GUIDE

STORMSHIELD ENDPOINT SECURITY EVOLUTION

SQL SERVER RECOMMENDATIONS

Version 2.7.1

Document last updated: June 30, 2025

Reference: ses-en-sql_server_recommendations-v2.7.1

Table of contents

1. Getting started ... 3
2. Requirements ... 3
2.1 Network ... 3
2.2 Active Directory accounts ... 4
2.3 Servers or virtual machines ... 4
2.4 CPU resources and RAM ... 4
2.5 Storage ... 4
3. Installing SQL Server ... 6
4. Installing SQL Server Management Studio ... 10
5. Configuring the server and the instance ... 10
5.1 Enabling automatic compression of backups ... 10
5.2 Enabling the remote administrator connection ... 10
5.3 Allowing the SQL Server service to lock pages in memory ... 10
5.4 Changing the listening port ... 10
5.5 Opening ports on the firewall ... 11
5.6 Testing the remote connection ... 12
6. Optimizing the maintenance of databases ... 13
6.1 Implementing the script supplied by Stormshield ... 13
6.2 Backing up databases ... 13
6.3 Checking the integrity of databases ... 15
6.4 Scheduling maintenance via SQL Agent ... 16
6.4.1 Requirements ... 16
6.4.2 Creating maintenance jobs ... 16
6.4.3 Customizing the maintenance job schedule ... 17
6.5 Restoring a SES Evolution database ... 17
6.5.1 Requirements ... 17
6.5.2 Restoring a database on the same environment ... 18
6.5.3 Restoring a database onto another server or instance ... 18
6.6 Moving the SES Evolution databases ... 20
6.6.1 Requirements ... 20
6.6.2 Moving the database from to another server or instance ... 20
6.7 Recreating the log database ... 21
6.7.1 Implementing a temporary target instance for creating databases ... 21
6.7.2 Creating a new log database ... 21
6.8 Reducing database size ... 22
6.8.1 Level 1 ... 22
6.8.2 Level 2 ... 22
7. Further reading ... 24

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

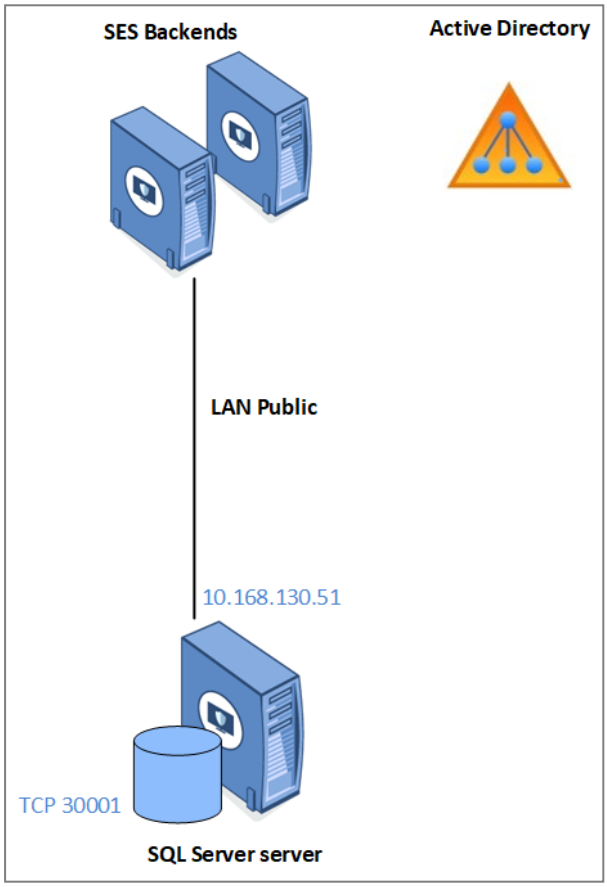
1. Getting started

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	192,168,128,254
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:

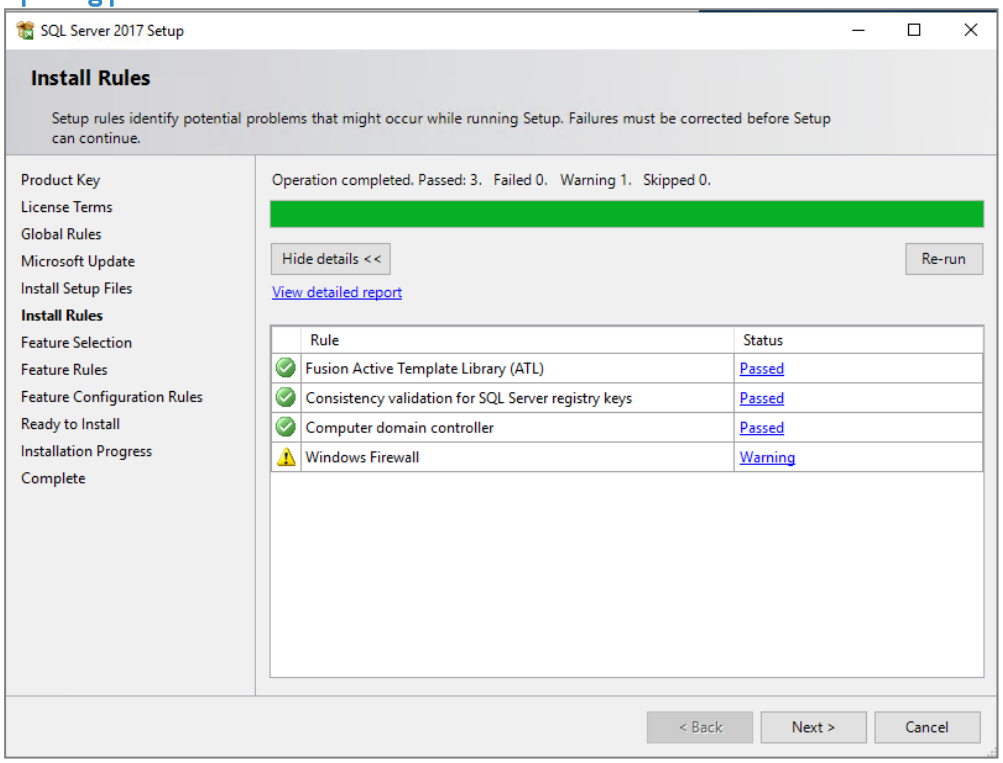
Disk	Content	Assigned volume
C: drive	Operating system	130 GB (fixed)
E: drive	SQL Server data	Depends on the number of agents (e.g., 150,000 agents = 500 GB)
F: drive	SQL Server logs	50% of the E: drive
G: drive	SQL Server backups	Same volume as the E: drive
H: drive	SQL Server TempDB data	20% of the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

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

3. Installing SQL Server

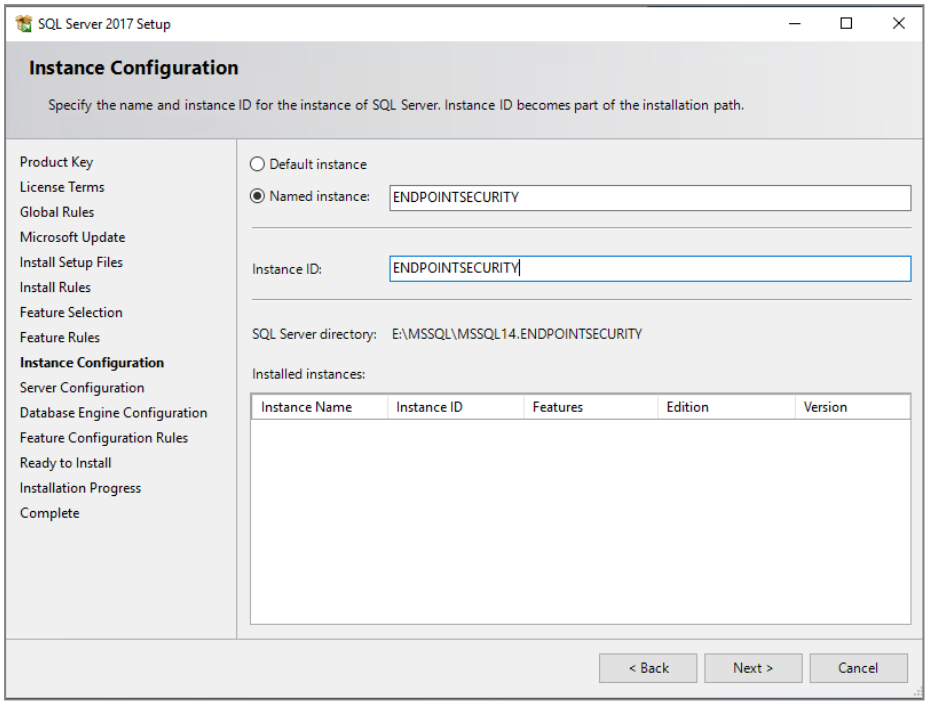
The SQL Server server must be a member of the Active Directory domain.

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.

