McAfee Database Security 4.6.0

Configuring SAP HANA monitoring

Configuring SAP HANA monitoring

You can use HANA's auditing feature to monitor SAP HANA databases running on Linux servers.

Note Only SAP HANA SPS09 Revision 90 or newer databases are supported

SAP HANA monitoring support was introduced into McAfee® Database Security in version 4.4.9-SP2 (Sensor version: 4.4.9-17647; Server version: 4.4.9-52429).

SAP HANA database auto-detection was introduced into McAfee® Database Security Sensor in version: 4.4.9-17837.

SAP HANA syslog auto-configuration and DAL features were introduced into McAfee® Database Security Sensor in version: 4.6.0-18166.

Overview

You can monitor a SAP HANA database using a locally installed sensor (on the same machine as the SAP HANA database) or a remote sensor (on any other machine reachable using TCP communication). The only difference between those two configurations is that the sensor detects locally installed databases automatically, whereas remotely monitored instances require additional configuration using a database list file.

Note SAP HANA monitoring is supported only by a Linux 64-bit sensor (local and remote monitoring).

The database sends its activity reports to the syslog infrastructure of the operating system, which then sends this information to the sensor over TCP using a configurable port.

A single sensor can monitor several databases, each on a different port (for example, by editing the database list file when monitoring remote databases).

When using auto-configuration, the sensor selects ports in coherency with HANA's indexserver SQL port (Every indexserver process opens three communication ports: Internal communication, SQL and HTTP in the range 30000-39999. More information regarding HANA's port selection policy can be found here and here). By default, for each port 3xxxx that the indexserver uses, the sensor uses the matching 5xxxx port (thus the valid range of 50000-59999).

You can change the sensor's port range as described in *Step 7: Change the sensor port range*.

To fully utilize the sensor's capabilities, we suggest you allow the sensor to connect to the database using a DAL connection as described in *Step 5: Configure the DAL connection*.

Tasks

- *Step 1: Create the database list file*
- *Step 2: Configure the auditing feature*
- *Step 3: Configure the syslog infrastructure*
- *Step 4: Enable SAP HANA monitoring*
- *Step 5: Configure the DAL connection*
- *Step 6: Configure an alternative path for the HDB ODBC library*
- *Step 7: Change the sensor port range*

Step 1: Create the database list file

If you are planning to monitor remotely installed databases, you need to create a database list file.

Note Locally installed databases do not require this configuration.

To boostrap the integration, the sensor uses a database list to specify the monitored databases. The database list is read from a directory of known databases (`/etc/mfe.dbs/dbs-list.d`).

The directory contains a list of files, including the configuration details of a database instance. File names are in free form but must end with the extension ".conf".

The sensor scans the directory and listens on the specified "activity-socket" for databases with the 'ACTIVE' monitor-state.

Note The external data source should constantly try to connect to the "activity-socket". The McAfee Sensor starts listening on the specified socket after the database is "approved" in the McAfee Database Security console.

The database configuration details are encoded in the following JSON format:

```
{
   socket-protocol-version: "1.0", //For future use
   db-type: SAPHANA,
   ip: <string>, //The IP address of the SAP HANA DB
   port: <string>, //The SQL-port of HANA DB's indexserver process
   host: <string>, //The output of the `hostname` bash command
   version: <string>, //DB version, required format: x.y.z
   instance-name: <string>, //HANA <SID><InstanceNo.>, e.g. HDB00
   monitor-state: <string: ACTIVE|STOPPED>, //ACTIVE to monitor DB
   activity-socket: <ip>:<port> //Sensor incoming comm. socket address
}
```

Note If you install the sensor on the same machine as the DB, activity-socket should use localhost (or 127.0.0.1) rather than the actual IP of the machine.

If you install the sensor remotely, the activity-socket must use the actual IP address of the machine.

Sample configuration records (all the examples in this document adhere to examples 1 and 3):

1 Manual configuration of a **single-container** running on the **same** host:

```
{"socket-protocol-version":"1.0","db-
type":"SAPHANA","ip":"localhost","port":"30015","host":"hana-dev-
```

```
srv","version":"1.00.091","instance-name":"HDB00","monitor-state":"ACTIVE",
"activity-socket":"localhost:50015"}
```

2 Manual configuration of a **single-container** running on a **remote** host:

```
{"socket-protocol-version":"1.0","db-
type":"SAPHANA","ip":"10.11.12.13","port":"30015","host":"hana-qa-
srv","version":"1.00.091","instance-name":"HDB00","monitor-state":"ACTIVE",
"activity-socket":"10.11.12.100:50015"}
```

Note

In this example the DB is running on a machine using IP 10.11.12.13 and the sensor on a machine using IP 10.11.12.100

3 Manual configuration of a **single tenant in a multi-container** on the **same** host:

```
{"socket-protocol-version":"1.0","db-
type":"SAPHANA","ip":"localhost","port":"50141","host":"hana-dev-
srv","version":"1.00.091","instance-name":"HDB01.HR","monitor-state":"ACTIVE",
"activity-socket":"localhost:50141"}
```

Step 2: Configure the auditing feature

You need to configure the SAP HANA database to enable the auditing feature and send reports to the operating system's syslog infrastructure.

