RSA NetWitness Platform

Event Source Log Configuration Guide

Microsoft SQL Server

Last Modified: Tuesday, December 11, 2018

Event Source Product Information:

Vendor: Microsoft
Event Source: SQL Server
Versions: 2000, 2005, 2008, 2012, 2014, 2016, and MS SQL Express
Additional Downloads:

- sqlServerAudit2000.sql
- sqlServerAudit2005.sql
- sqlServerAudit2008.sql
- sqlServerAudit2012.sql
- sqlServerAudit2014.sql
- sqlServerAudit2016.sql
- uninstallSqlServerAudit.sql
- RSA_MSSQLAuditStoredProcedures.dll
- sftpagent.conf.mssql

RSA Product Information:

Supported On: NetWitness Platform 10.0 and later
Event Source Log Parser: mssql
Collection Method: ODBC, File, and Windows event logs
Event Source Class.Subclass: Storage.Database

Microsoft SQL Collection Overview

RSA NetWitness Platform supports several different collection methods for Microsoft SQL Server, depending on the version of SQL Server and Microsoft Windows that you are using. The following table describes the various combinations of Windows version, MS SQL version, and the collection methods used for each.

MS SQL Version	Platform	Collection Methods
2000	Windows 2000, 2003	ODBC, Windows Legacy, File
2005, 2008 Standard	Windows 2003, 2008	File (ERRORLOG), Windows Eventing (MS SQL Service Logs), ODBC (SQL Auditing)
2008 Enterprise and later	Windows 2008 and later	File (ERRORLOG), Windows Eventing (MS SQL Service Logs and SQL Auditing) or ODBC (SQL Auditing)

Note the following:

- If you are running SQL Server 2000, RSA recommends that you configure collection for all three methods: the File Service, the ODBC Service, and the Windows Service.
- If you are using SQL Server 2008, then it must be SQL 2008 Enterprise Edition. SQL Server 2008 Standard Edition does not do SQL Auditing.

ODBC Service

The ODBC Service collects database traces stored in a local trace file, which includes important auditing features like logon, security, configuration, and database changes.

For all supported versions of Microsoft SQL Server, you can collect messages through the ODBC Service. Note the following:

- You must configure collection using SQL Server administrator credentials.
- You collect database traces stored in a local trace file, which includes important auditing features like logon, security, configuration, and database changes.
- The ODBC Service is a very stable method of collecting messages.

Windows and Windows Eventing Services

For all supported versions of Microsoft SQL Server, you can collect System and application messages stored in the Windows System and Application log files. Note the following:

- For SQL Server 2005, 2008, 2012, 2014 and 2016, you can collect audit level messages.
- For SQL Server 2000 or 2005, running on Windows Server 2003, you set up the Windows Legacy Collector.

File Service

The File Service collects system level messages stored in a local error log file.

Configure ODBC Collection

You must complete these tasks to configure Microsoft SQL Server collection:

- Set up SQL Server Auditing on SQL Server (2008 and later) on Windows Server (2008 and later)
- Set up the ODBC Service on the Microsoft SQL Server event source
- Set up the ODBC Service on RSA NetWitness Platform

Set up SQL Server Auditing on SQL Server (SQL Server 2008, 2012, 2014, and 2016)

To capture Microsoft SQL Server Auditing messages, you must configure both SQL Server and RSA NetWitness Platform.

To set up SQL Server Auditing on SQL Server:

1. On the SQL Server platform, open SQL Server Management Studio.
2. Log onto the server using administrator credentials.
3. Navigate to **Security** > **Audits** and create a new audit.
4. Depending on your system, set the **Audit Destination** to **Application Log** or **Security Log**, and set the values of all other fields with appropriate values for your organization.

   > **Note:** If you want to use security logs, you must set up administrative privileges on the SQL Server. To set up the appropriate privileges, follow the instructions from the Microsoft MSDN page.

5. Click **OK** to create the audit.
6. Create **Server Audit Specifications** and **Database Audit Specifications**, and point them to the audit you created.
7. Configure the **Server Audit Specifications** and **Database Audit Specifications** by selecting all of the event types for which you want to collect audit events.

Set up the ODBC Service on the Microsoft SQL Server event source

Warning: The ODBC Service only needs to be set up per SQL Server instance, not for every database on the SQL Server.

Configure the SQL Server and any custom events, as well.

Configure the SQL server

Go to the Microsoft SQL Server Additional Downloads page, and download the appropriate files:

Note: The files must be accessible by the SQL Server host.

- For SQL Server 2000, **sqlServerAudit2000.sql** script.
- For SQL Server 2005, **sqlServerAudit2005.sql** script.
- For SQL Server 2008, **sqlServerAudit2008.sql** script.
- For SQL Server 2012, **sqlServerAudit2012.sql** script.
- For SQL Server 2014, **sqlServerAudit2014.sql** script.
- For SQL Server 2016, **sqlServerAudit2016.sql** script.
- For SQL Server 2005, 2008, 2012 2014 or 2016 **RSA_MSSQLAuditStoredProcedures.dll**.

