1.1 PRODUCT DESCRIPTION

DataSunrise Database Security is an application firewall purpose-built to protect relational databases against hacker attacks and insider-driven threats. DataSunrise is compatible with Windows and Linux operating systems, runs fast and independently of any applications and doesn’t inflict any unnecessary load on database server.

DataSunrise can complete the following tasks:

• Data Auditing. DataSunrise performs real-time tracking and logging of all user actions and changes made to target database content. Data auditing results can be exported to an external system, such as SIEM.
• Data Protection. DataSunrise Data Protection intercepts all user queries to target database, detects and blocks unauthorized queries and SQL injections on-the-fly.
• Data Masking. DataSunrise prevents sensitive data exposure due to its dynamic masking capability. DataSunrise Data Masking is capable to hide an entire database or just selected tables or columns from an unwanted user by obfuscating sensitive data in the database output.

1.2 SUPPORTED DATABASES

DataSunrise is compatible with the following DBMSs:

• Oracle Database 9.2-12.1 working on Windows, Linux, Solaris (sparc) or IBM AIX servers
• PostgreSQL 7.4-9.6
• Netezza 6.0-7.2.1
• Greenplum 4.2-4.3
• IBM DB2 9.7-11.1
• MS SQL Server 2005-2016
• Amazon Aurora
• Amazon Redshift
• MariaDB 5.1-10.2
• MySQL 5.0-5.7
• Teradata 13-15
• Hive 1.0-2.1

1.3 DATASUNRISE OPERATION MODES

DataSunrise can be deployed in one of the following configurations: *Sniffer mode* or *Proxy mode*. 

---

General Information | 3
1.3.1 SNIFTER MODE

When deployed in Sniffer mode, DataSunrise is connected to a SPAN port of a network switch. Thus it acts as a traffic analyzer capable to capture copy of the database traffic from network switch's "mirrored" port.

Figure 1: Sniffer mode operation scheme.

In this configuration, DataSunrise can't interfere database traffic, so it is able to perform data auditing only. But running DataSunrise in sniffer mode does not require any additional database or client application tweaking.

1.3.2 PROXY MODE

When deployed in this configuration, DataSunrise works as an intermediary between the database server and its client applications. Thus it is able to process all incoming queries before redirecting them to the database server.

Figure 2: Proxy mode operation scheme.

DataSunrise intercepts SQL queries sent to protected database by database users, checks if they comply with existing security policies, and audits, blocks or modifies incoming queries or query results if necessary. When running in proxy mode DataSunrise sports its full functionality: data audit, database firewall, both dynamic and static data masking are available.
1.4 SYSTEM REQUIREMENTS

Before installing DataSunrise make sure that your server meets the following requirements:

- Operating system: 64-bit Windows (Windows Server 2008 or higher)
- CPU: 8 cores
- RAM: 8-16 Gb
- Available disk space: 3GB. 100 GB for storing audit records if necessary

1.5 USEFUL RESOURCES

- DataSunrise official web site: https://www.datasunrise.com/
- DataSunrise latest version download page: https://www.datasunrise.com/download
- DataSunrise Facebook page: https://www.facebook.com/datasunrise/
- DataSunrise administration guide for Linux (DataSunrise_Database_Security_Suite_Admin_Guide_Linux.pdf file located in the doc subfolder within the program installation folder). Describes installation and post-installation procedures, deployment schemes, includes troubleshooting subsection.
- DataSunrise administration guide for Windows (DataSunrise_Database_Security_Suite_Admin_Guide_Windows.pdf file located in the doc subfolder. Describes installation and post-installation procedures, deployment schemes, includes troubleshooting subsection.
- DataSunrise end user guide (DataSunrise_Database_Security_Suite_User_Guide.pdf file located in the doc subfolder). Describes GUI structure, program managing etc.
- Command Line Interface (CLI) guide (CLI_guide.pdf file located in doc subfolder). Contains CLI commands description, usage examples etc.
- Release notes (Release_notes.pdf file in doc subfolder). Describes changes and enhancements made in the latest DataSunrise version, known bugs and version history.
- EULA (DataSunrise_EULA.pdf file in doc subfolder). Contains End User License Agreement.
DEPLOYMENT TOPOLOGIES

DataSunrise can be installed either on the database server or on a separate server. In both cases, the program can be used in the sniffer mode and in the proxy mode.

