STORMSHIELD

GUIDE

STORMSHIELD ENDPOINT SECURITY EVOLUTION

SQL SERVER RECOMMENDATIONS

Version 2.4.1

Document last updated: May 25, 2023

Reference: ses-en-sql_server_recommendations-v2.4.1

Table of contents

In the documentation, Stormshield Endpoint Security Evolution is referred to in its short form: SES Evolution.

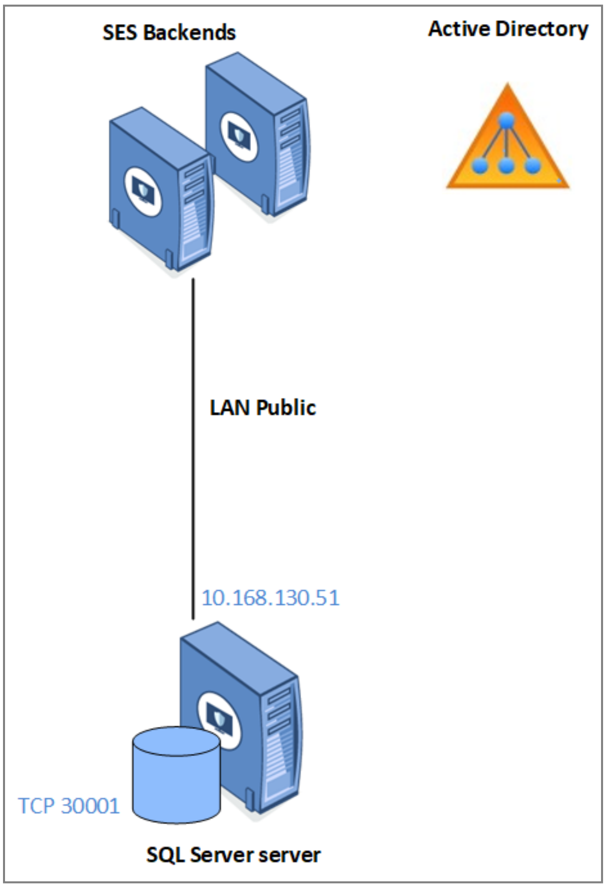
1. Getting started

Welcome to the SQL Server Recommendations Guide for Stormshield Endpoint Security Evolution.

In this document, you will find all the information needed for the installation, configuration and maintenance of a SQL Server instance used with Stormshield Endpoint Security Evolution.

2. Requirements

The components shown below are required in order to build the final architecture.



The IP address is only an example. Your own address range will determine the actual IP address.

2.1 Network

- The architecture is based on Active Directory.
- The public LAN is reserved for the connection to the database.

IP address	192.168.130.x
Subnet mask	255.255.252.0
Gateway	192168128254
DNS	192.168.130.50

The following ports must be opened on firewalls:

- TCP SQL SERVER: 30001 - TCP port for communication with the SQL Server instance,
- UDP (optional): 1434 - SQL Server Browser listening port (for *Server\Instance* connections).

For more information, refer to Configuring the server and the instance.

2.2 Active Directory accounts

- Installation account:
  The account used for the installation of SQL Server instances must have the following permissions:
  - CREATE OBJECT on Active Directory.
  - FULL CONTROL on the target OU.
  - LOCAL ADMIN of SQL Server servers.
- SQL Server service account:
  This service account is used for running SQL Server services. It has LOCAL ADMIN permissions on SQL Server servers. The password must not expire.

2.3 Servers or virtual machines

Power management on servers must be set to **High performance** mode. If the server is a HyperV or VMWare virtual machine, this step must be performed on the host (physical machine) side.

In Windows, change the **High performance** mode in the **Control panel > System and security > Power options**.

2.4 CPU resources and RAM

You must define the RAM quota that matches the amount of memory to allocate to the SQL Server, so that it does not use up all the memory on the server. This value can be configured in *SQL Server Management Studio* after the databases have been installed.

Refer to the recommendations regarding the required CPU resources and RAM in the Adapting the size of the SES Evolution server according to the number of agents section of the *SES installation guide*.