Create a directory, **C:\MyTraceFiles**, with enough space to hold the SQL Server trace files, and grant delete permissions to this directory for the account running the SQL server process "SQL Server (MSSQLSERVER)."

Note: You will need this directory to set up the ODBC type on the RSA NetWitness Platform.

Configure Custom Events for the SQL Server

1. On the SQL Server, start the **SQL Server Profile** tool.
2. Select **File** > **New** > **Trace**.
3. On the **Events** tab, select the events to trace.

4. On the **Data Columns** tab, select all the columns.

   > **Note:** The ODBC service works only if you select all the columns. You can also select a filter.

5. Click **Run**.
6. Select **File** > **Script Trace** > **For SQL Server 2000 or 7.0**.
7. Save the script file to a temporary location on your computer. You will need to open this file in step 9.
8. Close the **SQL Server Profile** tool.
9. Open the script file that you created, and follow these steps:

   a. Copy the blocks of code marked by `-- Set the events` that are similar to the following example:

   ```
   exec sp_trace_setevent @TraceID, 10, 1, @on
   ```

   Do not include the following:

   ```
   declare @on bit
   set @on = 1
   ```

   b. Copy the blocks of code marked by `-- Set the filters` that are similar to the following example:

   ```
   exec sp_trace_setfilter @TraceID, 10, 0, 7, N'SQL Profiler'
   ```

   Or

   ```
   set @intfilter = 100
   exec sp_trace_setfilter @TraceID, 22, 0, 4, intfilter
   ```

10. Open **sqlServerAudit2000.sql** and make the following changes:

    a. Find the **nic_aud_set_events** procedure.

    b. Paste the code that you copied for **events** in step 9 between the following comments, replacing any existing code within the comments:

    ```
    -- *************
    -- *** Custom events should be pasted below this line!!!
    -- *************

    <YOUR CUSTOM EVENTS HERE>

    -- *************
    -- *** Custom events should be pasted above this line!!!
    -- *************
    ```

    c. Paste the code that you copied for **filters** in step 9 between the following comments, replacing any existing code within the comments:

```
-- *************
-- *** Custom filters should be pasted below this line!!!
-- *************

<YOUR CUSTOM FILTERS HERE>

-- *************
-- *** Custom filters should be pasted above this line!!!
-- *************
```

11. Save the changes to **sqlServerAudit2000.sql**.

To configure custom events for SQL Server 2005 or later:

1. Open the **sqlServerAudit2005.sql** ,**sqlServerAudit2008.sql**, **sqlServerAudit2012.sql**, **sqlServerAudit2014.sql** or **sqlServerAudit2016.sql** file. These files include all possible events.

2. To enable or disable events, follow these steps:

   > **Note:** For the script to function, you must enable or disable all lines for an event.

   a. To enable events, delete **- -** from the beginning of each exec statement for a given event.

   b. To disable events, add **- -** to the beginning of each exec statement for a given event.

Install the Audit Procedures

> **Note:** You must be the database administrator to install the audit procedures.

To install the audit procedures:

1. The following steps are only for Microsoft SQL Server 2005 and later. If you are using Microsoft SQL Server 2000, proceed to step 2:

   a. Create a directory **C:\MyDBApp**, and place the **RSA_MSSQLAuditStoredProcedures.dll** file inside.

   b. Launch the SQL Server Management Studio.

   c. To enable the Common Language Runtime (CLR), click **New Query**, and type:

   > **Note:** The CLR is disabled by default in SQL Server 2005 and later. You must enable CLR on a server-wide basis. You only need to enable CLR once for each server.

```
EXEC sp_configure 'show advanced options' , '1';
go
reconfigure;
go
EXEC sp_configure 'clr enabled' , '1'
go
reconfigure;
go
```

d. Click **Execute**.

e. To add a certificate to the database and Grant Load permissions, click **New Query**, and type:

```
USE master
GO
CREATE CERTIFICATE SQLCLRTestCert FROM EXECUTABLE FILE =
'C:\MyDBApp\RSA_MSSQLAuditStoredProcedures.dll'
CREATE LOGIN SQLCLRTestLogin FROM CERTIFICATE SQLCLRTestCert
GRANT EXTERNAL ACCESS ASSEMBLY TO SQLCLRTestLogin
GO
```

f. Click **Execute**.

g. To load the .NET Assembly in the SQL server, click **New Query**, and type:

```
CREATE ASSEMBLY RSA_MSSQLAuditStoredProcedures
FROM 'C:\MyDBApp\RSA_MSSQLAuditStoredProcedures.dll'
WITH PERMISSION_SET = EXTERNAL_ACCESS;
GO
```

h. Click **Execute**.