2.1 INSTALLING DATASUNRISE ON A DATABASE SERVER

Figure 3: Deployment on a database server

2.1.1 PROXY MODE
To deploy DataSunrise in proxy mode, use one of the following methods:

A) DATABASE SETTINGS TWEAKING
• Reconfigure the database to use some free port on local interface (localhost). This eliminates the possibility to connect to the database directly by bypassing DataSunrise
• Configure DataSunrise proxy to use the port formerly used by the database to connect with the client applications. Thus, any clients trying to connect to the database will connect DataSunrise instead
• Configure DataSunrise connection with the database considering changes made in the previous steps.

Important: many operating systems reserve port numbers less than 1024 for privileged system processes. So it's preferable to use port numbers higher than 1024.

B) RECONFIGURING OF CLIENT APPLICATIONS
• Configure DataSunrise proxy to use any free port
• Configure all the client applications to connect to DataSunrise instead of the database

Tip: you can use this installation option during firewall testing, since some DB clients still retain direct access to the database. Use another firewall to block direct access to the database.

2.1.2 SNIFFER MODE
Configure DataSunrise sniffer. It is not required to tweak any client applications or database settings.
2.2 INSTALLING DATASUNRISE ON A SEPARATE SERVER

2.2.1 PROXY MODE

Figure 4: Proxy mode deployment scheme

To deploy DataSunrise in the proxy mode, perform the following:

• Configure DataSunrise connection with the database.
• Configure all the client applications to connect to DataSunrise proxy instead of the database.

Important: many operating systems reserve port numbers less than 1024 for privileged system processes, so it’s preferable to use port numbers higher than 1024.

2.2.2 SNIFER MODE

Figure 5: Sniffer mode deployment scheme

To deploy DataSunrise in sniffer mode, configure your network switch for transferring mirrored traffic to DataSunrise (refer to your network switch’s user guide for port mirroring procedure description).
Note: Before you begin DataSunrise installation process, please select an appropriate firewall deployment option (subsections 2.1 and 2.2) and perform all required preparations. Also make sure that a PC you want install DataSunrise on, meets system requirements listed in subsection 1.4.

3.1 REQUIRED COMPONENTS

Depending on RDBMS used it is necessary to install some additional components. Please note that you should use 64-bit drivers and components.

1. Install WinPcap library:
   http://www.winpcap.org/install/default.htm

2. To run DataSunrise with MySQL and PostgreSQL databases, install ODBC driver. You can download it here:
   http://www.postgresql.org/ftp/odbc versions/

3. To run DataSunrise with Oracle databases, install OCI driver. You can download it here:

   Note: having installed the Oracle Instant Client, add its home directory path to the %ORACLE_HOME% environment variable and to the %PATH% variable.

   And install Visual C++ 2010 package. You can download it here:


4. To run DataSunrise with Netezza database, install dedicated ODBC driver. Download it from IBM Fix Central:
   http://www-933.ibm.com/support/fixcentral/

   Note: your IBM ID should be associated with your IBM customer ID with active support and maintenance contract for Netezza appliance

   Refer to the following page for more details: https://www-304.ibm.com/support/knowledgecenter/SSULQD_7.0.3/com.ibm.nz.adm.doc/c_sysadm_client_software_packages.html

5. To run DataSunrise with DB2 databases, install ODBC driver. You can download it here:

6. To run DataSunrise with SQL Server, you might need to install ODBC driver. You can download it here:

7. To run DataSunrise with Hive, install Hortonworks ODBC driver. You can download it here:
   ODBC driver: https://hortonworks.com/downloads/

3.2 PROGRAM INSTALLATION

To install DataSunrise on your PC, perform the following:

1. Double-click DataSunrise installer file (DataSunrise Database Security Suite XXX.msi)
2. Follow the steps of the setup wizard
Note: Set password for DataSunrise administrator at the Set administrator password tab

3. If necessary, replace DataSunrise SSL certificate with a new one (refer to 4.2.2).

3.3 PROGRAM REMOVAL

To uninstall DataSunrise perform the standard program removal procedure (using Control panel) or use the method described below:

1. Double-click DataSunrise installation file
2. Click **Remove** button to initiate program removal

   **Note:** click **Repair** button to fix corrupted DataSunrise installation.

3. Follow the steps of the setup wizard.

3.4 DATASUNRISE INSTALLATION FOLDER

This subsection describes DataSunrise files and installation folder structure.