2.5 Storage

The data stored on the SQL Server server is distributed as follows:

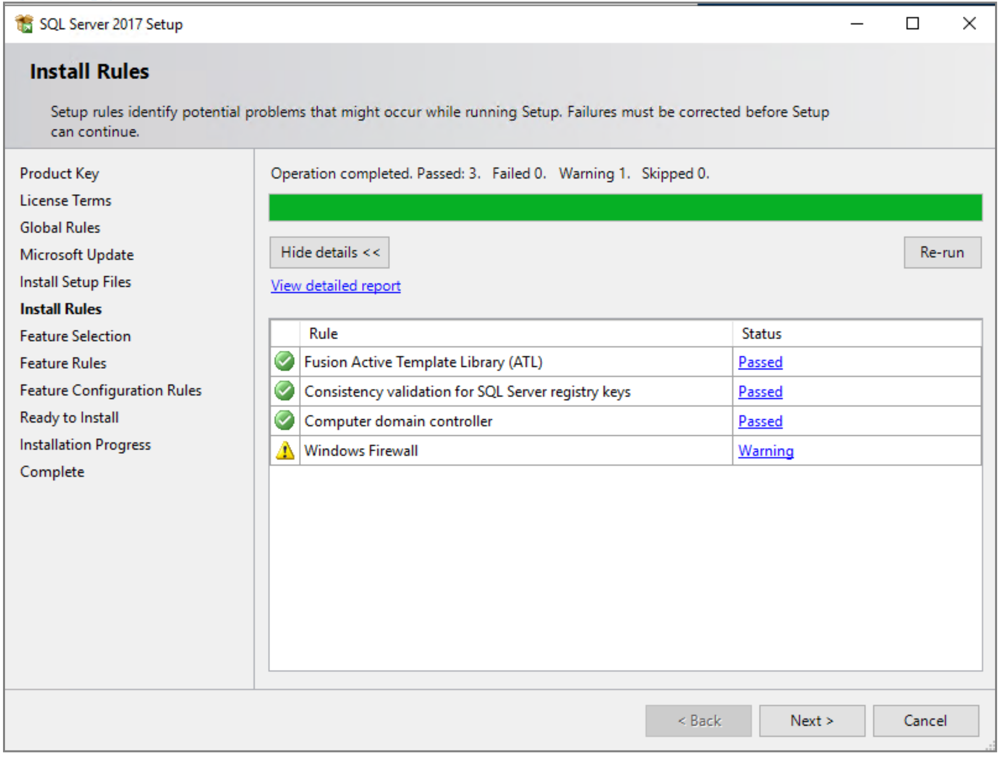
Disk	Contents	Assigned volume
C: drive	Operating system	130 GB (fixed)
E: drive	SQL Server data	Depends on the number of agents (e.g., 150,000 agents = 500 GB)
F: drive	SQL Server logs	50% of the E: drive
G: drive	SQL Server backups	Same volume as the E: drive
H: drive	SQL Server TempDB data	20% of the E: drive

The volumes dedicated to SQL Server (E:,F:,G: and H:) must be excluded from antivirus analyses.

3. Installing SQL Server

The SQL Server server must be a member of the Active Directory domain.

1. Run the SQL Server Installation Center.
2. Select **New SQL Server standalone installation**.
3. Enter the product key, then accept the license terms.
4. If necessary, automatically download the latest Windows and SQL Server updates.
5. After checking the **Install rules**, you will see a warning on the Windows firewall. You must configure it later to allow all SQL Server network traffic. For more information, see section Opening ports on the firewall.