2. Run the appropriate script using the SQL Server Query Analyzer (2000) or SQL Server Management Studio (2005, and later) utility against the **master** database:

Note: These scripts may have changed when you configured custom events.

- For SQL Server 2000, run the **sqlServerAudit2000.sql** script.
- For SQL Server 2005, run the **sqlServerAudit2005.sql** script.
- For SQL Server 2008, run the **sqlServerAudit2008.sql** script.
- For SQL Server 2012, run the **sqlServerAudit2012.sql** script.
- For SQL Server 2014, run the **sqlServerAudit2014.sql** script.
- For SQL Server 2016, run the **sqlServerAudit2016.sql** script.

If you are using SQL Server 2000, you only need to create a user, **audit_reader**, with **sysadmin** privileges. If you are using SQL Server 2005 or later, you must create a SQL Server logon. To create a SQL Server logon, follow these steps:

a. Open the SQL Server Management Studio with administrative credentials, and access the Database Engine.

b. To create a new login, follow these steps:

   i. From the **Object Explorer** navigation menu, expand your database server, which is the top item in the navigation pane.

   ii. Expand **Security**.

   iii. Right-click **Logins** and select **New Login**.

   iv. From the **Select a page** navigation menu, select **General**.

   v. From the **Login name** field, type **audit_reader**.

   vi. Select **SQL Server authentication**.

   vii. Create and confirm a password.

   viii. Ensure that **Enforce Password Expiration** is not selected.

   ix. Click **OK**.

   x. Click **Security** > **Login**, and right-click **audit_reader**.

   xi. Select **Properties**, and from the **Select a page** navigation menu, select **User Mapping**.

   xii. Ensure that **Map** is selected for the **master** database.

   xiii. Click **OK**.

c. To set the logon account permission, follow these steps:

   i. From the **Object Explorer** navigation menu, right-click your database server, and select **Properties**.

   ii. From the **Select a page** navigation menu, select **Permissions**.

   iii. From the **Login or roles** section, select **audit_reader**.

   iv. From the **Explicit permissions** section, select the Grant column for **Alter trace** and **Connect SQL**.

   v. Click **OK**.

d. To set the database access permission, follow these steps:

i. From the **Object Explorer** navigation menu, expand your database server.

ii. Expand **Databases** > **System Databases**.

iii. Right-click **master** and select **Properties**.

iv. From the **Select a page** navigation menu, select **Permissions**.

v. From the **Login or roles** section, select **audit_reader**.

vi. From the **Explicit permissions** section, select the Grant column for **Connect** and **Execute**.

vii. Click **OK**.

Set Up ODBC in RSA NetWitness Platform

To configure ODBC collection in RSA NetWitness Platform, perform the following procedures:

I. Ensure the required parser is enabled

II. Configure a DSN

III. Add the Event Source Type

Ensure the Required Parser is Enabled

If you do not see your parser in the list while performing this procedure, you need to download it in Security Analytics Live.

Ensure that the parser for your event source is enabled:

1. In the **NetWitness** menu, select **ADMIN** > **Services**.
2. In the Services grid, select a Log Decoder, and from the Actions menu, choose **View** > **Config**.
3. In the Service Parsers Configuration panel, search for your event source, and ensure that the **Config Value** field for your event source is selected.

Note: The required parser is **mssql**.

Configure a DSN

Configure a DSN (Data Source Name):

1. In the **NetWitness** menu, select **ADMIN** > **Services**.
2. In the **Services** grid, select a **Log Collector** service.
3. Click ⚙ under **Actions** and select **View** > **Config**.
4. In the Log Collector **Event Sources** tab, select **ODBC/DSNs** from the drop-down menu.
5. The DSNs panel is displayed with the existing DSNs, if any.
6. Click + to open the **Add DSN** dialog.

   > **Note:** If you need to add a DSN template, see the "Configure DSNs" topic in the *Log Collection Configuration Guide*, available in RSA Link.

7. Choose a DSN Template from the drop down menu and enter a name for the DSN. (You use the name when you set up the ODBC event source type.)
8. Fill in the parameters and click **Save**.