![DataSunrise files and folders](image)

Figure 6: DataSunrise files and folders

1. DataSunrise folders:

<table>
<thead>
<tr>
<th>Folder name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>cmdline</td>
<td>Contains DataSunrise Command Line Interface (CLI) files</td>
</tr>
<tr>
<td>doc</td>
<td>Contains DataSunrise docs (User guide, CLI guide, Release notes, EULA)</td>
</tr>
<tr>
<td>gwt</td>
<td>Contains GUI files</td>
</tr>
<tr>
<td>logs</td>
<td>Log files (back end, core, GUI logs)</td>
</tr>
</tbody>
</table>
2. DataSunrise files (except DLL files):

<table>
<thead>
<tr>
<th>File name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>audit.db</td>
<td>Database file to store audit data (Audit Storage)</td>
</tr>
<tr>
<td>dictionary.db</td>
<td>Contains program settings, firewall objects (such as database entries, user entries etc), rules, etc.</td>
</tr>
<tr>
<td>event.db</td>
<td>System events logs</td>
</tr>
<tr>
<td>standart_application_queries.db</td>
<td>Contains queries used by Oracle SQL Developer (refer to Query Groups subsection for more information)</td>
</tr>
<tr>
<td>install_firewall_service.bat</td>
<td>This script installs DATA_SUNRISE_SECURITY_SUITE service (it is run by setup wizard during program installation)</td>
</tr>
<tr>
<td>remove_firewall_service.bat</td>
<td>This script removes DATA_SUNRISE_SECURITY_SUITE service (it is run by setup wizard during program installation)</td>
</tr>
<tr>
<td>start_firewall_service.bat</td>
<td>This script starts DATA_SUNRISE_SECURITY_SUITE service</td>
</tr>
<tr>
<td>stop_firewall_service.bat</td>
<td>This script stops DATA_SUNRISE_SECURITY_SUITE service</td>
</tr>
<tr>
<td>AppBackendService.exe</td>
<td>System process required for GUI operation and AppFirewallCore.exe control</td>
</tr>
<tr>
<td>AppFirewallCore.exe</td>
<td>Core process. Performs all fundamental DataSunrise functions</td>
</tr>
<tr>
<td>appfirewall.pem</td>
<td>SSL certificate for GUI</td>
</tr>
<tr>
<td>cacert.pem</td>
<td>SSL certificate required for online update</td>
</tr>
<tr>
<td>proxy.pem</td>
<td>OpenSSL keys and certificated used for proxy on default</td>
</tr>
<tr>
<td>appfirewall.reg</td>
<td>Contains DataSunrise license key</td>
</tr>
</tbody>
</table>

3.5 UPDATING DATASUNRISE

To update DataSunrise, perform the following:
1. Go to the **System Settings** -> **About** subsection  
2. Click **Update** button
3. Wait for update to complete and reload the GUI page.

**Note:** You can also update the program in another way. Download the newest version of DataSunrise from the official web site and run the installation file. Follow the steps of the setup wizard to update the program.

### 3.6 MIGRATING DATASUNRISE TO OTHER SERVER

To export DataSunrise settings to other instance installed on other server, perform the following:

1. Stop DataSunrise system service (DATA_SUNRISE_SECURITY_SUITE) using Windows Task Manager
2. Copy dictionary.db, event.db and audit.db files from the source DataSunrise installation folder
3. Install new DataSunrise instance on another server. Stop DataSunrise system service
4. Paste dictionary.db, event.db and audit.db files to new DataSunrise instance installation folder
5. Start new DataSunrise system service
6. Check imported settings.
STARTING DATASUNRISE FOR THE FIRST TIME

4.1 STARTING DATASUNRISE

1. The firewall needs DATA_SUNRISE_SECURITY_SUITE service running to operate. This service starts DataSunrise back end and core on Windows startup.
   - If you've stopped DataSunrise process or it's stopped because of a problem of some kind, you can start the process manually by double-clicking the AppBackendService.exe file located in program installation folder.