Note

This guide specifies the steps required when using SAP HANA Studio.
A more comprehensive explanation can be found at SAP HANA's website.

Task

1 In SAP HANA Studio, open the Systems view.

2 From the toolbar, select **Window > Show View > Systems**.

3 Right-click the system you need to monitor, then select **Open SQL Console**.

4 Copy these statements into the console:

```
ALTER SYSTEM ALTER CONFIGURATION ('global.ini','SYSTEM') set ('auditing
configuration','default_audit_trail_type' ) =  'SYSLOGPROTOCOL' with reconfigure;

ALTER SYSTEM ALTER CONFIGURATION ('global.ini','SYSTEM') set ('auditing
configuration','global_auditing_state' ) =  'true' with reconfigure;

CREATE AUDIT POLICY "DBS" AUDITING ALL ACTIONS EXCEPT FOR SYS, _SYS_STATISTICS,
_SYS_REPO, _SYS_EPM, _SYS_AFL LEVEL INFO;

ALTER AUDIT POLICY "DBS" ENABLE
```

5 Press **F8** or click the green button to deploy the configuration.

Step 3: Configure the syslog infrastructure

You now need to configure the syslog infrastructure to forward messages sent by the SAP HANA database to the sensor.

Note

This guide specifies the required steps when using syslog-ng or rsyslog. If you are using a different syslog infrastructure, contact support with your specific request details.

Auto configuration

If you installed the sensor on the same machine as the HANA DB, you can configure the sensor to perform an auto-configuration of the syslog infrastructure by following these auto configuration steps.

1 On the **Sensor properties** page, click **Advanced**.

2 In the text box, add an entry on a separate line:
`enable.syslog.auto.config=1`

3 Restart the sensor.

Note

The auto-configuration feature creates a backup of the original file in the same directory with the extension `.backup`.

Manual configuration

You can manually configure the syslog infrastructure, for example, if the sensor is installed on a different machine.

Syslog-ng manual configuration

Please follow these steps to manually configure you syslog-ng application.

1 TaskOpen the file `/etc/syslog-ng/syslog-ng.conf` for editing.

2 Add these lines to the end of the file:

```
#
# Database Security Auditing
#
filter f_dbs { facility(authpriv); };
```

3 For each monitored instance and/or tenant (If you are using multi-tenant DBs), add these lines:

```
filter f_dbs.<sid>.<instance-no>.<db-name> { filter(f_dbs) and
match('<sid>;<instance-no>') and match('<db-name>'); }

destination dbs.<sid>.<instance-no>.<db-name> { tcp("<ip>" port(<port>)); }

log { source(src); filter(f_dbs.<sid>.<instance-no>.<db-name>);
destination(dbs.<sid>.<instance-no>.<db-name>); };
```

Where:

- <hostname> – The output of the bash command `hostname`.
- <sid> – The system ID, in at toal of three uppercase letters and/or numbers.

- <instance-no> – The database instance number.
- <db-name> – The name of the database in a multi-tenant environment in uppercase letters. If you are not using multi-tenant environments, this should be replaced by an empty string.

Note

If you are using a single-container configuration, omit the last part of the instance filter definition `and match(`<db-name>`)`. This addition is required only for multi-container configurations.

The <ip> and <port> values should be defined as follows:

- If you have used the manual configuration files, these values must match the activity-socket values specified in the configuration file.
- If you are using auto detection, the <ip> should be "localhost" and the port should be calculated according to the formula: **`hana-sql – hana-offset + sensor-offset`**
  - hana-sql = The port number HANA's indexserver process uses for incoming SQL requests (Hint: This is the port you specify when connecting to the HANA using hdbsql)
  - hana-offset = 30,000
  - sensor-offset = 50,000 (If you have altered the sensor configuration per Step 7 of this guide, use the <base_port_value> you set.)

4 The final result should resemble one of these examples:

An example adhering to configuration examples 1 and 3 from Step 1:

```
#
# Database Security Auditing
#
filter f_dbs { facility(authpriv); };

filter f_dbs.HDB.00 { filter(f_dbs) and match('HDB;00'); };
destination dbs.HDB.00 { tcp("localhost" port(50015)); };
log { source(src); filter(f_dbs.HDB.00); destination(dbs.HDB.00); };

filter f_dbs.HDB.01.HR { filter(f_dbs) and match('HDB;01') and match('HR'); };
destination dbs.HDB.01.HR { tcp("localhost" port(50141)); };
log { source(src); filter(f_dbs.HDB.01.HR); destination(dbs.HDB.01.HR); };
```

An even simpler example - Monitoring a machine with a single single-container instance:

```
#
# Database Security Auditing
#
filter f_dbs { facility(authpriv); };
destination dbs { tcp("localhost" port(50015)); };
log { source(src); filter(f_dbs); destination(dbs); };
```

5 Reload the syslog-ng configuration using the command:
`/etc/init.d/syslog reload`

Rsyslog manual configuration

Please follow these steps to manually configure you rsyslog application.