Field	Description
DSN Template	Choose the correct template from the available choices.
DSN Name	Enter a descriptive name for the DSN
Parameters section	
Database	Specify the database used by MSSQL
PortNumber	Specify the Port Number. The default port number is **1433**
HostName	Specify the hostname or IP Address of MSSQL
Driver	Depending on your NetWitness Log Collector version: • For 10.6.2 and newer, use /opt/netwitness/odbc/lib/R3sqls27.so • For 10.6.1 and older, use /opt/netwitness/odbc/lib/R3sqls26.so

Field	Description
Note: For encrypted communication, **EncryptionMethod** and **ValidateServerCertificate** are required.	
EncryptionMethod	Set to **1**
ValidateServerCertificate	Set to **0**
GSSClient	Used for Kerberos authentication. This value is your RPM for your Kerberos 5 client. For example, `libgssapi_krb5.so.2`

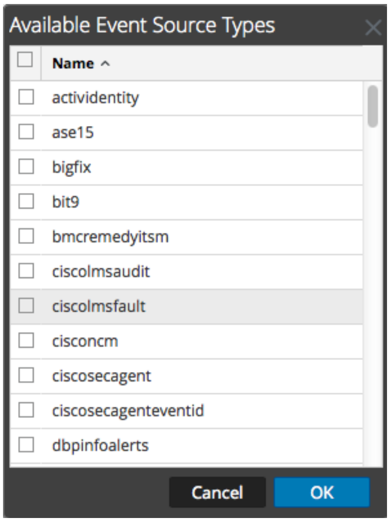
Note: There are more parameters that can be configured, depending your environment. For more details, see the Progress DataDirect site here: http://documentation.progress.com/output/DataDirect/jdbcredshifthelp/index.html#page/redshiftjdbc/connection-property-descriptions.html

Add the Event Source Type

Add the ODBC Event Source Type:

1. In the **NetWitness** menu, select **ADMIN** > **Services**.
2. In the **Services** grid, select a **Log Collector** service.
3. Click ⚙ under **Actions** and select **View** > **Config**.
4. In the Log Collector **Event Sources** tab, select **ODBC/Config** from the drop-down menu.

   The Event Categories panel is displayed with the existing sources, if any.
5. Click + to open the **Available Event Source Types** dialog.



Available Event Source Types

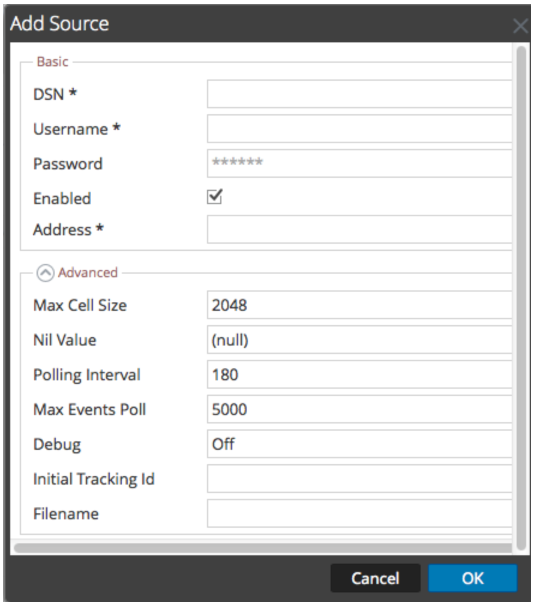
□	Name ^
□	activridentity
□	ase15
□	bigfix
□	bit9
□	bmcremedyitsm
□	ciscolmsaudit
□	ciscolmsfault
□	cisconcm
□	ciscosecagent
□	ciscosecagenteventid
□	dbpinfoalerts

6. Choose the log collector configuration type for your event source type and click **OK**.

   Select **mssql** from the **Available Event Source Types** dialog.

7. In the **Event Categories** panel, select the event source type that you just added.

8. In the **Sources** panel, click + to open the **Add Source** dialog.



Add Source

Basic

DSN *	
Username *	
Password	******
Enabled	☑
Address *	

Advanced

Max Cell Size	2048
Nil Value	(null)
Polling Interval	180
Max Events Poll	5000
Debug	Off
Initial Tracking Id	
Filename	

9. Enter the DSN you configured during the **Configure a DSN** procedure.

10. For the other parameters, see the "ODBC Event Source Configuration Parameters" topic in the *RSA NetWitness Platform Log Collection Guide*.

Troubleshooting ODBC Collection

This section describes the solutions for some common problems encountered when configuring ODBC collection for the Microsoft SQL event source.

Authentication Method

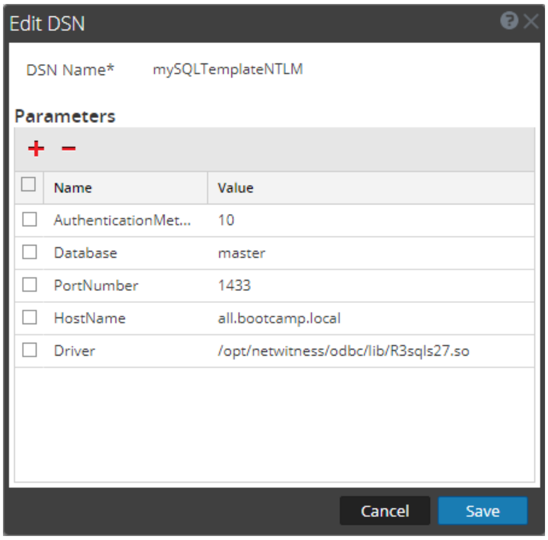
You may need to change the **AuthenticationMethod** from the default value of **1**. From the Progress® DataDirect® documentation (Progress DataDirect Connect for ODBC Version 7.1.6: Authentication Method):