2. Enter DataSunrise web interface (refer to subs. 4.2).

4.2 CONNECTING TO DATASUNRISE WEB INTERFACE

DataSunrise is provided with comprehensive web-based interface used to control all the firewall actions.

1. To enter the web interface, perform the following:
   - To connect to GUI using HTTPS protocol (on default), open the following address via your web browser:
     https://DataSunrise_ip_address:11000
     
     **Note:** DataSunrise_ip_address is DataSunrise's IP address or host name, 11000 is the firewall's port number.

   - If you want to connect to DataSunrise using HTTP protocol, you should activate HTTP in system settings (System Settings → General → Ports). Then open the following address via your web browser:
     http://DataSunrise_ip_address:11000
     
     **Note:** DataSunrise_ip_address is DataSunrise's IP address or host name, 11000 is the firewall's port number.

2. You browser will display "Unsecure connection" prompt because of untrusted SSL certificate. That's normal. Follow your browser's prompts to confirm security exception for DataSunrise GUI.

3. Enter your credentials (you've set the password while installing the program) and click Login button to enter the web interface

   **Important:** on first startup, use admin username.

4.2.1 RESTORING ACCESS TO GUI IF THE PASSWORD IS LOST

You can't restore DataSunrise administrator password if you've lost it, but you can set a new one. To change admin user's password, perform the following:

1. Start Windows Command Prompt as an administrator.

2. Use cd command to go to the DataSunrise installation folder (C:\Program Files\DataSunrise Database Security Suite on default)

3. Run AppBackendService.exe file with set_admin_password parameter. Specify a new password as the parameter's value:
Starting DataSunrise for the first time

13

AppbackEndService.exe set_admin_password=new_password

4. Restart DataSunrise service for changes to take effect.

4.2.2 CREATING A CERTIFICATE FOR UI

On the first startup, a web browser used to access DataSunrise GUI, warns about unsecure connection and prompts to add a security exception for the GUI. This issue is caused by DataSunrise's self-signed SSL certificate. To avoid this, you can use a signed SSL certificate from a certain certification authority. For example, you can get such a certificate for free from Let's Encrypt service.

1. Refer to the following link for a guide on obtaining a certificate from Let's Encrypt: https://www.datasunrise.com/blog/getting-an-ssl-certificate-with-lets-encrypt/

2. Paste the private key and the certificate you got from Let's Encrypt into appfirewall.pem file and move the file to the DataSunrise installation folder.

4.3 PRODUCT REGISTRATION

The first time you start DataSunrise, you will be prompted to register it.

1. Paste the license key you got from the firewall developers into the dedicated text field

   Note: You can also paste the license key into appfirewall.reg within the program installation folder.

2. Click Save button

4.4 PREPARING YOUR DATABASE

DataSunrise interacts with target DB and receives all information required for operation through a user account of a target database (the account, user name and password of which are specified in the target database profile). On default, the firewall uses DB's administrator account but it is possible to use any other account with sufficient privileges.

This sections describes actions required to establish connection between DataSunrise and various databases

4.4.1 CREATING AN ORACLE DATABASE USER

1. Connect to the Oracle target database using SYS user account.

2. To create a new user, perform the following:
   • For Oracle 11 g Release 2 or earlier. Run the following command:

     CREATE USER DataSunrise_user IDENTIFIED BY DataSunrise_password;

   • For Oracle 12 c. Create a global user (for all Oracle containers). Connect to CDB$ROOT and run the following command:

     CREATE USER c##DataSunrise_user IDENTIFIED BY DataSunrise_password;

     You can also create a local user (for one container). To do this run the following commands:

     ALTER SESSION SET CONTAINE = pdborcl;
     CREATE USER DataSunrise_user IDENTIFIED BY DataSunrise_password;
Warning: in most cases it is preferable to use global user for establishing connection with target databases, because if you use local user (created for one container) DataSunrise would not be able to work with other containers.