Install Rules

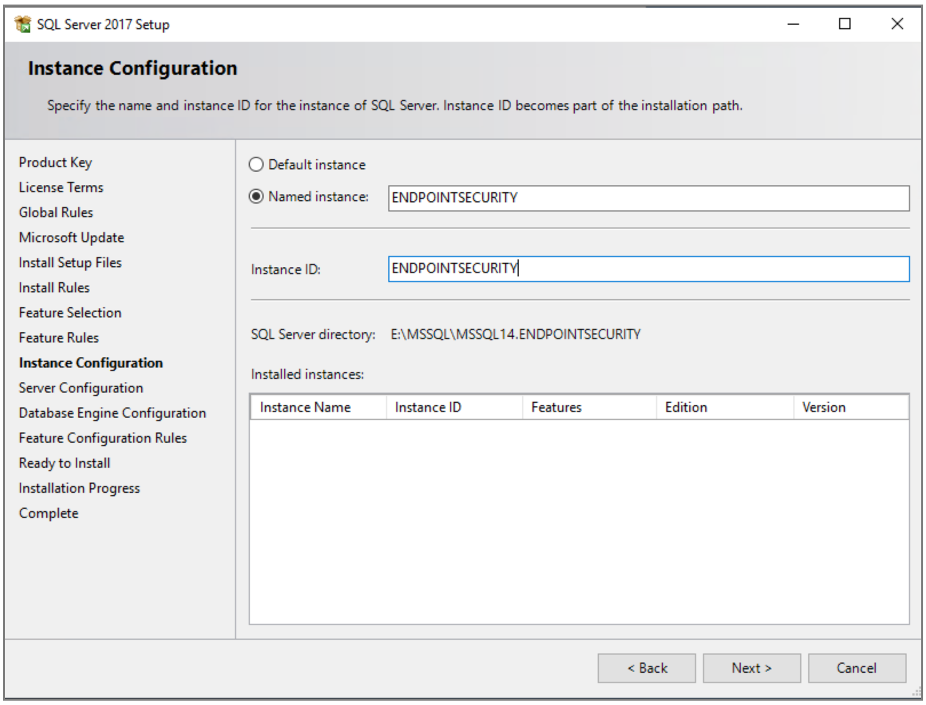
Setup rules identify potential problems that might occur while running Setup. Failures must be corrected before Setup can continue.

Operation completed. Passed: 3. Failed 0. Warning 1. Skipped 0.

	Rule	Status
✓	Fusion Active Template Library (ATL)	Passed
✓	Consistency validation for SQL Server registry keys	Passed
✓	Computer domain controller	Passed
⚠	Windows Firewall	Warning

6. On the **Feature selection** screen, select **Database engine services**, and in the **Instance root directory** field, enter E:\MSSQL.

7. On the **Instance configuration** screen, enter the following parameters:
   Named instance: ENDPOINTSECURITY
   Instance ID: ENDPOINTSECURITY



SQL Server 2017 Setup

Instance Configuration

Specify the name and instance ID for the instance of SQL Server. Instance ID becomes part of the installation path.

Product Key
License Terms
Global Rules
Microsoft Update
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Ready to Install
Installation Progress
Complete

○ Default instance
● Named instance: ENDPOINTSECURITY

Instance ID: ENDPOINTSECURITY

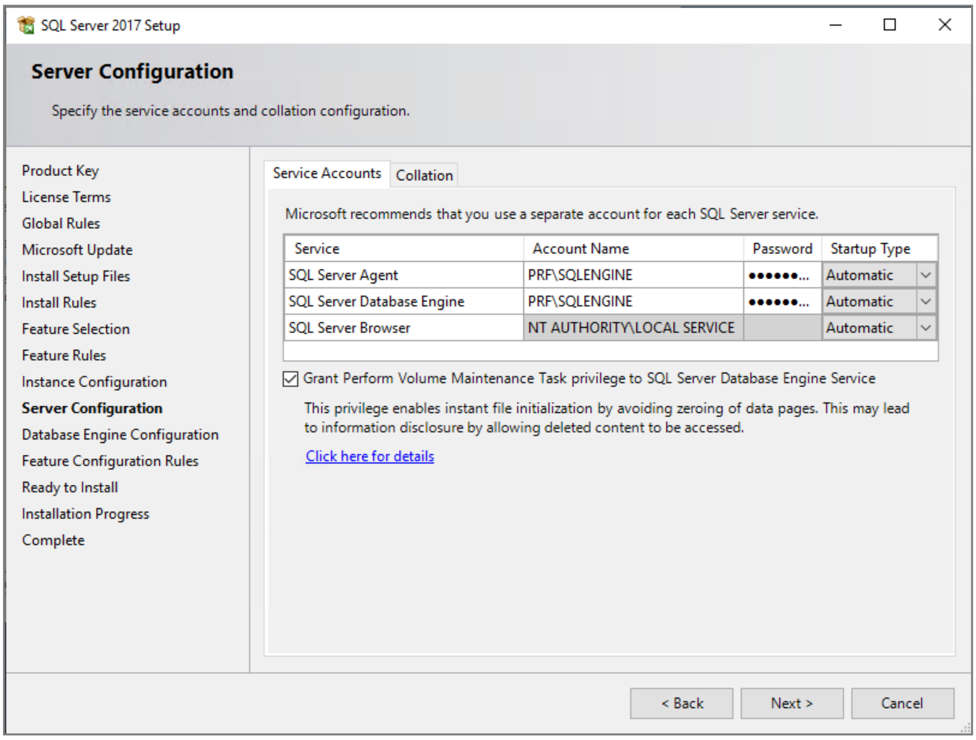
SQL Server directory: E:\MSSQL\MSSQL14.ENDPOINTSECURITY