- **Purpose:** Specifies the method the driver uses to authenticate the user to the server when a connection is established. If the specified authentication method is not supported by the database server, the connection fails and the driver generates an error.
- **Valid Values**: 1 | 4 | 9 | 10 | 13 (default is 1)
- **Behavior**:
  - If set to 1 (Encrypt Password), the driver sends the user ID in clear text and an encrypted password to the server for authentication.
  - If set to 4 (Kerberos Authentication), the driver uses Kerberos authentication. This method supports both Windows Active Directory Kerberos and MIT Kerberos environments.

    Setting this value to 4 also enables NTLMv2 and NTLMv1 authentication on Windows platforms. The protocol used for a connection is determined by the local security policy settings for the client.
  - (UNIX and Linux only) If set to 9 on Linux and UNIX platforms, the driver uses NTLMv1 or NTLMv2 authentication. The driver determines which protocol to use based on the size of the password provided. For passwords 14 bytes or less, the driver uses NTLMv1; otherwise, the driver uses NTLMv2. To connect to the database, users must supply the Windows User Id, Password, and, in some cases, Domain to the driver.
  - (UNIX and Linux only) If set to 10, the driver uses NTLMv2 authentication. To connect to the database, users must supply the Windows User Id, Password, and, in some cases, Domain to the driver.
  - If set to 13 (Active Directory Password), the driver uses Azure Active Directory (Azure AD) authentication when establishing a connection to an Azure SQL Database data store. All communications to the service are encrypted using SSL.

    > **Note:** Make sure you have already configured Azure Active Directory Authentication before setting the value to 13.

Example screen shot of connection for NTLM that uses a value of 10 for **AuthenticationMethod**:

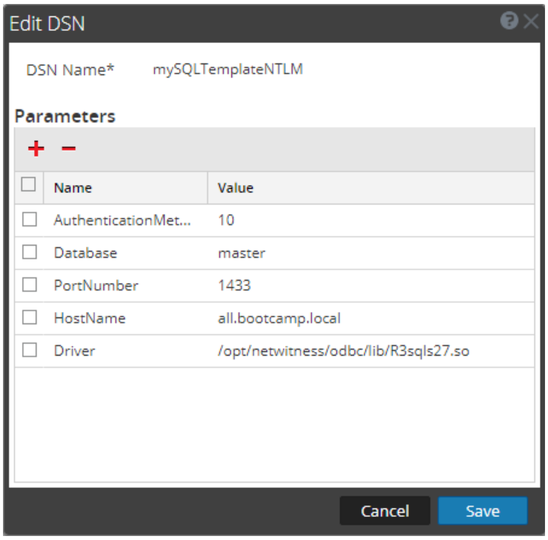


DSN Name* mySQLTemplateNTLM

Parameters

□	Name	Value
□	AuthenticationMet...	10
□	Database	master
□	PortNumber	1433
□	HostName	all.bootcamp.local
□	Driver	/opt/netwitness/odbc/lib/R3sqls27.so

Example screen shot of connection for Kerberos that uses a value of 4 for **AuthenticationMethod**:



DSN Name* mySQLTemplateNTLM

Parameters

□	Name	Value
□	AuthenticationMet...	10
□	Database	master
□	PortNumber	1433
□	HostName	all.bootcamp.local
□	Driver	/opt/netwitness/odbc/lib/R3sqls27.so

Tip

Once the Kerberos or NTLM connection is working, make sure to disable any previous event source (the old SQL credentialed one using **audit_reader** account for example) to that same DB instance. You cannot have more than one event source to same instance or data will be lost, because we are calling the trace function in SQL server and having it write to potentially the same physical location on the SQL Server. For example, the **Filename** field in the advance section of the event source causes a collision when we delete trace data after it has been pulled.