3. Grant all required privileges to new user if necessary. To do this, run the following commands:
   - For Oracle 11g Release 2 or earlier:
     ```sql
     GRANT CONNECT TO DataSunrise_user;
     GRANT SELECT on "SYS"."DBA_OBJECTS" TO DataSunrise_user;
     GRANT SELECT on "SYS"."DBA_TAB_COLUMNS" TO DataSunrise_user;
     GRANT SELECT on "SYS"."DBA_SYNONYMS" TO DataSunrise_user;
     GRANT SELECT on "SYS"."DBA_NESTED_TABLES" TO DataSunrise_user;
     GRANT SELECT on "SYS"."V_$SERVICES" TO DataSunrise_user;
     GRANT SELECT on "SYS"."V_$INSTANCE" TO DataSunrise_user;
     GRANT SELECT on "SYS"."DBA_USERS" TO DataSunrise_user;
     GRANT SELECT on "SYS"."DBA_PROCEDURES" TO DataSunrise_user;
     ```
   - For Oracle 12c. For a local user:
     ```sql
     GRANT SELECT on "SYS"."V_$SERVICES" to DataSunrise_user;
     GRANT SELECT on "SYS"."CDB_USERS" to DataSunrise_user;
     GRANT SELECT on "SYS"."CDB_OBJECTS" to DataSunrise_user;
     GRANT SELECT on "SYS"."CDB_TAB_COLUMNS" to DataSunrise_user;
     GRANT SELECT on "SYS"."CDB_SYNONYMS" to DataSunrise_user;
     GRANT SELECT on "SYS"."CDB_NESTED_TABLES" to DataSunrise_user;
     GRANT SELECT on "SYS"."V_$INSTANCE" to DataSunrise_user;
     GRANT SELECT on "SYS"."CDB_PROCEDURES" to DataSunrise_user;
     GRANT CREATE TABLE to DataSunrise_user;
     ```

   Tip: you can create a required table manually instead of giving CREATE TABLE privilege to new user:
   ```sql
   create global temporary table DAF_OBJECTS ON COMMIT DELETE ROWS as select * from CDB_OBJECTS where 1 != 1;
   ```

   To grant required privileges to a global user, run the following commands:
   ```sql
   GRANT CONNECT to c##DataSunrise_user CONTAINER=ALL;
   GRANT SYSDBA to c##DataSunrise_user;
   ```

4.4.2 CREATING POSTGRESQL OR REDSHIFT DATABASE USER

To create a PostgreSQL or Redshift user, run the following command:
```sql
CREATE USER DataSunrise_user WITH PASSWORD 'DataSunrise_password';
```

Note: the user should be able to get information about database structure from the following system tables:
- pg_database
- pg_namespace
- pg_class
- pg_catalog
- pg_attribute
- pg_user
- pg_settings
4.4.3 CREATING NETEZZA DATABASE USER

To create new Netezza user, run the following command:

```
CREATE USER DataSunrise_user WITH PASSWORD 'DataSunrise_password';
```

**Note:** grant all required privileges to the new user. Connect to the SYSTEM database and send it an appropriate SQL query:

- For Netezza 6.X:
  ```
  GRANT LIST ON AGGREGATE, DATABASE, EXTERNAL TABLE, FUNCTION, GROUP, MANAGEMENT TABLE, MANAGEMENT VIEW, PROCEDURE, SEQUENCE, SYNONYM, SYSTEM TABLE, SYSTEM VIEW, TABLE, USER, VIEW to DataSunrise_user;
  ```
- For Netezza 7.X:
  ```
  GRANT LIST ON AGGREGATE, DATABASE, EXTERNAL TABLE, FUNCTION, GROUP, MANAGEMENT TABLE, MANAGEMENT VIEW, PROCEDURE, SCHEMA, SEQUENCE, SYNONYM, SYSTEM TABLE, SYSTEM VIEW, TABLE, USER, VIEW to DataSunrise_user;
  ```

4.4.3.1 ENABLING "REXP REPLACE" DATA MASKING IN NETEZZA

IBM Netezza does not support regular expressions on default, so it is impossible to use "Regexp replace" out of the box. To enable Regexp masking, it is required to install additional Netezza package:


4.4.4 CREATING GREENPLUM USER

To create Greenplum user, run the following command:

```
CREATE USER DataSunrise_user WITH PASSWORD 'DataSunrise_password';
```