Tasks

1 Open the file `/etc/rsyslog.conf` for editing.

2 Add the title to the end of the file:

```
#
# Database Security Auditing
#
```

3 For each monitored instance and/or tenant (If you are using multi-tenant DBs), add these lines after the title:

```
if      ($syslogfacility-text == 'authpriv' and \
        $msg contains '<sid>;<instance-no>' and $msg contains '<db-name>') \
then    @@<ip>:<port>
```

Where the fields <sid>, <instance-no>, <db-name>, <ip> and <port> should be defined according to the explanation provided at the adjacent task under **Manual syslog-ng configuration**.

Note

If you are using a single-container configuration, omit the last part of the filter condition `and $msg contains `<db-name>'`. This addition is required only for multi-container configurations.

4 The final result should resemble one of these examples:

An example adhering to configuration examples 1 and 3 from Step 1:

```
#
# Database Security Auditing
#
if      ($syslogfacility-text == 'authpriv' and \
        $msg contains 'HDB;00') \
then    @@localhost:50015

if      ($syslogfacility-text == 'authpriv' and \
        $msg contains 'HDB;01' and $msg contains 'HR') \
then    @@localhost:50141
```

An even simpler example - Monitoring a machine with a single single-container instance:

```
#
# Database Security Auditing
#
if      ($syslogfacility-text == 'authpriv')
then    @@localhost:50015
```

5 Restart rsyslog for the configuration to take place using this command:
`/etc/init.d/syslog restart`

Note Rsyslog doesn't update remote server configurations when the reload command is run.

Step 4: Enable SAP HANA monitoring

Before you can monitor SAP HANA databases, you need to enable SAP monitoring in the McAfee Database Security console.

Task

1 On the **Sensor properties** page, click **Advanced**.

2 In the text box, add an entry on a separate line:
`hana.enable=1`

3 Restart the sensor.
The sensor detects the SAP HANA database configured in the database list file (Step 1).

4 Start monitoring for the detected SAP HANA database.

5 Add rules for the database.

Step 5: Configure the DAL connection

A DAL connection allows the sensor to:

- Gather more information on active sessions, e.g. Application name.
- Use the sensor's kill session feature.
- Improve open session state monitoring.

If you want to allow the sensor to connect to the database and fully utilize its power, you must acquire the SAP HANA Client. This package, which includes the HDB ODBC Driver required by the sensor, is usually installed as a part of the database installation.

To configure the DAL connection, you must create a database user and configure the sensor in the McAfee Database Security Console.

Before you begin

- Verify the SAP HANA Client installation and the location of the required library by searching for **libodbcHDB.so**. If it has not been installed, download and install it now. (You can find the package here and an installation guide here.)

Note

The sensor expects to find the package at `/usr/sap/hdbclient/`. If you have installed the package in a different path, see *Step 6: Configure an alternative path for the HDB ODBC library*.

Task

1. In the McAfee Database Security Console, on the **DBMS configuration** page, click **Alternative DBMS Connection** and select the **Enable Alternative DBMS Connection**
2. Click the link adjacent to the user name field, then download the script file (`sap_hana_externalUserScript`) to the machine where the sensor is installed.
3. Copy the script file to the root directory of your SAP HANA database user (<SID>adm)
4. Switch to your your SAP HANA database user (<SID>adm)'.
5. Run the command `` `sh hana_create_user*.sh` `` to execute the script, then follow the on-screen instructions.
6. In the McAfee Database Security console, enter these parameters:
   - **User Name** – The username you have selected for the new user when running the script
   - **Password** – The password you have selected for the new user when running the script
   - **Connection String** – An <ip>:<port> combination describing the SQL API port of the database

Step 6: Configure an alternative path for the HDB ODBC library

Although the sensor assumes a default path to the HDB Client, it is can be configured to access the required file in any location on the machine.

Task

1. On the **Sensor properties** page, click **Advanced**.
2. In the text box, add an entry on a separate line:
   `hana.odbc.driver=<absolute-path-to-libodbcHDB.so>`
3. Restart the sensor.

Step 7: Change the sensor port range

Although the sensor uses a default port range, which is not registered by any application (50000-59999), this port range might be sub-optimal for some users.

You can change the sensor port range to any sequential range of 10000 ports. This ensures flawless implementation of our port allocation algorithm, which works in conjuction with the SAP HANA indexserver process.

You can change the port range from the McAfee Database Security Console:

Task

1. On the **Sensor properties** page, click **Advanced**.

2 In the text box, add an entry on a separate line:
`hana.base.port=<base_port_value>`
The base port for any given range is the first port in that range, e.g., 13500 for 13500-14499.

3 Restart the sensor.