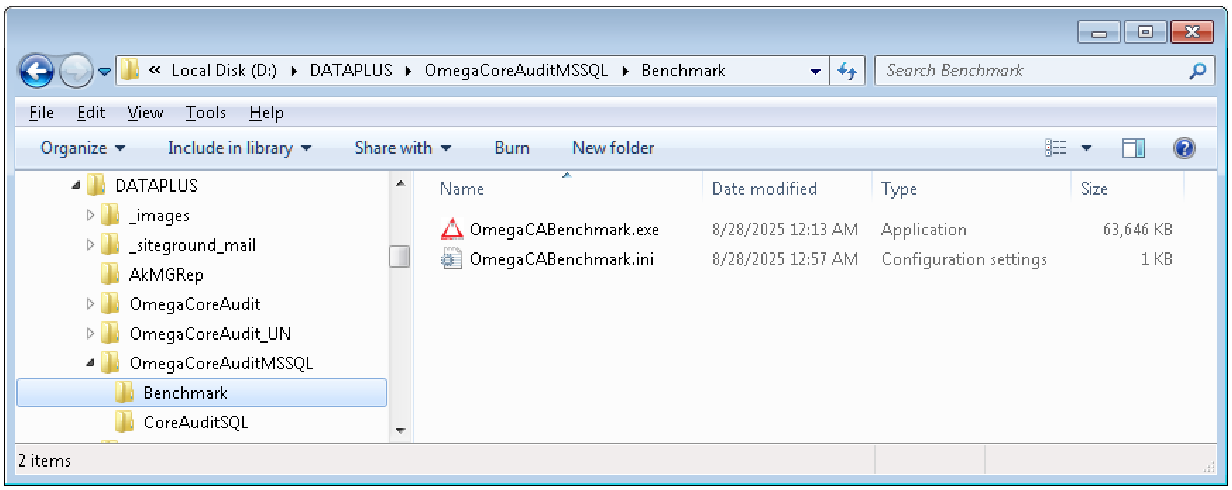
There is no uninstalling routine for the evaluation policies; you can delete them in the Policy forms of Omega Core Audit's each module.

1.4.3 Omega Core Audit Benchmark

The Omega Core Audit Benchmark has been purposely written for testing and benchmarking the Omega Core Audit solution. It generates database activity related to the evaluation environment and evaluation policies, allowing the testing of the Audit and Protection modules of the Omega Core Audit.

Deployment

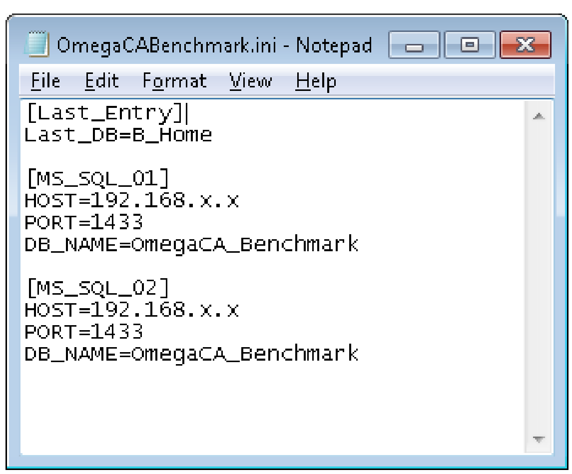
The Omega Core Audit Benchmark software is part of the Omega Core compressed Install package, the software files are found under a single directory named Benchmark. Technology used is the same as that of the Omega Core Audit Application and so is connection to the MS SQL server. There is no installation routine in the proper sense, no installer and no registry entries created, just the files you are extracting from the compressed package.



The application files deployed in this version are:

OmegaCABenchmark.exe	The application's executable binary file
OmegaCABenchmark.ini	System parameters initialization file