Common Errors

Problem: If you forget to fill in the trace file name that was created when installing the callback DLL (RSA_MSSQLAuditStoredProcedures.dll), then you will get this error (note the word file_template which should be the actual tracefile path):

```
[OdbcCollection] [failure] [mssql.mySQLTemplateClear] [processing]
[mySQLTemplateClear] [processing] Data query failed; dataQuery: exec nic_
aud_swap_trace 30, 'file_template', 1, 'WHERE StartTime > 2018-05-29
15:12:42.133', exception Unable to execute statement: Statement: "exec nic_
aud_swap_trace 30, 'file_template', 1, 'WHERE StartTime > 2018-05-29
15:12:42.133'"; Reason: state: 60; error-code: 19068; description: [RSA]
[ODBC 20101 driver][Microsoft SQL Server]The trace file path is not valid
or not supported.state: 2; error-code: 50000; description: [RSA][ODBC 20101
driver][Microsoft SQL Server]ERROR: Error occured trying to start tracing
for file - 53, file_template-1
```

Problem: If you forget to add the GSSClient configuration entry for Kerberos connection type then you get the following error:

```
An error occurred creating an ODBC connection for DSN: <DSN_Name> The
trapped error is: Unable to create an ODBC connection. DSN: <DSN_Name>;
username: <user_name>; reason: state: 60; error-code: 2733; description:
[RSA][ODBC 20101 driver]2733
```

Problem: If you enter a bad password for NTLM then you get the following error:

```
An error occurred creating an ODBC connection for DSN: <DSN_Name> The
trapped error is: Unable to create an ODBC connection. DSN: <DSN_Name>;
username: <user_name>; reason: state: 08S01; error-code: 0; description:
[RSA][ODBC 20101 driver]7503state: 60; error-code: 18452; description:
[RSA][ODBC 20101 driver][20101]Login failed. The login is from an untrusted
domain and cannot be used with Windows authentication
```

You also get same error if you use Kerberos but specify you MSSQL Server startup as a non-Domain account (because with the a local account SQL Server cannot validate your window credentials against a DC).

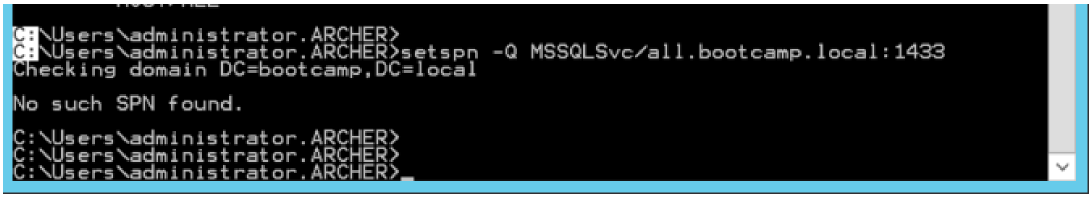
Problem: If there is no TGT in the cache during a poll cycle, you receive the following error:

```
An error occurred creating an ODBC connection.#011 The DSN for this
connection is: <DSN_Name> The trapped error is: Unable to create an ODBC
connection. DSN: <DSN_Name>; username: <user_name>; reason: state: 60; error-
code: 2755; description: [RSA][ODBC 20101 driver]2755state: 60; error-code:
2764; description: [RSA][ODBC 20101 driver]2764
```

Problem: The same error occurs if there is no service ticket in the cache: that is, the driver cannot acquire one (most likely in this case the SPN is missing from the SQL Server. To verify this, open a DOS window as Administrator and run the following command:

```
setspn -Q MSSQLSvc/<DSN_Name>:1433
```

Example:



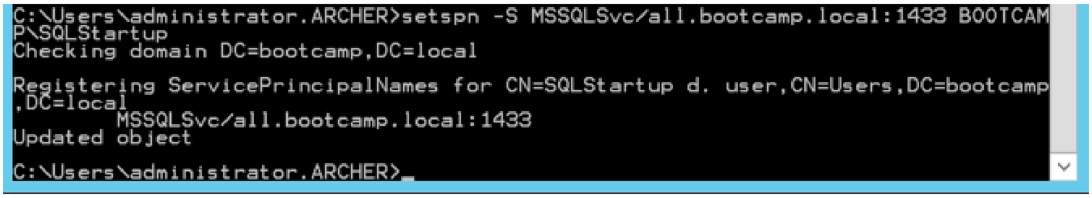
```
C:\Users\administrator.ARCHER>
C:\Users\administrator.ARCHER>setspn -Q MSSQLSvc/all.bootcamp.local:1433
Checking domain DC=bootcamp,DC=local

No such SPN found.

C:\Users\administrator.ARCHER>
C:\Users\administrator.ARCHER>
C:\Users\administrator.ARCHER>_
```

To fix, manually add an SPN which follows the hostname and port of the DB instance. To set an SPN, you need to pass in an account to tie it to. Use the domain account at was used to start the service. For example:

```
setspn -S MSSQLSvc/all.bootcamp.local:1433 BOOTCAMP\SQLStartup
```



```
C:\Users\administrator.ARCHER>setspn -S MSSQLSvc/all.bootcamp.local:1433 BOOTCAM
P\SQLStartup
Checking domain DC=bootcamp,DC=local

Registering ServicePrincipalNames for CN=SQLStartup d. user,CN=Users,DC=bootcamp
,DC=local
        MSSQLSvc/all.bootcamp.local:1433
Updated object

C:\Users\administrator.ARCHER>_
```

Configure Windows Collection

If you have not yet configured WinRM collection for your system, and you want to collect login events from MSSQL via the event logs, then you should use the **winrmconfig.ps1** script. This script is available for downloading from this URL: https://community.rsa.com/docs/DOC-58018. (Note that you need credentials to access this download file.)