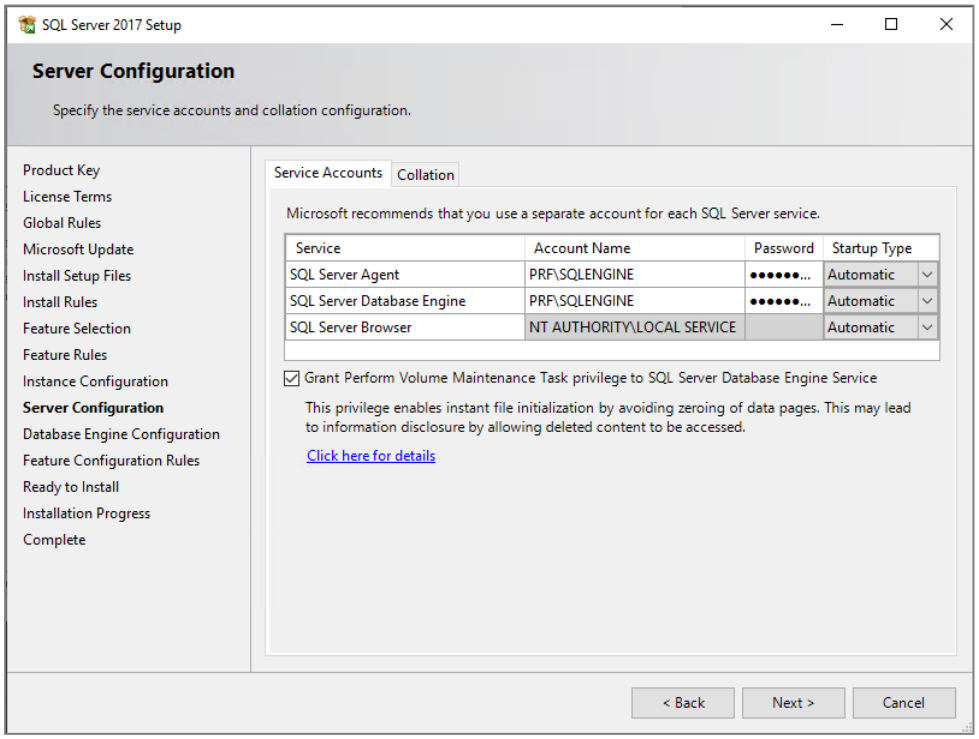


6. On the **Feature selection** screen, select **Database engine services**, and in the **Instance root directory** field, enter E:\MSSQL.

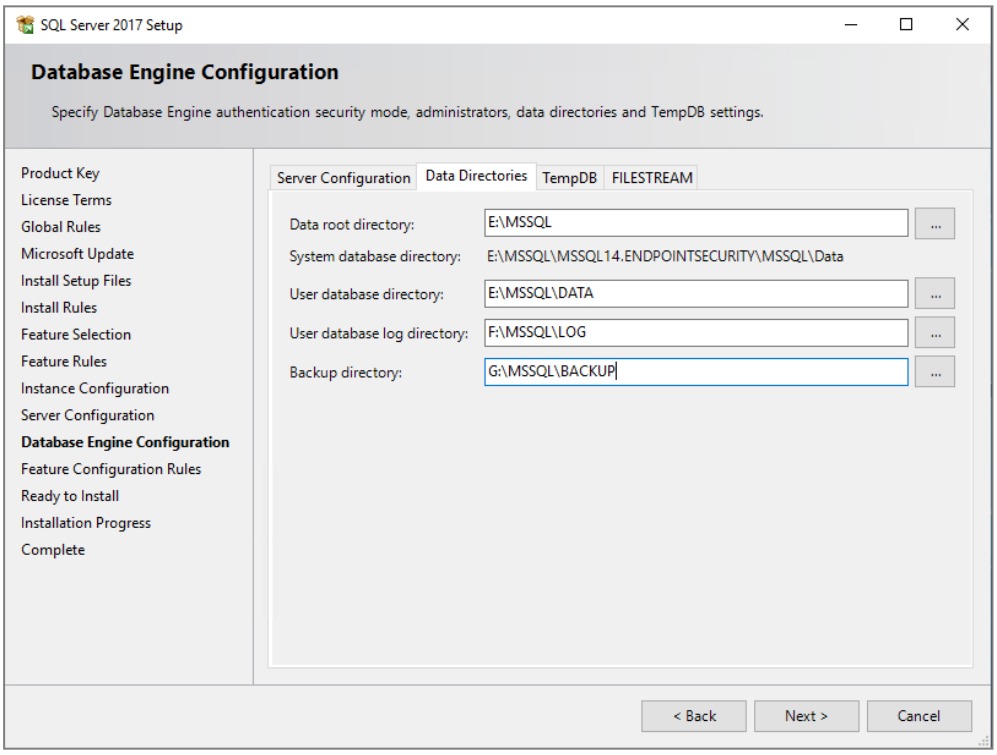
7. On the **Instance configuration** screen, enter the following parameters:
Named instance: ENDPOINTSECURITY
Instance ID: ENDPOINTSECURITY



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.