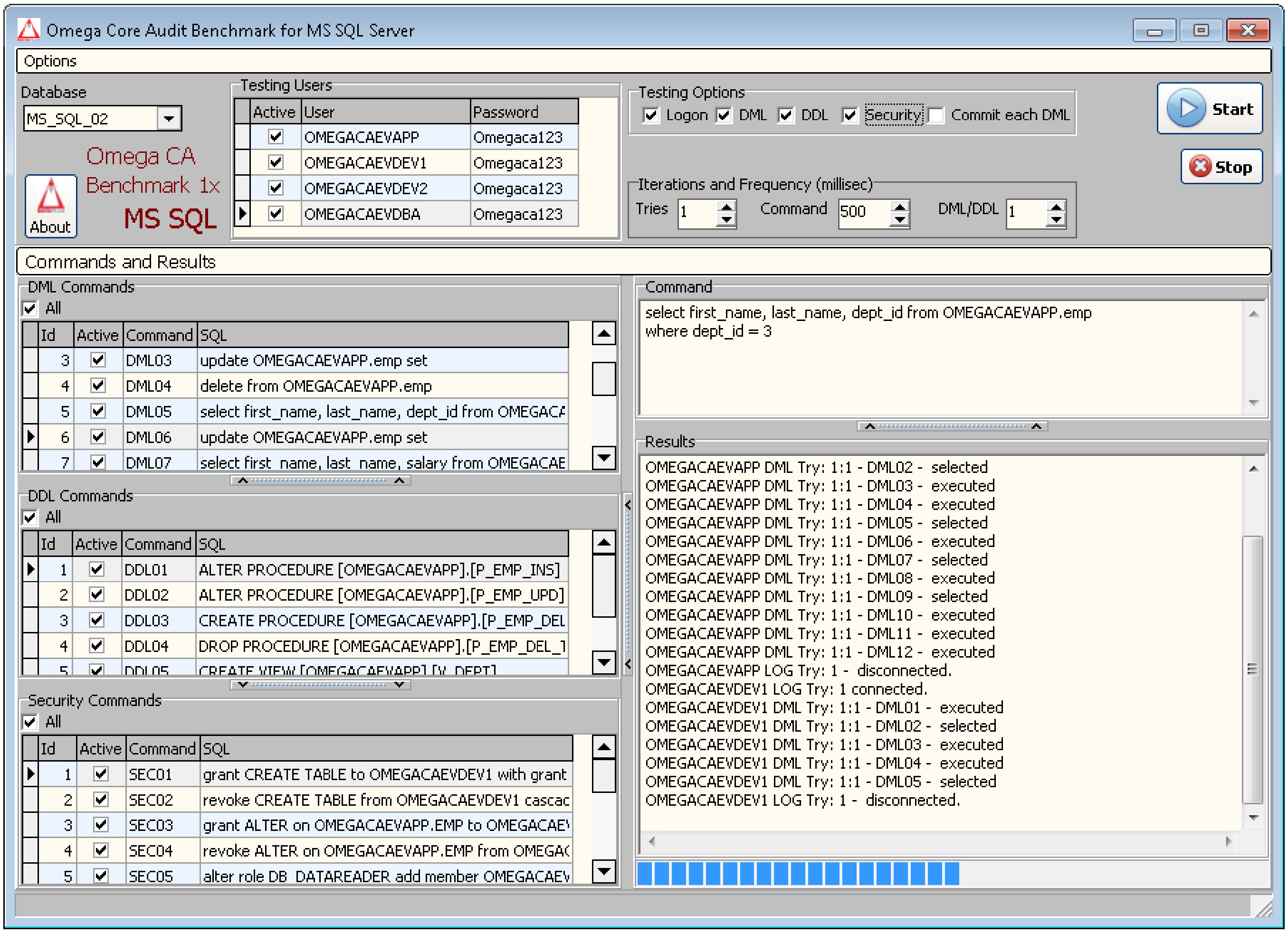
Before you first connect to the target server[s] with the Omega Core Audit Benchmark application, you need to configure the target server[s] section entries in the OmegaCABenchmark.ini initialization parameter file.



OmegaCABenchmark.ini comes with two target Server connection sections, "MS_SQL_01" and "MS_SQL_02"

The application

To launch the application, double-click on the OmegaCABenchmark.exe (or its shortcut).



The application's main form contains two panels:

- Options — Benchmark testing options
- Commands and Results — Commands used for testing and result output

The first panel Options features:

Database

Target server where testing is performed. The Database combo box loads the target server section entries configured in the initialization file OmegaCABenchmark.ini. The last successfully connected is persisted on the next application run.

Testing Users

The four logins mapped to the DB users of the database evaluation environment; least one must be checked (Active) for testing.

Testing Options

This group box contains the following checkboxes optioning for:

Logon	The logon/logoff activity generated for the benchmarking of the Access Control and Audit modules. This is the only mandatory option.
DML	Enables the execution of the selected DML commands for the benchmarking of the Audit module.
DDL	Enables the execution of the selected DDL commands for the benchmarking of the Audit and Real-Time Protection DDL modules.
Security	Enables the execution of the selected security-related commands for the benchmarking of the Audit and Real-Time Protection DDL modules.
Commit Every DML	Enables Commit on every DML command performed.