Installed instances:

Instance Name	Instance ID	Features	Edition	Version

8. On the **Server configuration** screen, under the **Service accounts** tab, fill in the name of the account and the password for the **SQL Server Account** and **SQL Server Database Engine** services. The same account has been used for both services in this example, but you can dissociate them.

9. The **Grant Perform Volume Maintenance Task privilege to SQL Server Database Engine Service** option must be selected. For more information, refer to the related Microsoft documentation.



SQL Server 2017 Setup

Server Configuration

Specify the service accounts and collation configuration.

Product Key
License Terms
Global Rules
Microsoft Update
Install Setup Files
Install Rules
Feature Selection
Feature Rules
Instance Configuration
Server Configuration
Database Engine Configuration
Feature Configuration Rules
Ready to Install
Installation Progress
Complete

Microsoft recommends that you use a separate account for each SQL Server service.

Service	Account Name	Password	Startup Type
SQL Server Agent	PRF\SQLENGINE	●●●●●●...	Automatic
SQL Server Database Engine	PRF\SQLENGINE	●●●●●●...	Automatic
SQL Server Browser	NT AUTHORITY\LOCAL SERVICE		Automatic

☑ Grant Perform Volume Maintenance Task privilege to SQL Server Database Engine Service

This privilege enables instant file initialization by avoiding zeroing of data pages. This may lead to information disclosure by allowing deleted content to be accessed.