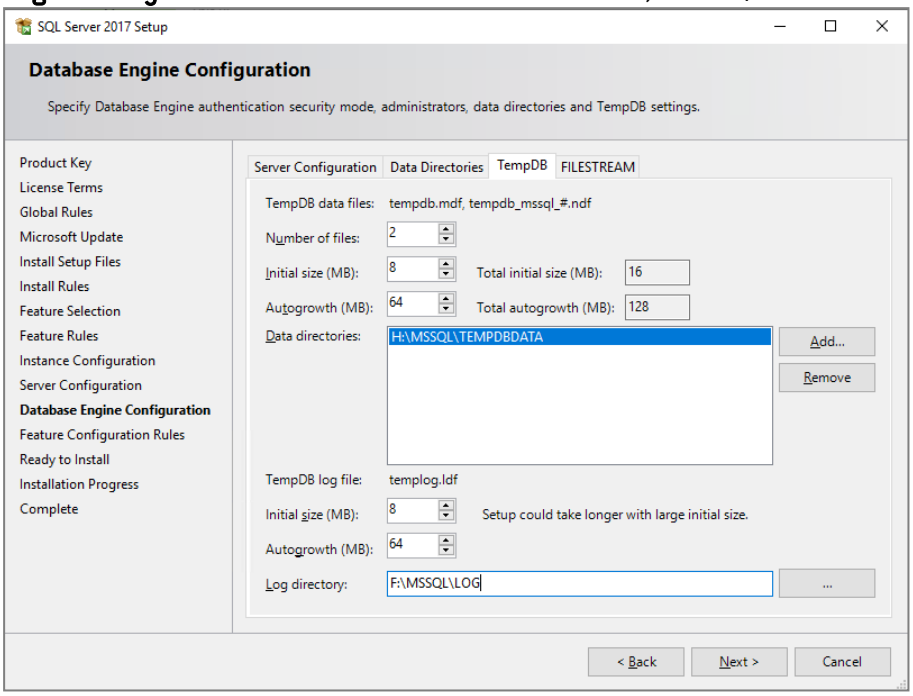
10. In the **Collation** tab, select *French_CI_AS*. For more information, refer to the related Microsoft documentation.
11. On the **Database engine configuration** screen, under the **Server configuration** tab, select **Mixed mode** and set a password for the *sa* account.
    The account needed for 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



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



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
```

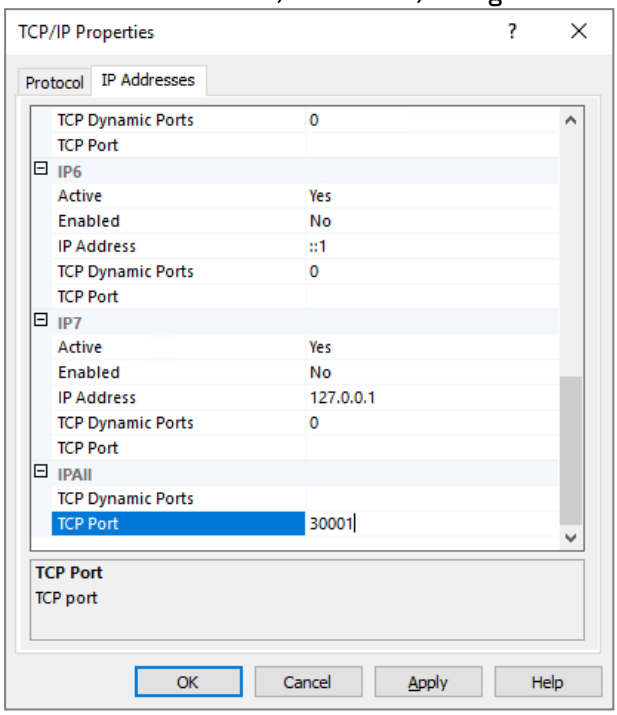
5.3 Allowing the SQL Server service to lock pages in memory

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.



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:

- Back up the databases regularly,
- Check the integrity of the databases.

Stormshield provides the *Stormshield_Database_Maintenance_Procedures.sql*SQL script, which installs SQL Server stored procedures to facilitate the implementation of a maintenance solution for your databases.

These procedures can be used with all versions of SES Evolution later than 2.5.0.

6.1 Implementing the script supplied by Stormshield

1. In your Mystormshield client area, select the **Downloads > Stormshield Endpoint Security > Evolution > Tools** menu and click the *Stormshield_Database_Maintenance_Procedures.sql* link to download it.
2. In SQL Server Management Studio, run the script on each SQL Server instance hosting at lease one SES Evolution database. This script creates the procedures stored in the master database.

6.2 Backing up databases

Backing up 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.

We recommend that you schedule automatic backups of your databases so that they are carried out regularly. For more information on scheduling, refer to Scheduling maintenance via SQL Agent.

To mitigate the risks, you can also back up the databases before important operations, such as:

- Before updating SES Evolution: If you are unable to back up the entire physical or virtual machine, you can make a backup of both databases so that you can reinstall the product and restore the databases in the event of a serious incident during the update.
- After updating SES Evolution: Performing a backup immediately after the product update enables restoration in the event of an incident occurring between the end of the update and the next scheduled backup. This avoids having to reinstall the previous version of SES Evolution, restoring the previous backup, and finally repeating the update.

To back up the databases:

1. Create the destination directories containing the backups of the log and administration databases, for example *E:\Backups\EsAdministration* et *E:\Backups\EsLogs*.
2. Ensure that *SQL Server* has write privileges for these directories. The SQL Server execution account is in the form *MSSQL$ENDPOINTSECURITY* if your instance is named ENDPOINTSECURITY.
3. In SQL Server Management Studio, call the stored procedure *Stormshield_BackupDatabase* and provide the following parameters, specific to your environment:

Parameter	Description
DatabaseName	Name of the database to be backed up. The value can be `EsAdministration` or `EsLogs`.
BackupDirectory	Absolute path of the directory in which the backup file is created. This directory must exist. Network paths are accepted, for example: *\\storage\backups\EsAdministration*. The file created is in the format *DATABASENAME_YYYY-MM-DD_HH-MM-SS_TYPE.bak*. For example: *EsAdministration_2024-07-14_22-30-42_full.bak* or *EsLogs_2024-07-14_22-30-42_diff.bak*.
BackupType	Type of backup to create. The value can be: • `'full'` for a full backup of the database to the backup file, • `'diff'` for a differential backup. The backup file contains only those data that have changed since the last full backup. The file size depends on the use of the database. Usually, a differential backup file is much smaller than a full backup file. Differential backups can be produced more frequently as they are faster and generate a smaller file. However, restoring is more complex.
Compress	Enables or disables compression during the backup. Compression produces a smaller file with a slightly higher CPU consumption. The value can be: • 1: Compression enabled (default and recommended value), • 0: Compression disabled.
CheckSum	Enables or disables the creation of data integrity checksums. These checksums increase the resilience of the backup files in terms of corruption. The value can be: • 1: Creation enabled (default and recommended value), • 0: Creation disabled.
Verify	Enables or disables backup file verification once the backup operation is complete. If verify is enabled, SQL Server checks the backup file (e.g., structure, integrity, checksum if enabled). This verification extends the operation's duration, but allows early detection of errors in the backup file. • 1: Verification enabled (default and recommended value), • 0: Verification disabled.
DryRun	Enables or disables running the procedure in test mode. when the value is 1, the procedure does not actually run the commands and merely displays them. This allows the procedure to be tested before running it in a real-life situation. • 1: Test mode enabled, • 0: Test mode disabled (default value).
CopyOnly	Enables or disables the ability to create a backup that will not be recorded in the log of database backups. This can be useful when exporting the database. • 1: Backup absent from history enabled, • 0: Backup absent from history disabled (default value).

Example commands for a complete point-in-time backup of both databases:

```
EXECUTE master.dbo.Stormshield_BackupDatabase
@DatabaseName = 'EsAdministration',
```

```
@BackupDirectory = 'E:\Backups\EsAdministration',
@BackupType = 'full';
```

```
EXECUTE master.dbo.Stormshield_BackupDatabase
@DatabaseName = 'EsLogs',
@BackupDirectory = 'E:\Backups\EsLogs',
@BackupType = 'full';
```

NOTE
To comply with best practices and avoid potential deletion errors, the procedure does not delete existing backup files. To recover space on your backup medium, implement a policy for deleting obsolete backup files meeting your data preservation needs.

6.3 Checking the integrity of databases

Regularly check the integrity of the databases to detect any corruptions.

- In SQL Server Management Studio, call the *Stormshield_CheckDatabases* stored procedure to check the SES Evolution databases, and optionally the system databases. Supply the following parameters, specific to your environment:

Parameter	Description
FullCheck	Enables or disables full database checking. This operation takes longer but detects more errors than a basic check. For example, for a database of approximately 200 GB, a full check takes about 30 minutes versus 10 minutes for a basic check. The time depends on the capacity of the machine, as well as the CPU and HD load at the time of checking. The value can be: 1: Full check enabled (Default value, recommended for a weekly or monthly interval). 0: Full check disabled (Value recommended for a daily interval).
IncludeSystemDatabases	Includes or excludes checking of the SQL Server system databases. These databases are not linked to SES Evolution but are essential proper operation of SQL Server itself. • 1: System database check enabled (default value and recommended), • 0: System database check disabled. If this check has already been performed by another product or another maintenance plan on the same server, repeating it is not useful.

Example of database integrity check:

In SQL Server Management Studio, run the check procedure only once:

- Full check using the default parameters:

```
EXECUTE master.dbo.Stormshield_CheckDatabases;
```

- Basic and faster check:

```
EXECUTE master.dbo.Stormshield_CheckDatabases @FullCheck = 0;
```

If the command fails, the steps to be taken will be determined by the error or warning messages returned by SQL Server. In most cases, Stormshield recommends restoring a backup of the database rather than having SQL Server repair the data. Indeed, repairing is likely to delete data.

Below is an example of messages returned by SQL Server in the event of corruption of the administration database data file:

```
[2024-07-14T22:30:45.7482429+02:00] Checking master...
[2024-07-14T22:30:46.0763424+02:00] Checking msdb...
[2024-07-14T22:30:46.5159005+02:00] Checking model...
[2024-07-14T22:30:46.6096237+02:00] Checking EsAdministration...
Msg 8939, Level 16, State 98, Line 3
Table error: Object ID 1483152329, index ID 1, partition ID 72057594055557120, alloc unit ID 72057594066436096 (type In-row data), page (1:7501). Test (IS_OFF (BUF_IOERR, pBUF->bstat)) failed. Values are 133129 and -4.
Msg 8928, Level 16, State 1, Line 3
Object ID 1483152329, index ID 1, partition ID 72057594055557120, alloc unit ID 72057594066436096 (type In-row data): Page (1:7501) could not be processed. See other errors for details.
Msg 8978, Level 16, State 1, Line 3
Table error: Object ID 1483152329, index ID 1, partition ID 72057594055557120, alloc unit ID 72057594066436096 (type In-row data). Page (1:7249) is missing a reference from previous page (1:7501). Possible chain linkage problem.
Msg 8976, Level 16, State 1, Line 3
Table error: Object ID 1483152329, index ID 1, partition ID 72057594055557120, alloc unit ID 72057594066436096 (type In-row data). Page (1:7501) was not seen in the scan although its parent (1:7073) and previous (1:7665) refer to it. Check any previous errors.
CHECKDB found 0 allocation errors and 4 consistency errors in table 'IdentifierVersion' (object ID 1483152329).
CHECKDB found 0 allocation errors and 4 consistency errors in database 'EsAdministration'.
repair_allow_data_loss is the minimum repair level for the errors found by DBCC CHECKDB (EsAdministration).
[2024-07-14T22:30:51.8276754+02:00] Checking EsLogs...
```

6.4 Scheduling maintenance via SQL Agent

You can schedule backup and integrity check operations with SQL Agent.

Ensure that backup operations are scheduled after the end of daily SES Evolution maintenance. This latter ca be configured from the SES Evolution administration console. For more information, see Configuring the daily maintenance task in the *Administration guide*.

6.4.1 Requirements

- You must have a Standard or Enterprise SQL Server edition, which includes SQL Agent. With SQL Server Express, use the Windows task scheduler or an external scheduler, as SQL Agent is not available.
- The *SQL Server Agent* Windows service must be running and set to start automatically to ensure that SQL jobs are performed properly.

6.4.2 Creating maintenance jobs

You can create simple maintenance jobs by running the *Stormshield_CreateBasicDailySqlAgentJobs* procedure in SQL Server Management Studio with the following parameters adapted to your environment:

Parameter	Description
EsAdministrationBackupDirectory	Absolute path of the directory in which the backup file of the *EsAdministration* database is created.
EsLogsBackupDirectory	Absolute path of the directory in which the backup file of the *EsLogs* database is created.
CheckDatabaseStartTime	Integrity checking verification time, in HHMMSS format. The default value is 030000, i.e. 3:00 a.m.

Parameter	Description
BackupDatabasesStartTime	Backup start time, in HHMMSS format. The default value is 050000, i.e. 5:00 AM a.m.

If the *EsAdministration* and *EsLogs* databases are hosted on two different SQL Servers, you must perform the operation in two steps: one for each server.

For example, use the following command:

```
EXECUTE master.dbo.Stormshield_CreateBasicDailySqlAgentJobs
@EsAdministrationBackupDirectory = 'E:\Backups\EsAdministration',
@EsLogsBackupDirectory = 'E:\Backups\EsLogs',
@CheckDatabaseStartTime = 020000,
@BackupDatabasesStartTime = 040000
```

Creates two SQL Agent jobs:

- *SES Check databases*: Checks the integrity of the system databases and SES Evolution, on Sundays at 2 a.m. (SQL Server local time),
- *SES Backup (full)*: Creates a full backup of SES Evolution databases (administration and logs), every day at 4 a.m. (SQL Server local time).

If the two jobs above already exist, the procedure overwrites them with the new jobs matching the parameters provided.

6.4.3 Customizing the maintenance job schedule

In SQL Server Management Studio, you can customize the scheduling parameters as well as the parameters of the procedures performed in each job. The image below lists the SQL Agent jobs:

To customize the maintenance job schedule:

1. Right-click the job you want to customize, and select **Properties**.
   The **Job Properties** window opens.
2. In the **Schedules** page, select the job, and then click **Edit** to edit the scheduling parameters as desired.
3. In the **Steps** page, select the job, and then click **Edit** to view and edit the SQL commands run by the script.

6.5 Restoring a SES Evolution database

If a database fails or becomes corrupted, you can restore a backup.

6.5.1 Requirements

- The Windows account used for the backend identity when installing SES Evolution must be a domain account.
- The backup to restore must have been created with the same version as the version of SES Evolution in production. For example, restoring a SES Evolution 2.6.1 database backup on a version 2.6.3 will prevent SES Evolution from starting. Hence the point in backing up databases after each SES Evolution update. For more information, see Performing ad hoc database backups.

- The administration console, the agent handlers, and the SES Evolution backend must be stopped in this sequence before restoring.

6.5.2 Restoring a database on the same environment

- In SQL Server Management Studio, use the *Stormshield_RestoreDatabase* procedure with the following parameters suited to your environment:

Parameter	Description
DatabaseName	Name of the database to which the file will be restored. The value can be *EsAdministration* or *EsLogs*. **CAUTION** All data in this database will be overwritten by the data from the file.
BackupFilePath	Absolute path of the backup file to restore. This file must exist for the procedure to run properly. Network paths are accepted, e.g. *\\storage\backups\EsAdministration_2024-07-14_22-30-42_full.bak*.
DestinationDataDirectory (optional)	Absolute path of the directory to which the SQL Server data files will be restored. When this parameter is used, this directory must exist for the procedure to run properly.
DestinationDataDirectory (optional)	Absolute path of the directory to which the SQL Server transaction log files are restored. If this parameter is used, this directory must exist for the procedure to run properly.

For example, run the following command:

```
EXECUTE master.dbo.Stormshield_RestoreDatabase
@DatabaseName = 'EsAdministration',
@BackupFilePath = 'E:\Backups\EsAdministration\EsAdministration_2024-07-
14_22-30-42_full.bak';
```

6.5.3 Restoring a database onto another server or instance

To restore a SES Evolution database onto another SQL Server or instance, the destination instance must be in a later or equal version of SQL Server.

Follow this procedure for each database, starting with the administration database:

1. On the target SQL Server instance, run the *Stormshield_Database_Maintenance_Procedures.sql* script.

2. Restore the database on the target SQL Server:
   - If the target instance is running on a different machine, run this SQL command:
     ```
     EXECUTE master.dbo.Stormshield_RestoreDatabase
     @DatabaseName = 'EsXxx',
     @BackupFilePath = 'E:\Backups\EsXxx\EsXxx_2024-07-14_22-30-42_
     full.bak';
     ```
     where `Xxx` should be replaced with `Administration` or `Logs`.
   - To customize the destination directories for the SQL files, specify them manually in the command:
     ```
     EXECUTE master.dbo.Stormshield_RestoreDatabase
     @DatabaseName = 'EsXxx',
     @BackupFilePath = 'C:\backups\EsXxx_2024-07-14_22-30-42_
     full.bak',
     @DestinationDataDirectory = 'F:\Data',
     @DestinationLogDirectory = 'G:\Logs';
     ```
   - If the target machine contains several other instances, specify the destination directory explicitly to ensure the files are restored to the directories matching the appropriate instance. For example, for an instance named *DESTINATION*:
     ```
     EXECUTE master.dbo.Stormshield_RestoreDatabase
     @DatabaseName = 'EsXxx',
     @BackupFilePath = 'C:\backups\EsXxx_2024-07-14_22-30-42_
     full.bak',
     @DestinationDataDirectory = 'C:\Program Files\Microsoft SQL
     Server\MSSQL15.DESTINATION\MSSQL\DATA',
     @DestinationLogDirectory = 'C:\Program Files\Microsoft SQL
     Server\MSSQL15.DESTINATION\MSSQL\DATA';
     ```
3. Run the following command on the restored database to automatically recreate the IDs required for SES Evolution operation:
   ```
   EXECUTE master.dbo.Stormshield_RestoreLoginUserMappings;
   ```
4. If you are restoring the log database, update the reference to its instance in the SES Evolution administration database restored beforehand. For this, run the following command on the instance hosting the administration database:
   ```
   EXECUTE master.dbo.Stormshield_ChangeLogsDatabaseInstance
   @NewInstanceName = 'LOGS_SERVER_ADDRESS\LOGS_INSTANCE_NAME';
   ```
   where `LOGS_SERVER_ADDRESS` and `LOGS_INSTANCE_NAME` are the address and SQL instance of the log database. For default instances (which are not named), the address alone, without backslash, is sufficient: `LOGS_SERVER_ADDRESS`.
5. Update the address of the new SQL Server instance in the SES Evolution configuration files:
   1. Back up the following files on each backend:
      C:\Program Files\Stormshield\SES Evolution\Backend\Api\web.config
      C:\Program Files\Stormshield\SES Evolution\Backend\PublicApi\web.config
   2. In each of these files, modify the lines:
      `<add name="Administration" connectionString="Data Source=`**`ADM_SERVER_ADDRESS\ADM_INSTANCE_NAME;`**`Initial Catalog=EsAdministration;..." ... />`
      `<add name="Logs" connectionString="Data Source=`**`LOGS_SERVER_ADDRESS\LOGS_INSTANCE_NAME;`**`Initial Catalog=EsLogs;..." ... />`
      to ensure the `ADM_SERVER_ADDRESS`, `ADM_INSTANCE_NAME`, `LOGS_SERVER_ADDRESS`, and `LOGS_INSTANCE_NAME` values match the addresses and SQL instances of the administration and log databases.
      For default instances (which are not named), the address alone, without backslash, is sufficient: `ADM_SERVER_ADDRESS` and `LOGS_SERVER_ADDRESS`.

6.6 Moving the SES Evolution databases

You can move the SES Evolution databases to another SQL server or another SQL Server instance. For example, this enables you to migrate the databases from one SQL Server release to another release without risking an update on the production instance.

6.6.1 Requirements

- The Windows account used for the backend identity when installing SES Evolution must be a domain account.
- Before moving a database, stop the administration console, the agent handlers, and the SES Evolution backend in this sequence. During the operation, the administration console is not available, but the agents continue to protect the workstations and store the logs generated locally. These logs are sent to the databases and/or to Syslog once the procedure is complete, when the agents reconnect.

6.6.2 Moving the database from to another server or instance

If you move the two databases, first move *EsAdministration* and then *EsLogs*.

1. Close all administration consoles to ensure no data are being edited.
2. On each machine where an agent handler is installed, stop the "Stormshield Endpoint Security Server" Windows service:
   - Either via Windows services,
   - Or via the `net stop EsrCoreSvc` command.
3. On each machine where a backend is installed, stop the IIS server:
   - Either via the "Internet Information Services (IIS) Manager",
   - Or via the `iisreset/stop` command.
4. For each database to move:
   a. Check that the *Stormshield_Database_Maintenance_Procedures.sql* script has been run on the "destination" instance
   b. Create a full backup of the database to move:
   ```
   EXECUTE master.dbo.Stormshield_BackupDatabase
   @DatabaseName = 'EsXxx',
   @BackupDirectory = 'E:\Backups\EsXxx',
   @BackupType = 'full',
   @CopyOnly = 1;
   ```
   where Xxx is either *Administration* or *Logs*.
   c. Perform the Restoring a database onto another server or instance procedure.
5. On each machine where a backend is installed, restart the IIS server:
   - Either via the "Internet Information Services (IIS) Manager"
   - Or via the `iisreset/start` command.
6. On each machine where an agent handler is installed, restart the "Stormshield Endpoint Security Server" Windows service:
   - Either via Windows services,
   - Or the `net start EsrCoreSvc` command.
7. Delete the source databases once you have confirmed that SES Evolution is operational.

If the procedure fails, return to the initial status:

1. Restore the *web.config* files you have saved in the restore procedure.
2. Restart the backends and agent handlers.

6.7 Recreating the log database

If you have not backed up the SES Evolution log database and if a major incident occurs on the server hosting this database, the SES Evolution backends no longer start because no log database can be reached.

You must hence recreate the *EsLogs* log database. For this, you must have a SES Evolution Installation center *EsInstaller.exe* with the same version as your current installation.

> **NOTE**
> If you do not save the *EsAdministration* database and if this database is lost, the only way to recreate it is to reinstall the product.

6.7.1 Implementing a temporary target instance for creating databases

The log database must be created on an existing SQL Server instance.

The *EsAdministration* and *EsLogs* databases were hosted in the same SQL Server instance:

- If the *EsAdministration* database is still operational:
  1. Use or create a temporary SQL Server instance to create a new *EsLogs* database.
  2. Restore this database to the target instance.
- If the *EsAdministration* database is no longer operational:
  1. Restore the *EsAdministration* database.
  2. Use or create a temporary SQL Server instance to create a new *EsLogs* database.
  3. Restore this database to the target instance.

If the *EsLogs* database was hosted on a separate SQL Server instance

1. Set up a new SQL Server and a new SQL Server instance using the same machine name, IP address and DNS name as the instance hosting the lost *EsLogs* database.

6.7.2 Creating a new log database

New *EsLogs* databases are created with a SES Evolution Installation Centre *EsInstaller.exe* of the same version as your current installation.

This database is created on the "destination" instance, which is either a temporary instance, or directly the final target.

1. Launch the SES Evolution Installation center.
2. Choose to perform a new installation.
3. Uncheck as many components as possible to leave only the mandatory components.
4. Configure the installation:
   a. For the administration and log database, select the "destination" instance as target instance.
   b. Configure all the other parameters as for the original installation. The passwords entered in the **Certificates** section are of no use.

5. Start the installation.
6. Run the *Stormshield_Database_Maintenance_Procedures.sql* script on the "destination" instance.
7. Run the following command on the "destination" instance to remove the *EsAdministration* database that was created during installation:

```
EXECUTE master.dbo.Stormshield_DropDatabase @DatabaseName =
'EsAdministration';
```

> **CAUTION**
> Take care not to remove the real *EsAdministration* database.

8. Run the following command on the "destination" instance to create the accesses for the Windows account used by the backends:
```
EXECUTE master.dbo.Stormshield_CreateBackendAccess
@BackendAccountName = 'DOMAIN_NAME\BACKEND_USER_NAME'
```
where DOMAIN_NAME\BACKEND_USER_NAME corresponds to the Windows domain account entered during installation of the SES backends.
9. If the "destination" instance is a temporary instance, move the newly created *EsLogs* database to the final target instance by running the Moving the SES Evolution databases procedure.

6.8 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 Managing log deletion section of the *Administration guide* SES Evolution. 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. It may result in SES Evolution being unavailable.

6.8.1 Level 1

- Run the following script:

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

6.8.2 Level 2

This procedure may make SES Evolution temporarily unavailable and have an impact on its future performance. It 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).

documentation@stormshield.eu

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