Iterations and Frequency

This GroupBox contains the following options:

Tries	Number of tries the set of Benchmarking commands (Logon, DML and DDL) are run as a whole for each user.
Command	Pause in milliseconds on each command, default 500, could be higher for real-life conditions, but can be set lower (or 0) for hard-benchmarking!
DML/DDL	The ratio number of times DML Set will be repeated versus the DDL one. Default 1 set (for testing purposes), but to simulate real conditions DML ratio should be much higher *.

* In real-life conditions DDLs do happen much rarely compared to the DMLs!

The second panel Commands and Results features:

DML Commands

The DML Commands grid contains the DML commands used in this Benchmark testing. Commands used are SELECTs, INSERTs, UPDATEs and DELETEs performed on the evaluation environment EMP table, either directly, or through views and stored procedures. Least one must be checked (Active) for testing if DML benchmarking.

DDL Commands

The DDL Commands grid contains the DDL commands used in this Benchmark testing. Commands used are ALTERs and CREATEs performed on the evaluation environment schema procedures, views and functions. Least one must be checked (Active) for testing if DDL benchmarking.

Security Commands

The Security Commands grid contains the security-related commands used in this Benchmark testing. Commands used are GRANTs and REVOKEs performed on the evaluation environment users and objects for System, Object and Role privileges; and also AUDITs and NOAUDITs performed for System and Object privileges. Least one must be checked (Active) for testing if Security benchmarking.

Command

The current SQL benchmark command being executed.

Results

This memo shows the output of all the benchmarked commands.

After the benchmark testing options are set, click on the Start button to initialize the benchmark testing.
Best do not interact with the form while the test is running (especially with any benchmark test option!).
Only exception to the rule above is the stopping the Benchmark test by use of the button Stop.

1.5 Evaluation Reminders

1. The evaluation described later in this guide is cumulative, so while testing the current module, the settings of the previous one might generate Security Events. To simplify testing, when searching Security Events, you can set the Policy Type Search Option field to retrieve only records produced by the module being currently evaluated.

2. The evaluations performed below in this guide are exercised by keeping the default value of Tries = 1 (one try per user)! This parameter logically impacts in proportion the number of Security Events produced - frequently referred to below in this guide.

3. The Benchmark application might “freeze” during testing and long operations, but it will be working instead. Please wait until the operation completes and do not interrupt!

4. For immediate results of Audit activity on Security Events trail, you can set a shorter DB Audit Trails Collect interval run, other than the default of 60 seconds.

2 Evaluating the Access Control

This chapter is a walkthrough on the Access Control module implementation administered through the Omega Core Audit Application and benchmarked via the Omega Core Audit Benchmark software.

2.1 Prerequisites

- The Access Control Module has been activated.
- The Access Control Module remains in Silent mode (default in install).

2.2 Benchmark Testing

2.2.1 Initialization

Rationale:
Access Control Module is activated in Silent mode and there is yet no Active Policy.

In the Omega Core Audit Benchmark, Testing Users group, check all the four Users. Check Testing Option Logon.

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see an Access Control security event (Policy Type equal to Access Control) generated for each testing user. In the Record Details view, check the Trail Evaluation field for each record. It will be empty because there is no policy activated yet. The logons in the Benchmark have been successful only because the whole Access Control Module is in Silent mode (module's specific feature and default on install). Although no policy created yet, this is a specific behavior of the Access Control module that when non-compliant (i.e. no policy evaluated TRUE for whatever reason, even non-existing or all Inactive) a security event is created.

2.2.2 Access Control for Application Schema Owner

Rationale:
The "EV Application Access" policy establishes a secure logon channel for the Application Schema Owner connections. Policy is created with User Appliance of type "Users Apply" for user OMEGACAEVAPP only.

In Omega Core Audit open the Access Control policy "EV Application Access". Open its only rule "App Owner Access". Activate the two Operand Conditions of Login Name and Client IP. For the later set the right value instead of the provided 192.168.x.x. Activate the rule "App Owner Access". Activate the policy "EV Application Access".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see an Access Control security event generated for each testing user. In the Record Details view, check the Trail Evaluation field for each record. It will have a value only for the security event of OMEGACAEVAPP user and empty for the other three. The policy Id will be the one of the "EV Application Access". The policy Result will be TRUE.

2.2.3 Access Control for Developers

Rationale:
The "EV Developer Access" policy establishes a secure logon channel for the two Developer Users. Policy is created with User Appliance of type "Users Apply" for users OMEGACAEVDEV1 and OMEGACAEVDEV2.