4.4.5 GRANTING NECESSARY PRIVILEGES TO DB2 USER

To make DataSunrise work correctly with DB2 database it’s necessary to grant database user rights to select data from the following system views:

- `syscat.schemata`
- `syscat.procedures`
- `syscat.functions`
- `syscat.tables`
- `syscat.columns`
- `syscat.sequences`
- `syscat.packages`

To grant necessary user privileges, run the following script:

```
GRANT SELECT ON SYSCAT.COLUMNS TO USER DataSunrise_user;
GRANT SELECT ON SYSCAT.FUNCTIONS TO USER DataSunrise_user;
```
4.4.6 Configuring MS SQL Server Connection

To establish connection between DataSunrise and SQL Server database, perform the following:

1. Run SQL Server configuration manager utility (it is included in SQL Server pack). Open SQL Server Network Configuration -> Protocols for (DB instance name)

2. Right-click on TCP/IP protocol name and select Properties in the context menu

3. In the TCP/IP Properties window, in the Protocol tab, set Yes value for Enabled parameter. Then open IP-addresses tab, IPA11 subsection and set TCP-port parameter value to 1433. Click OK to close the window

4. Open SQL Server Services subsection, right-click on SQL Server (DB instance name) parameter to open its context menu, and click Restart

5. If you’re using some firewall application (including Windows Firewall), you should allow the following inbound connections: TCP/IP, port 1433 and UDP, port 1434

6. When configuring is done, it is recommended to restart your PC.

7. Connect to the database server with SQL Server Management Studio (SSMS)

**Important:** SSMS’s Encrypt connection option forces encryption and server certificate check on client’s side (except SSMS 2016 and higher). Thus when this option is enabled, the client would not be able to connect to DataSunrise proxy if the certificate included into proxy.pem or dictionary.db does not include proxy’s host name. In the case when encryption is enabled (it is disabled on default), it is necessary to have a properly signed SSL certificate. Otherwise, disable Encrypt connection.

**Important:** use SQL Server authentication instead of Windows authentication. Otherwise, refer to subs. 4.5.3

When configuring database connection, specify the database server’s host name or IP address instead of SPN.

4.4.6.1 Granting Necessary Privileges to SQL Server User

To make DataSunrise work correctly with SQL Server database, it’s necessary to grant database user rights for fetching metadata properly. To grant necessary user privileges, use VIEW DEFINITION as described below

1. Grant the rights to view metadata in MASTER database

   ```sql
   GRANT VIEW DEFINITION ON DATABASE::MASTER to [user_name];
   ```

2. Grant SELECT rights for the following tables:
   - [master].[sys].[databases]
   - [all_bases].[sys].[database_principals]
   - [all_bases].[sys].[database_permissions]
   - for SQL Azure: [master].[sys].[sql_logins] or: [master].[sys].[server_principals]

   Use the following query for the forementioned tables:

   ```sql
   USE MASTER;
   GO
   GRANT SELECT ON OBJECT::sys.databases to [user_name];
   GO
   ```
4.4.6.2 Enabling "Regexp Replace" data masking in SQL Server

SQL Server database does not support regular expressions but provides a possibility to use external addons by plugging them in the server through DLLs. "Regexp replace" masking function is built as an external addon as well. The key point here is that an addon should be plugged in a specific database, thus it could be used only inside a specific database and schema. Thus there are two ways you can use "Regexp replace" masking in MS SQL Server:

1. Plug the addon into each database when installing the firewall, and use default schema (DBO). It allows to skip database when calling the masking function.

   ```
   SELECT [DBO].[RegexReplace]('9731246ab456cde', '[a-z]{2}', '__') AS "T2"
   ```

2. Plug the addon into the database and schema by default (MASTER.DBO).

   ```
   SELECT [MASTER].[DBO].[RegexReplace]('9731246ab456cde', '[a-z]{2}', '__') AS "T2"
   ```

   **Important:** in either case it would be necessary to grant the DB user a privilege to run RegexpReplace. You can do it with the following query:

   ```
   GRANT EXECUTE ON [MASTER].[DBO].[RegexReplace] to [name of user to obtain the privilege]
   ```