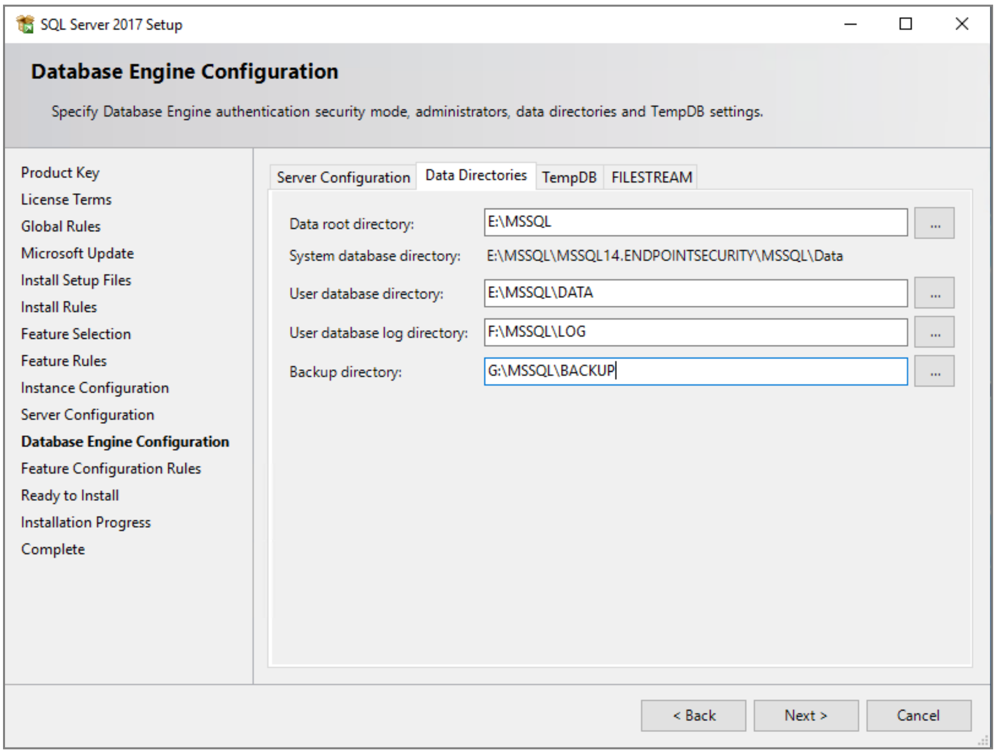
Click here for details

10. In the **Collation** tab, select *French_CI_AS*. For more information, refer to the related Microsoft documentation.
11. On the **Database engine configuration** screen, under the **Server configuration** tab, select **Mixed mode** and set a password for the *sa* account.
    The account needed for the installation will automatically be added to the instance.
12. In the **Data directories** tab, spread out the database files as follows:
    - **Data root directory**: E:\MSSQL

      Instance-specific binaries and libraries.
    - **User database directory**: E:\MSSQL\DATA

      Data files (.mdf or ndf) for user databases.
    - **User database log directory**: F:\MSSQL\LOG

      Log files (.ldf) for user databases.
    - **Backup directory**: G:\MSSQL\BACKUP
      Backup files



SQL Server 2017 Setup

Database Engine Configuration

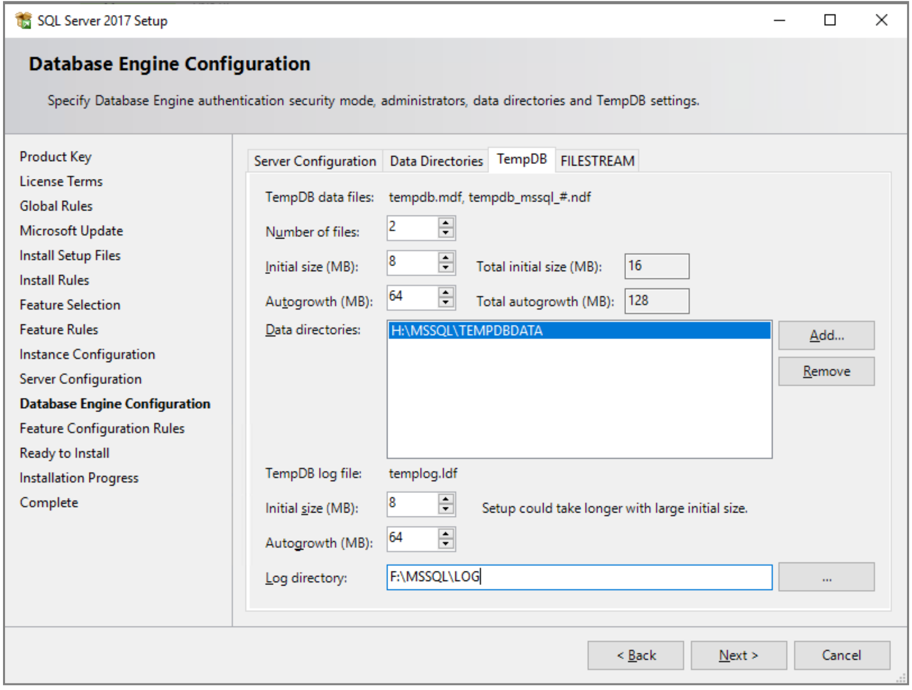
Specify Database Engine authentication security mode, administrators, data directories and TempDB settings.

Data root directory: E:\MSSQL
System database directory: E:\MSSQL\MSSQL14.ENDPOINTSECURITY\MSSQL\Data
User database directory: E:\MSSQL\DATA
User database log directory: F:\MSSQL\LOG
Backup directory: G:\MSSQL\BACKUP

With regard to storage, follow the recommendations given below:

- Do not install SQL Server on the C:\ drive with the operating system.
- Do not store data files and log files on the same disk.
- Isolate the backups of other files.