In Omega Core Audit open the Access Control policy "EV Developer Access". Open its rule "Developer 01 Access". Activate the two Operand Conditions of Login Name and Client IP. For the later set the right value instead of the provided 192.168.x.x. Activate the rule "Developer 01 Access". Activate the policy "EV Developer Access".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see an Access Control security event generated for each testing user. In the Record Details view, check the Trail Evaluation field for each record. It will have a value even for the security event of OMEGACAEVDEV1 user and empty for the other two. The policy Id will be the one of the "EV Developer Access". The policy Result will be TRUE.

Open the rule "Developer 02 Access". Activate the two Operand Conditions of Login Name and Client IP. For the later set the right value instead of the provided 192.168.x.x. Activate the rule "Developer 02 Access".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see an Access Control security event generated for each testing user. In the Record Details view, check the Trail Evaluation field for each record. It will have a value even for security event of OMEGACAEVDEV2 user and empty for the last OMEGACAEVDBA. The policy Id will be the one of the "EV Developer Access". The policy Result will be TRUE.

2.2.4 Access Control for DBA

Rationale:
The "EV DBA Access" policy establishes a secure logon channel for the DBA User OMEGACAEVDBA. Policy is created with User Appliance of type "Users Apply" for user OMEGACAEVDBA only.

In Omega Core Audit open the Access Control policy "EV DBA Access". Open its rule "DBA Access". Activate the two Operand Conditions of Login Name and Client IP. For the later set the right value instead of the provided 192.168.x.x Activate the rule "DBA Access". Activate the policy "EV DBA Access".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see an Access Control security event generated for each testing user. In the Record Details view, check the Trail Evaluation field for each record. It will have a value even for the security event of OMEGACAEVDBA user. The policy Id will be the one of the "EV DBA Access". The policy Result will be TRUE.

2.2.5 Tuning the Access Control

Rationale:
Optimize Access Control for auditing knowledgeably and performance.

Having all the Access Control policies above with an Audit Option of "On Success/Failure" might not be a good idea for the Application Access policy. For example, behind the Application Schema Owner MS SQL account may be hundreds of application users and new sessions might be (application's developer choice) created at every form open, or refresh, in case of web-design. Having the policy "EV Application Access" on such a configuration could generate hundreds if not thousands of repetitive log entries per minute, so it would be logically to have the security event only when the policy is not satisfied.

On the Access Control policy "EV Application Access", set the Audit Option to "On Failure".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will notice that Access Control security event will be created for all users, except OMEGACAEVAPP. Thus we shall avoid the thousands correct logons of the Application Schema Owner, but have all the violations of it!

Note:

Using the "Users Apply" option of the Access Control policies User Appliance setting is a recommended way of reducing even the scope of the policies evaluation; this is further reinforced by the "mandatory" specific of the Access Control. Its counterpart "Users Exclude" would achieve the same goals.

2.2.6 Enforcing the Access Control

Rationale:
Enforce Access Control on the server by rejecting non-complying logons.

Until now we have used the Access Control module in its default Silent Mode. This means the logon actions were allowed to continue whatever the compliance evaluation (Policy Result TRUE/FALSE) was.

To test the protective capabilities of the Access Control module, into the Omega Core Audit software, System Components form, Access Control tab, uncheck the Silent Checkbox and press the Set button to set the changed value. Also change the "Developer 02 Access" rule of the "Developer Access" and wrongly set the Client Host Operand Condition's value, so that rule evaluates to FALSE.

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

You will notice the logon failure "Logon failed for login 'OMEGACAEVDEV2' due to trigger execution" for the OMEGACAEVDEV2 user. The OMEGACAEVDEV2 logon action has been rejected!

In Omega Core Audit, search the Security Events trail. You will notice the failure in the Security Events trail fields Succeeded (0 - unchecked) and Return Message ("Access Control Policy non-compliance") for OMEGACAEVDEV2 user. At field Trail Evaluation the policy Id will be the one of the "EV Developer Access". The policy Result will be FALSE.

Note

If you will continue the testing with Silent Mode unset, make sure to have rightly ensured the logon of the four testing users, so that they continue perform other module's commands.

The same is required for the OMEGACAADM (Omega Core Audit pre-configured Administrator) user, needed for operating on the Omega Core Audit Application.

3 Evaluating the Audit

This chapter is a walkthrough on the Audit module implementation administered through the Omega Core Audit Application and benchmarked via the Omega Core Audit Benchmark software.