You can install the following function to enable RegExp data masking.

```csharp
using System;
using Microsoft.SqlServer.Server;
using System.Text.RegularExpressions;

public partial class RegExBase{
    [SqlFunction(IsDeterministic = true, IsPrecise = true)]
    public static string RegexReplace(string input, string pattern, string replacement){
        return Regex.Replace(input, pattern, replacement);
    }
};
```

You can download the full script here: [https://www.datasunrise.com/support-files/ms_sql_regexp_replace.sql](https://www.datasunrise.com/support-files/ms_sql_regexp_replace.sql)

**Note:** more on user-defined functions here: [https://msdn.microsoft.com/en-us/library/w2kae45k(v=vs.80).aspx](https://msdn.microsoft.com/en-us/library/w2kae45k(v=vs.80).aspx)

4.5 Additional Proxy Configuration

4.5.1 Changing PostgreSQL port number

When configuring DataSunrise proxy it would be necessary to change database port number. It is necessary if DataSunrise proxy is configured to use the port number assigned to original database. To do this, perform the following:

1. Open the postgresql.conf file which is located in the data subfolder of PostgreSQL installation folder.
2. In the CONNECTIONS AND AUTHENTICATION sections, change port parameter value (5432 on default) to the new port number.
3. Restart PostgreSQL for changes to take effect.
4.5.2 Configuring Authorization of Local Users in PostgreSQL

If DataSunrise proxy is deployed on the same host as the database is, remote users which connect to the database through proxy, would be treated by the database as the local users, thus they can have some preferences like password-free or simplified authorization. Thus it is necessary to disable password-free authorization for local users in the database settings, if it is enabled. To do this, perform the following:

1. Open the pg.hba file which is located in the data subfolder of PostgreSQL installation folder.
2. Edit pg.hba file in the following way:

   # TYPE DATABASE USER ADDRESS METHOD
   # IPv4 local connections:
   host all all 127.0.0.1/32 md5
   host all all all md5
   # IPv6 local connections:
   host all all ::1/128 md5
3. As a result, MD5 or Password authentication method should be assigned for all database connections.

4.5.3 Configuring Windows Authentication for Microsoft SQL Server

On default, SQL Server authorization is used to access the database. If it is required to use Windows Authentication, and DataSunrise, database server and client applications are installed on separate machines, it is necessary to use Active Directory (AD) service.

When working with AD, SSPI user authorization is used (based on NTLM or Kerberos protocols). Since Kerberos-based authorization is preferable, it is necessary to perform the following to activate this protocol:

1. Enable delegation for the DataSunrise proxy's host account. Enter Active Directory Users and Computers and find a profile of a machine DataSunrise is installed on. Open its properties —> Delegation tab and enable the Trust this computer for delegation to any service switch.
2. Proxy's address should match or resolve a name into a registered SPN (more on SPNs here: https://msdn.microsoft.com/en-en/library/ms191153.aspx) for Kerberos connection (MSSQLSvc service). To do this, use Setspn.exe tool which is supplied with Windows Server's support tools, to register two required SPNs for a profile of a machine for which the delegation is enabled:

   ```
   setspn -A MSSQLSvc/proxy-host:proxy-port proxy-host
   setspn -A MSSQLSvc/full-fqdn-proxy-host:proxy-port proxy-host
   ```

   For example:

   ```
   setspn -A MSSQLSvc/varmor-03:1435 varmor-03
   setspn -A MSSQLSvc/varmor-03.db.local:1435 varmor-03
   ```

   **Important:** run setspn.exe as a domain administrator or as a domain user with a privilege of "Validated write to service principal name" for AD object for which it is necessary to configure SPN.
Starting DataSunrise for the first time

To grant this privilege, go to *Active Directory Users and Computers*, select a server the database is installed on, open its properties —> Security tab, add a required user and check the "Validated write to service principal name" check box. More information here: [https://technet.microsoft.com/en-us/library/cc731241(v=ws.10).aspx](https://technet.microsoft.com/en-us/library/cc731241(v=ws.10).aspx)

**Note:** you can get a list of all registered SPNs with this command:

```bash
setspn -L proxy-host
```

To delete an SPN, execute:

```bash
setspn -D MSSQLSvc/proxy-host:proxy-port proxy-host
```

To check an authorization scheme, connect to the server and execute the following query:

```sql
SELECT auth_scheme FROM sys.dm_exec_connections WHERE session_id = @spid
```

The query result will show an authorization scheme used by the database server (SQL, NTLM or Kerberos)

### 4.6 PROCESSING ENCRYPTED TRAFFIC

This subsection describes how to configure encrypted traffic processing.