Note the following:

- If you are already collecting Application logs via WinRM, you do not need to configure Windows Collection.
- If not, and you wish to collect Application logs, you need to configure collection from the Application event logs to get the MSSQL events, because MSSQL logs to the Application event log.

To configure collection from Application event logs:

1. In the SQL Server Management Studio, Connect to the SQL Server in Object Explorer.
2. Right-click on the SQL Server and choose the Properties option from the pop-up menu.
3. Select Security, then choose the required option from **Login auditing** and click **OK**.

For more details about WinRM collection, please see the following:

- Microsoft WinRM Configuration guide here: https://community.rsa.com/docs/DOC-58163
- Test and Troubleshoot Microsoft WinRM guide here: https://community.rsa.com/docs/DOC-58164

Configure File Collection

To configure File collection for Microsoft SQL Server, set up the SFTP agent and configure the Log Collector for file collection.

Set Up the SFTP Agent

To set up the SFTP Agent Collector, download the appropriate PDF from RSA Link:

- To set up the SFTP agent on Windows, see Install and Update SFTP Agent
- To set up the SFTP agent on Linux, see Configure SFTP Shell Script File Transfer

Configure the Log Collector for File Collection

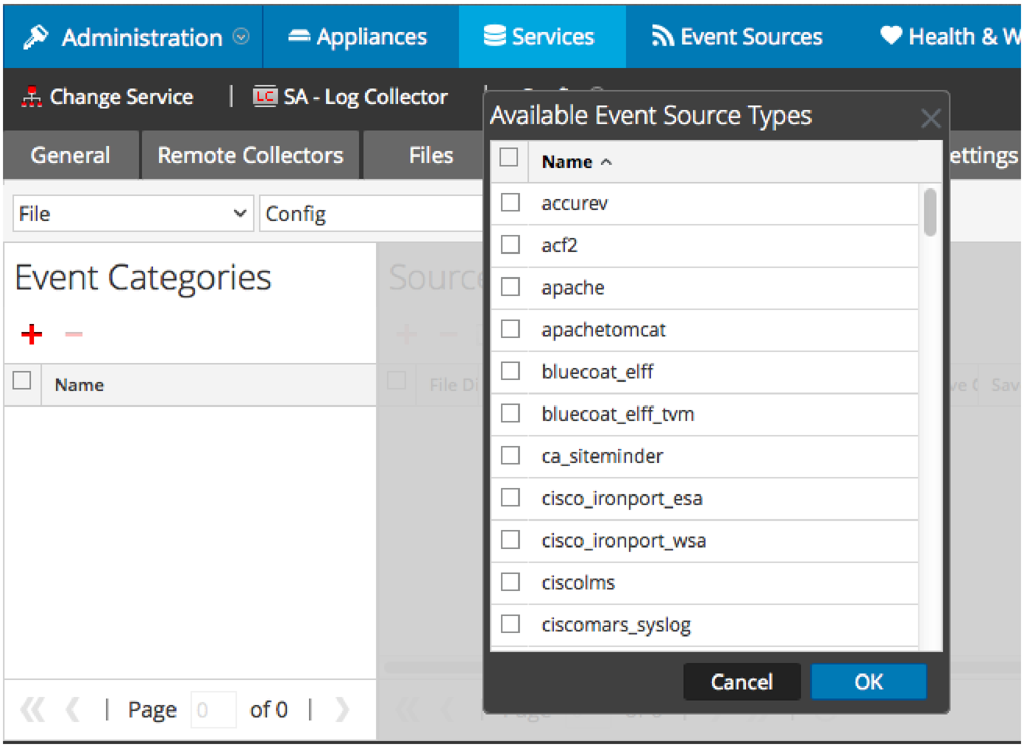
Perform the following steps to configure the Log Collector for File collection.

To configure the Log Collector for file collection:

1. In the **NetWitness** menu, select **Administration** > **Services**.
2. In the Services grid, select a Log Collector, and from the Actions menu, choose **View** > **Config** > **Event Sources**.
3. Select **File/Config** from the drop-down menu.

   The Event Categories panel displays the File event sources that are configured, if any.
4. In the Event Categories panel toolbar, click +.

   The Available Event Source Types dialog is displayed.