3.1 Prerequisites

- The server audit Omega_CA_Srv_Aud has been created (part of Omega Core Audit install).
- The server audit Omega_CA_Bench_Srv_Aud has been created.
- The server audit specification Omega_CA_Srv_Aud_SPC has been created, and is disabled
- The database audit specification Omega_CA_Bench_DB_Aud_Spc has been created in database OmegaCA_Benchmark, and is disabled

3.2 Benchmark Testing

3.2.1 Initialization

Rationale:
There is yet no active audit. Both audit specifications – on server and database – are disabled.

Enable the Omega_CA_Bench_Srv_Aud database audit specification on database OmegaCA_Benchmark.

3.2.2 Audit at Application Schema Owner

Rationale:
Test audit on Application schema owner.

Select user OMEGACAEVAPP only in the benchmark application. Check Testing Options DML and DDL. Check all DML and DDL commands. Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

Wait for the next run of the DB Audit Trails Collect.

In Omega Core Audit, search the Security Events trail and note events of OMEGACAEVAPP.

3.2.3 Audit for Developer Users

Rationale:
Test audit on Developer users.

Select users OMEGACAEVDEV1 and OMEGACAEVDEV2 only in the benchmark application. Check Testing Options DML and DDL. Check all DML and DDL commands. Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

Wait for the next run of the DB Audit Trails Collect.

In Omega Core Audit, search the Security Events trail and note events of OMEGACAEVDEV1 and OMEGACAEVDEV2.

3.2.4 Audit for SYSADMIN Users

Rationale:
Test audit on DBA users.

Select users OMEGACAEVDBA only in the benchmark application. Check Testing Options DML and DDL. Check all DML and DDL commands.

Enable server audit specification Omega_CA_Srv_Aud_SPC. Check Testing Option Security. Check all Security commands.

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

Wait for the next run of the DB Audit Trails Collect.

In Omega Core Audit, search the Security Events trail and note events of OMEGACAEVDBA.

3.2.5 Tuning the Standard Audit

Rationale:
Optimize Audit for auditing knowledgeably and performance.

Actions performed on the database by Developers and DBA Users are expected to be infrequent and insignificant comparing to the whole system work. But these actions are exercised by privileged users and as such are worthy for audit.

As for auditing actions launched by the Application Owner itself, using exactly the method described so far would not be a good idea, considering the case that behind the Application Schema Owner MS SQL account may be hundreds of application users and new sessions might be (application's developer choice) created at every form open, or refresh, in case of web-design. Such a configuration could generate hundreds if not thousands of repetitive log entries per minute.

So it would be logical to implement Standard Audit policies of type "Object" only on the objects of importance, and for specific actions performed, as we have done with the policy "Application Objects", on the application owner OMEGACAEVAPP table's EMP. Here the audit would be very reasonable (and set) for DELETE and UPDATE commands - usually (in modern transaction based and data-warehouse systems) expected to be few, or even not at all issued from the Application Schema Owner, and as such of much interest. While not set for the SELECT and INSERT - for which a high volume is generally expected from the Application Schema Owner.

4 Evaluating the Real-Time Protection DDL

This chapter is a walkthrough on Real-Time Protection DDL implementation administered through the Omega Core Audit Application and benchmarked via the Omega Core Audit Benchmark software.

4.1 Prerequisites

- The Real-Time Protection DDL Module has been activated
- The policies are in Silent mode

4.2 Benchmark Testing

4.2.1 Initialization

Rationale:
Real-Time Protection DDL Module is activated and there is yet no Active Policy.

In the Omega Core Audit Benchmark, Testing Options group, check only the DDL Checkbox. Check all the DDL commands in the grid (DDL01-DDL06). Check all the four Users in the grid.

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. There will be no Real-Time Protection DDL records (Policy Type equal to RTP-DDL), because no policy has been activated yet.

4.2.2 Real-Time Protection DDL for Application Objects

Rationale:
The "EV Application DDls" policy establishes a secure channel for DDLs performed by Developers on the Application Owner schema objects. Policy is created with User Appliance of type "All Users".

In Omega Core Audit open the Real-Time Protection DDL policy "EV Application DDls". Open its rule "Developer 01 DDL". Activate the two Operand Conditions of Login Name and Client IP. For the later set the right value instead of the provided 192.168.x.x. Activate the rule "Developer 01 DDL". Activate the policy "EV Application DDls".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see Real-Time Protection DDL security events generated for each testing user. In the Record Details view, check the Trail Evaluation field for each record. The policy Id will be the one of the "EV Application DDls". The policy Result will be TRUE only for the OMEGACAEVDEV1 user and FALSE for all three other users.