13. In the **TempDB** tab, the tempDB database is configured by default with one data file per virtual processor. Do not exceed 8 files.
    - **Data directory**: H:\MSSQL\TEMPDBDATA
    - **Log directory**: The same as the one for user databases, F:\MSSQL\LOG



SQL Server 2017 Setup

Database Engine Configuration

Specify Database Engine authentication security mode, administrators, data directories and TempDB settings.

TempDB data files: tempdb.mdf, tempdb_mssql_#.ndf

Number of files: 2

Initial size (MB): 8 Total initial size (MB): 16

Autogrowth (MB): 64 Total autogrowth (MB): 128

Data directories: H:\MSSQL\TEMPDBDATA

TempDB log file: templog.ldf

Initial size (MB): 8 Setup could take longer with large initial size.

Autogrowth (MB): 64

Log directory: F:\MSSQL\LOG

Follow the recommendations given below for tempDB:

- For optimal performance and administration, isolate tempDB data files on a dedicated volume.
- Do not store data files and log files on the same volume.

14. In the **Ready to install** screen, click on **Install**. The SQL Server instance will start installing.

4. Installing SQL Server Management Studio

SQL Server Management Studio (SSMS) is the official utility with which SQL Server instances and databases can be managed. We recommend installing it on a client workstation and managing instances remotely to limit the impact on the server's performance.

SSMS can be installed on the server that hosts the instance, but only for one-off troubleshooting purposes.

1. Download the latest version of the installation program.
2. Run the installation program.
3. Once the installation is complete, restart the workstation.
4. Open SSMS and check whether you are able to connect to the instance locally.

5. Configuring the server and the instance

Make changes to the configuration with an installation account that holds the following privileges:

- SysAdmin on the SQL Server instance,
- Local Admin on the Windows server.

5.1 Enabling automatic compression of backups

- In SQL Server Management Studio, run the following TSQL script on the instance:

```
exec sp_configure 'backup compression default',1
reconfigure
```

5.2 Enabling the remote administrator connection

- In SQL Server Management Studio, run the following TSQL script on the instance:

```
exec sp_configure 'show advanced options',1
reconfigure
exec sp_configure 'remote admin connections',1
reconfigure
```

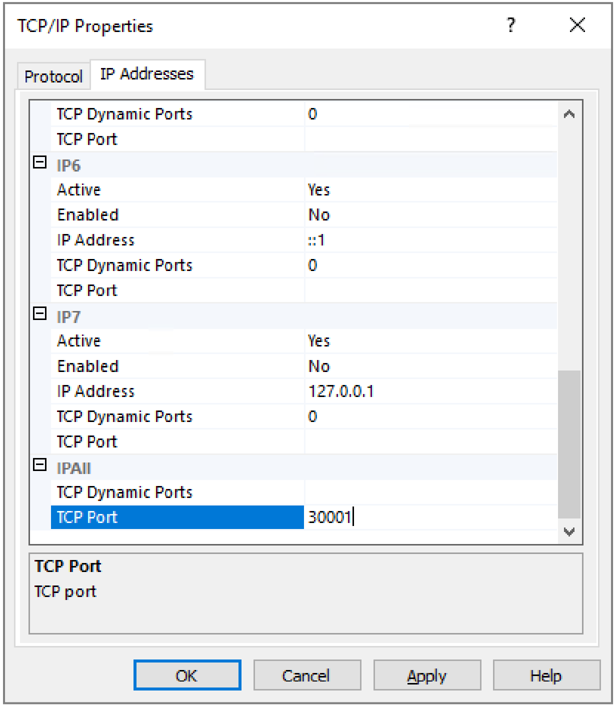
5.3 Allowing the SQL Server service to lock pages in memory

1. Open the Windows local security policy manager.
2. Go to **Local policies > User Rights Assignment**.
3. In the **Lock pages in memory** setting, add the SQL Server service account, *PRF\SQLENGINE* in our example.

5.4 Changing the listening port

The SQL Server listening port must be changed for security reasons.

1. Open the SQL Server Configuration Manager utility.
2. Go to **SQL Server Network Configuration** > **Protocols for ENDPOINTSECURITY**.
3. Right-click on **TCP/IP** and select **Properties**.
4. In the **IP Addresses** tab, under **IPAll**, change the TCP port. Enter port 30001.