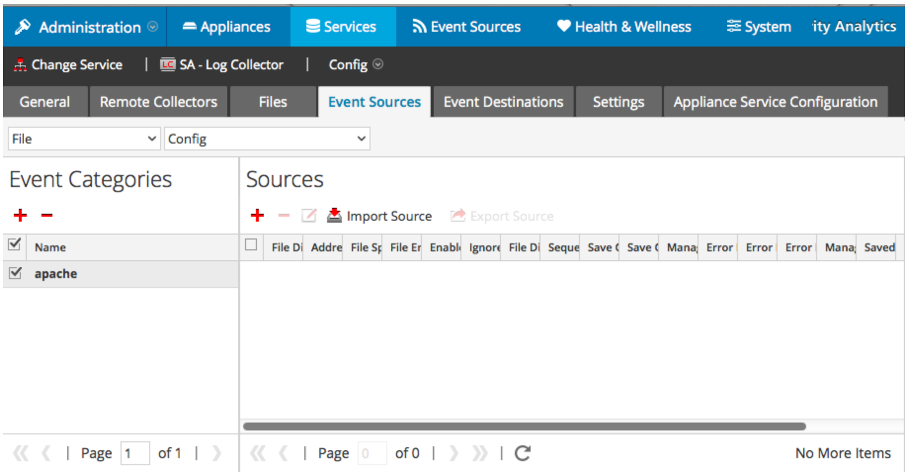


5. Select the correct type from the list, and click **OK**.

Select **mssql** from the **Available Event Source Types** dialog.

The newly added event source type is displayed in the Event Categories panel.

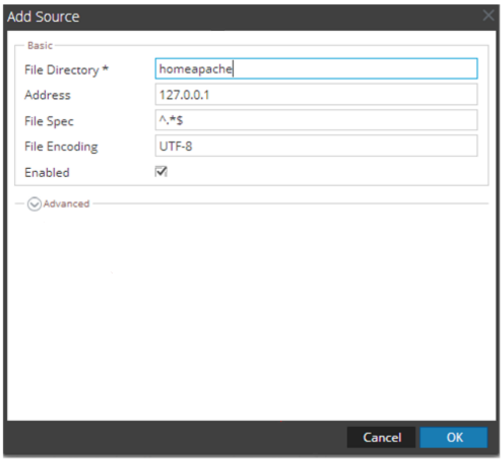
> **Note:** The image below uses **Apache** as an example only. Your screen will look different, depending on which Event Source type you are configuring.



6. Select the new type in the Event Categories panel and click + in the Sources panel toolbar.

   The Add Source dialog is displayed.

   > **Note:** Again, the image below uses **Apache** as an example only. Your screen will look different, depending on which Event Source type you are configuring.



Add Source

Basic

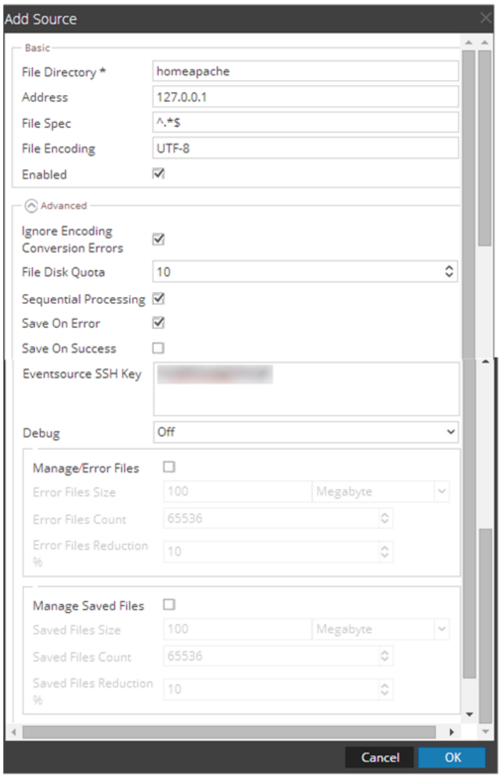
File Directory * homeapache

Address 127.0.0.1

File Spec ^.*$

File Encoding UTF-8

Enabled ☑



Add Source

Basic

File Directory * homeapache

Address 127.0.0.1

File Spec ^.*$

File Encoding UTF-8

Enabled ☑

Advanced

Ignore Encoding Conversion Errors ☑

File Disk Quota 10

Sequential Processing ☑

Save On Error ☑

Save On Success ☐

Eventsource SSH Key [redacted]

Debug Off

Manage/Error Files ☐

Error Files Size 100 Megabyte

Error Files Count 65536

Error Files Reduction % 10

Manage Saved Files ☐

Saved Files Size 100 Megabyte

Saved Files Count 65536

Saved Files Reduction % 10

7. Add a File Directory name, modify any other parameters that require changes, and click **OK**.

8. Stop and Restart File Collection. After you add a new event source that uses file collection, you must stop and restart the NetWitness File Collection service. This is necessary to add the key to the new event source.

Trademarks

For a list of RSA trademarks, go to www.emc.com/legal/emc-corporation-trademarks.htm#rsa.