Open the rule "Developer 02 DDL". Activate the two Operand Conditions of Login Name and Client IP. For the later set the right value instead of the provided 192.168.x.x. Activate the rule "Developer 02 DDL".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see Real-Time Protection DDL security events generated for each testing user. In the Record Details view, check the Trail Evaluation field for each record. The policy Id will be the one of the "EV Application DDls". The policy Result will be TRUE even for the OMEGACAEVDEV2.

At this stage, only the OMEGACAEVAPP (Schema Owner Itself) and OMEGACAEVDBA's (the DBA) records show evaluate FALSE of the policy, which means are not authorized to perform DDLs on the Application Objects.

4.2.3 Real-Time Protection DDL for Security and Audit

Rationale:
The "EV Security and Audit" policy establishes a secure channel for the Grant and Revoke Security (DDL) commands performed on the database. Policy is created with User Appliance of type "All Users".

In the Omega Core Audit Benchmark, Testing Options group, uncheck the DDL Checkbox and check only the Security Checkbox. Check all the Security. In the Users grid check only the OMEGACAEVDBA user.

In Omega Core Audit open the Real-Time Protection DDL policy "EV Security and Audit". Open its rule "DBA Security". Activate the two Operand Conditions of Login Name and Client IP. For the later set the right value instead of the provided 192.168.x.x. Activate the rule "DBA Security". Activate the policy "EV Security and Audit".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see Real-Time Protection DDL security events generated for the OMEGACAEVDBA user. In the Record Details view, check the Trail Evaluation field for each record. The policy Id will be the one of the "EV Security and Audit". The policy Result will be TRUE.

4.2.4 Real-Time Protection DDL for Logins and Roles

Rationale:
The "EV Logins and Roles" policy establishes a secure channel for any action performed on Server Logins and Roles. Policy is created with User Appliance of type "All Users".

In Omega Core Audit Benchmark uncheck all the Security Commands and check only the last one (SEC11-12).

In Omega Core Audit open the Real-Time Protection DDL policy "EV Logins and Roles". Open its rule "DBA Login-Role". Activate the two Operand Conditions of Login Name and Client IP. For the later set the right value instead of the provided 192.168.x.x. Activate the rule "DBA Security". Activate the policy "EV Users and Roles".

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

In Omega Core Audit, search the Security Events trail. You will see Real-Time Protection DDL security events generated for the OMEGACAEVDBA user. In the Record Details view, check the Trail Evaluation field for each record. The policy Id will be the one of the "EV Logins and Roles". The policy Result will be TRUE.

4.2.5 Tuning the Real-Time Protection DDL

Rationale:
Optimize Real-Time Protection DDL for auditing knowledgeably and performance.

The Real-Time Protection DDL generates insignificant load to the system, considering that DDL (Data definition language) commands are very rare comparing to the rest of system actions. Due to the importance of the subject, such as structural change is, detailed audit is advised, like existing DDL Body and SQL given.

Although full DDL audit can be performed, it is recommended to narrow the scope of the auditing to the areas of interest. This is approached by featuring of the Secured Areas of the Real-Time Protection DDL policies, using the database, DDLs action, object owner, name and type as factors in designing an area best narrowed to your needs.

4.2.6 Enforcing the Real-Time Protection DDL

Rationale:
Enforce Real-Time Protection DDL on the database by rejecting non-complying DDLs.

Until now we have used the Real-Time Protection DDL Policies in their default Silent Deny Mode. This means the DDL actions were allowed to continue whatever the compliance evaluation was.

To test the protective capabilities of the Real-Time Protection module, in the Omega Core Audit software, open the Real-Time Protection DDL policy "EV Application DDls". Uncheck the Silent Deny Checkbox and press the Save button to save the policy.

In the Omega Core Audit Benchmark, Testing Options group, uncheck the Security Checkbox and re-check only the DDL Checkbox. Check all the DDL Commands. Check all the four users.

Remember that in the RTP DDL Policy "EV Application DDls", we have authorized only the two Developers to perform DDLs on the Application Schema Owner, through two respective RTP DDL rules.

Press the Run button in Omega Core Audit Benchmark and wait until it finishes.

You will notice DDL failure "DDL Policy non-compliance!" for the OMEGACAEVAPP and OMEGACAEVDBA users. The Application Owner's itself and the DBAs DDL actions have been rejected.