TCP Dynamic Ports	0
TCP Port	
⊟ IP6	
Active	Yes
Enabled	No
IP Address	::1
TCP Dynamic Ports	0
TCP Port	
⊟ IP7	
Active	Yes
Enabled	No
IP Address	127.0.0.1
TCP Dynamic Ports	0
TCP Port	
⊟ IPAll	
TCP Dynamic Ports	
TCP Port	30001

TCP Port
TCP port

5. Select **SQL Server services**.
6. In the panel on the right, right-click on SQL Server (ENDPOINTSECURITY) and select **Restart**.

5.5 Opening ports on the firewall

On new Windows servers, the firewall is enabled and TCP ports are closed by default. All the traffic streams that SQL Server requires must be opened:

- SQL TCP: TCP 30001 (SQL Engine)
- SQL UDP: UDP 1434 (SQL Browser)

1. Open the Windows Defender firewall application with advanced security features.
2. In **Incoming traffic rules**, create a **Port** rule with the following parameters:
   - **Protocol** TCP and **Port** 30001,
   - **Action**: Allow connection,
   - **Profile**: Domain, Private and Public
   - **Name**: SQL TCP.
3. Create a second **Port** rule for UDP 1434 with the same parameters, that you will name "SQL UDP".

> **TIP**
> You can also create rules using Powershell:
> ```
> New-NetFirewallRule -Name "SQL TCP" -DisplayName "SQL TCP" -Profile Any
> -Enabled True -Protocol TCP -LocalPort 30001 -Action Allow
> ```

```
New-NetFirewallRule -Name "SQL UDP" -DisplayName "SQL UDP" -Profile Any
-Enabled True -Protocol UDP -LocalPort 1434 -Action Allow
```

5.6 Testing the remote connection

- In SQL Server Management Studio, test connections with a Windows authentication, then a SQL Server authentication.

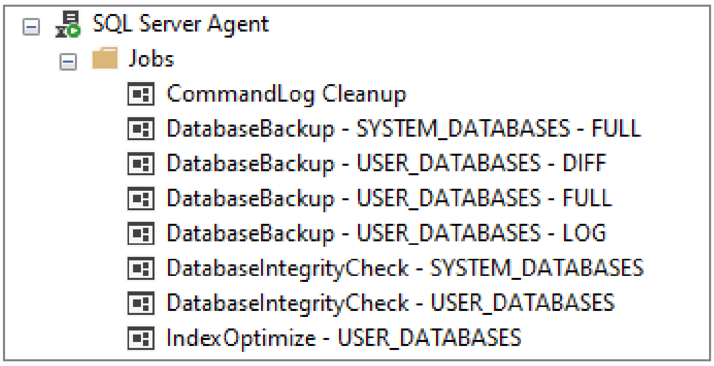
6. Optimizing the maintenance of databases

Perform the following operations to ensure that your SQL Server databases are in optimal working condition:

- Regular database backups,
- Integrity checks.

Maintenance operations on SES Evolution are best performed with the help of scripts and SQL Server stored procedures. You can use the free script that **Ola Hallengren** provides, with which a full maintenance solution can be set up. The *MaintenanceSolution.sql* script installs the following components:

- Several stored procedures:
  - *DatabaseBackup*: SQL Server backup
  - *DatabaseIntegrityCheck*: SQL Server integrity check
  - *IndexOptimize*: Maintenance of SQL Server statistics and indexes. This item will not be used because this step is managed by SES Evolution. For more information, refer to the Administration guide.
- A *CommandLog* table containing logs of operations performed.
- Jobs that allow task execution to be scheduled.



SQL Server Agent
Jobs
CommandLog Cleanup
DatabaseBackup - SYSTEM_DATABASES - FULL
DatabaseBackup - USER_DATABASES - DIFF
DatabaseBackup - USER_DATABASES - FULL
DatabaseBackup - USER_DATABASES - LOG
DatabaseIntegrityCheck - SYSTEM_DATABASES
DatabaseIntegrityCheck - USER_DATABASES
IndexOptimize - USER_DATABASES

> **NOTE**
> With SQLServer Express, use the Windows Task Scheduler or an external scheduler as SQLAgent is not available.