In Omega Core Audit, search the Security Events trail. You will notice the failure in the Security Events trail fields Succeeded (0 - unchecked) and Return Message ("DDL Policy non-compliance") for users OMEGACAEVAPP and OMEGACAEVDBA. At field Trail Evaluation the policy Id will be the one of the "EV Application DDls". The policy Result will be FALSE.

You are invited to test the same (disable Silent Deny) with the other policies.

5 Summary and Conclusions

The tests described so far in this document represent only a small view of the Omega Core Audit capabilities. The Omega Core Audit's provided evaluation environment and policies account for a usual production system's schema in miniature, while the Benchmark software, tailored for the environment and polices, generates actions on them in automatic mode.

The above is made to demonstrate the basic features of the Omega Core Audit. You are encouraged to try other combinations of audit/protection settings in Omega Core Audit and test the software's behavior by combining the options into the Omega Core Audit Benchmark.

It would be logical that some knowledge from the Omega User's Guide is required, may be not for exactly the above walkthrough guide, but surely for advanced testing options you may perform, or furthermore in your test-simulated environment and finally your production database.

6 Appendixes

6.1 Appendix A. Evaluation Policies

Below are the evaluation policies, as originally created directly on the repository by the provided SQL script.

6.1.1 Access Control

Policy	**Policy Name**	**Description**	**Audit Option**	**User Appliance**	**Rule Eval.**
1.	EV Application Access	EV Application Access	On Success/Failure	Users Apply	Any True
Rules	**Rule Name**	**Rule Description**	**Condition Eval.**		
1.	App. Owner 01 Access	App. Owner 01 Access	All True		
Conditions	**Factor**	**Condition Evaluation**	**Operation Code**	**Operand 1**	**Operand 2**
1.	Session User	Operand	=	OMEGACAEVAPP	
2.	Client IP	Operand	=	192.168.x.x	

Policy	**Policy Name**	**Description**	**Audit Option**	**User Appliance**	**Rule Eval.**
1.	EV Developer Access	EV Developer Access	On Success/Failure	Users Apply	Any True
Rules	**Rule Name**	**Rule Description**	**Condition Eval.**		
1.	Developer 01 Access	Developer 01 Access	All True		
Conditions	**Factor**	**Condition Evaluation**	**Operation Code**	**Operand 1**	**Operand 2**
1.	Session User	Operand	=	OMEGACAEVDEV1	
2.	Client IP	Operand	=	192.168.x.x	
Rules	**Rule Name**	**Rule Description**	**Condition Eval.**		
2.	Developer 02 Access	Developer 02 Access	All True		
Conditions	**Factor**	**Condition Evaluation**	**Operation Code**	**Operand 1**	**Operand 2**
1.	Session User	Operand	=	OMEGACAEVDEV2	
2.	Client IP	Operand	=	192.168.x.x	

Policy	**Policy Name**	**Description**	**Audit Option**	**User Appliance**	**Rule Eval.**
1.	EV DBA Access	EV DBA Access	On Success/Failure	Users Apply	Any True
Rules	**Rule Name**	**Rule Description**	**Condition Eval.**		
1.	DBA 01 Access	DBA 01 Access	All True		
Conditions	**Factor**	**Condition Evaluation**	**Operation Code**	**Operand 1**	**Operand 2**
1.	Session User	Operand	=	OMEGACAEVDBA	
2.	Client IP	Operand	=	192.168.x.x	

6.1.2 Audit

Server audit specification details

Audit Action Name
APPLICATION_ROLE_CHANGE_PASSWORD_GROUP
AUDIT_CHANGE_GROUP
BACKUP_RESTORE_GROUP
BROKER_LOGIN_GROUP
DATABASE_CHANGE_GROUP
DATABASE_OBJECT_CHANGE_GROUP
DATABASE_OBJECT_OWNERSHIP_CHANGE_GROUP
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP
DATABASE_OWNERSHIP_CHANGE_GROUP
DATABASE_PERMISSION_CHANGE_GROUP
DATABASE_PRINCIPAL_CHANGE_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
DATABASE_ROLE_MEMBER_CHANGE_GROUP
DBCC_GROUP
FAILED_DATABASE_AUTHENTICATION_GROUP
FAILED_LOGIN_GROUP
LOGIN_CHANGE_PASSWORD_GROUP
SCHEMA_OBJECT_CHANGE_GROUP

SCHEMA_OBJECT_OWNERSHIP_CHANGE_GROUP
SCHEMA_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_OBJECT_CHANGE_GROUP
SERVER_OBJECT_OWNERSHIP_CHANGE_GROUP
SERVER_OBJECT_PERMISSION_CHANGE_GROUP
SERVER_PERMISSION_CHANGE_GROUP
SERVER_PRINCIPAL_CHANGE_GROUP
SERVER_ROLE_MEMBER_CHANGE_GROUP