6.1 Backing up databases

Creating backups is the most important task in database administration. Backups allow you to retrieve your data when a server is down, or when configurations, data files, etc. are lost.

Use the *DatabaseBackup* script to make the following backups:

- A full backup of SYSTEM_DATABASES databases to be scheduled **once a day**.

```
EXECUTE [dbo].[DatabaseBackup]
@Databases = 'SYSTEM_DATABASES',
@Directory = NULL,
@BackupType = 'FULL',
@Verify = 'Y',
@CleanupTime = NULL,
@CheckSum = 'Y',
@LogToTable = 'Y'
```

- A full backup of USER_DATABASES databases, to be scheduled **once a day**.

```
EXECUTE [dbo].[DatabaseBackup]
@Databases = 'USER_DATABASES',
@Directory = NULL,
@BackupType = 'FULL',
@Verify = 'Y',
@CleanupTime = NULL,
@CheckSum = 'Y',
@LogToTable = 'Y'
```

- A backup of USER_DATABASES database logs, to be scheduled **frequently several times per day**: every 15 minutes, 30 minutes or hour.

```
EXECUTE [dbo].[DatabaseBackup]
@Databases = 'USER_DATABASES',
@Directory = NULL,
@BackupType = 'LOG',
@Verify = 'Y',
@CleanupTime = NULL,
@CheckSum = 'Y',
@LogToTable = 'Y'
```

6.2 Checking the integrity of databases

Integrity checks must be conducted regularly so that any form of corruption on the database can be detected.

Use the *DatabaseIntegrityCheck* script to launch the following integrity checks:

- Integrity check on SYSTEM_DATABASES to be scheduled once a week at a different time from other jobs.

```
EXECUTE [dbo].[DatabaseIntegrityCheck]
@Databases = 'SYSTEM_DATABASES',
@LogToTable = 'Y'
```

- Integrity check on USER_DATABASES to be scheduled once a week at a different time from other tasks.

```
EXECUTE [dbo].[DatabaseIntegrityCheck]
@Databases = 'USER_DATABASES',
@LogToTable = 'Y'
```

6.3 Cleaning up the CommandLog table

The *CommandLog* table containing logs of operations performed must be cleaned up daily.

Use the following command to delete logs older than 30 days:

```
DELETE FROM master.dbo.CommandLog
WHERE StartTime < DATEADD(day, -30, GETDATE())
```

6.4 Reducing database size

SES Evolution deletes logs by default when they are 12 months old, or 2 months for SQL Server Express. This setting can be configured in the **System** panel, as shown in the section Managing the deletion of logs in the SES Evolution Administration guide. However, SQL Server will not free up nany allocated disk space and keeps it to reuse it later.

If you think that your SQL Server database is taking up too much space on the disk, you can manually reduce it. This operation is not absolutely essential to the proper operation of the database.

There are two possible levels of reduction:

- Level 1 is quick and has no adverse impact on how SES Evolution runs, but the database is not reduced to its full extent.
- Level 2 takes much longer as it depends on the size of the database, and may even make SES Evolution temporarily unavailable.

6.4.1 Level 1

- Run the following script:

```
DBCC SHRINKDATABASE (EsAdministration, 10, TRUNCATEONLY);
GO
DBCC SHRINKDATABASE (EsLogs, 10, TRUNCATEONLY);
GO
```

6.4.2 Level 2

This process may make SES Evolution temporarily unavailable and affect its future performance, and is therefore not recommended. If you want to run it anyway, do so outside busy periods.

1. Shut down all agent handlers.
2. Run the following script:

```
USE EsLogs;
GO
DBCC SHRINKFILE (N'EsLogs_Events');
GO
DBCC SHRINKFILE (N'EsLogs');
GO
CHECKPOINT;
GO
DBCC SHRINKDATABASE (EsLogs, 5, TRUNCATEONLY);
GO
```

3. Restart the agent handlers.

7. Further reading

Additional information and answers to questions you may have about SES Evolution are available on the **Documentation** website and in the **Stormshield knowledge base** (authentication required).

STORMSHIELD

documentation@stormshield.eu

All images in this document are for representational purposes only, actual products may differ.