Database audit specification details

Audit Action Name	DB User
AUDIT_CHANGE_GROUP	public
DATABASE_OBJECT_PERMISSION_CHANGE_GROUP	public
DATABASE_PERMISSION_CHANGE_GROUP	public
DATABASE_ROLE_MEMBER_CHANGE_GROUP	public
DELETE	public
DELETE	OMEGACAEVDEV2
DELETE	OMEGACAEVDEV1
DELETE	OMEGACAEVDBA
EXECUTE	OMEGACAEVDBA
EXECUTE	OMEGACAEVDEV1
EXECUTE	OMEGACAEVDEV2
INSERT	OMEGACAEVDEV2
INSERT	OMEGACAEVDEV1
INSERT	OMEGACAEVDBA
SCHEMA_OBJECT_CHANGE_GROUP	public
SELECT	OMEGACAEVDEV2
SELECT	OMEGACAEVDBA
SELECT	OMEGACAEVDEV1
UPDATE	OMEGACAEVDEV1
UPDATE	OMEGACAEVDBA
UPDATE	OMEGACAEVDEV2
UPDATE	public

6.1.3 Real-Time Protection DDL

Policy	**Policy Name**	**Description**	**Audit Option**	**User Appliance**	**Rule Eval.**
1.	EV Application DDls	EV Application DDls	On Success/Failure	All Users	Any True
Rule	**Rule Name**	**Rule Description**	**Condition Eval.**		
1.	Developer 01 DDL	Developer 01 DDL	All True		
Condition	**Factor**	**Condition Evaluation**	**Operation Code**	**Operand 1**	**Operand 2**
1.	Session User	Operand	=	OMEGACAEVDEV1	
2.	Client IP	Operand	=	192.168.x.x	
Rule	**Rule Name**	**Rule Description**	**Condition Eval.**		
2.	Developer 02 DDL	Developer 02 DDL	All True		
Condition	**Factor**	**Condition Evaluation**	**Operation Code**	**Operand 1**	**Operand 2**
1.	Session User	Operand	=	OMEGACAEVDEV2	
2.	Client IP	Operand	=	192.168.x.x	

Policy	**Policy Name**	**Description**	**Audit Option**	**User Appliance**	**Rule Eval.**
1.	EV Security and Audit	EV Security and Audit	On Success/Failure	All Users	Any True
Rule	**Rule Name**	**Rule Description**	**Condition Eval.**		
1.	DBA Security	DBA Security	All True		
Condition	**Factor**	**Condition Evaluation**	**Operation Code**	**Operand 1**	**Operand 2**
1.	Session User	Operand	=	OMEGACAEVDBA	
2.	Client IP	Operand	=	192.168.x.x	

Policy	**Policy Name**	**Description**	**Audit Option**	**User Appliance**	**Rule Eval.**
1.	EV Users and Roles	EV Users and Roles	On Success/Failure	All Users	Any True
Rule	**Rule Name**	**Rule Description**	**Condition Eval.**		
1.	DBA Security	DBA Security	All True		
Condition	**Factor**	**Condition Evaluation**	**Operation Code**	**Operand 1**	**Operand 2**
1.	Session User	Operand	=	OMEGACAEVDBA	
2.	Client IP	Operand	=	192.168.x.x	

6.2 Appendix B. Support and Licensing

6.2.1 Support

Omega Core Audit are entitled to:

- new versions
- upgrades and fixes
- new security controls
- online support

SLA:

Response Time SLA:	2 Business Days
Support Call/Online:	09:00 GMT to 21:00 GMT, Monday to Friday
Emergencies:	we are here to help
Resolution Time:	case-related

For product documentation, forum and knowledge base, visit our site:
www.dataplus-al.com/omega-core-audit

For technical issues, comments, ideas and impressions, e-mail us at:
support@dataplus-al.com

Also follow us on the next social media sites where DATAPLUS is present:

YouTube	https://www.youtube.com/channel/UCa59qQuGg5tvd2vIe1MsMOw
LinkedIn	https://www.linkedin.com/company/dataplus-al

6.2.2 Licensing

Omega Core Audit license is perpetual. It allows the customer to use the licensed software indefinitely and not tied to the product version.

Omega Core Auditing for MS SQL Server licenses number is the same as the number of licenses of your MS SQL Server.

For pricing and licensing information, contact our Sale specialists at:
sales@dataplus-al.com