#### 4.6.1 Configuring SSL encryption for DB2

To configure DataSunrise to process SSL-encrypted traffic, perform the following:

1. Prepare DB2 server for working with SSL. You need to get certificate server delivers to client during SSL connection (hereafter db2_server.crt). Refer to the following page for example: [http://www.ibm.com/support/knowledgecenter/SSEP_10.5.0/com.ibm.db2.luw.admin.sec.doc/doc/t0025241.html](http://www.ibm.com/support/knowledgecenter/SSEP_10.5.0/com.ibm.db2.luw.admin.sec.doc/doc/t0025241.html)


3. Specify full path to certificate storages in Db2KeyStoragePath and Db2KeyStashPath parameters (refer to subs. 6.1.3)


#### 4.6.2 Configuring SSL for Microsoft SQL Server
4.6.2.1 Enabling SSL Encryption for MS SQL Server
To configure DataSunrise to process SSL-encrypted traffic, perform the following:

1. Install MakeCert utility (it is included in Windows SDK). You can download Windows SDK from this page: https://www.microsoft.com/en-us/download/details.aspx?id=8279

2. Run the following command to create new certificate:

```
makecert -r -pe -n "CN= SERVER_HOST" -b 01/01/2016
-e 01/01/2036 -eku 1.3.6.1.5.5.7.3.1 -ss my
-sr localMachine -sky exchange
-sp "Microsoft RSA SChannel Cryptographic Provider" -sy 12
```

Replace SERVER_HOST with actual SQL Server host name and set required certificate lifetime.

3. Run SQL Server Configuration Manager utility and select SQL Server Network Configuration —> Protocols for (DB instance_name)

4. Right-click on Protocols for... and select Properties

5. In Certificate tab select certificate generated in step 2 of this instruction.

6. In Flags tab you may set Force Encryption parameter to Yes to encrypt all TDS traffic. Or set it to No to encrypt client authorization packet only.

7. Restart SQL Server. To do this, select SQL Server Services —> SQL Server (DB instance name) and click Restart Service.

   **Important:** Refer to this page for more information: https://thesqldude.com/tag/makecert/

4.6.2.2 Generating an SSL Certificate with OpenSSL
To create an SSL certificate for SQL Server using OpenSSL, perform the following:

1. Create configuration file named `config.cfg`:

```
[req]
distinguished_name = req_distinguished_name
prompt = no

[req_distinguished_name]
countryName = USA
stateOrProvinceName = Washington
localityName = Seattle
organizationName = DataSunrise
organizationalUnitName = IT
commonName = SERVER_HOST
emailAddress = support@myemail.local

[ext]
extendedKeyUsage = 1.3.6.1.5.5.7.3.1
```

   **Note:** replace SERVER_HOST with actual SQL Server host name.

2. Run the following script:

```
openssl genrsa -des3 -out key.pem 2048
openssl rsa -in mssql-rsa.pem -out mssql-rsa.pem
openssl req -config config.cfg -new -key key.pem -out req
openssl req -x509 -config config.cfg -extensions ext -days 365 -key key.pem -in req -out certificate.cer
openssl pkcs12 -export -in certificate.cer -inkey key.pem -out certificate.pfx
```
Starting DataSunrise for the first time

Note: when executing the first command you would need to enter some password twice. The second command resets the password, but you would need to enter it once more. The third command creates certificate request within req file. The fourth command generates self-signed certificate within certificate.cer file. The last command packs the key and the certificate into certificate.pfx file, protecting it with password (enter password twice). Then you should import certificate.pfx via MMC console to Personal container.

3. Install certificate for proxy (refer to subs. 4.5.2.3, Step 4)

4.6.2.3 Generating a signed SSL certificate with OpenSSL

If certificate check is enabled (checked “Encrypt connection” check box for SSMS lower that 2016 and unchecked “Trust server certificate” for SSMS 2016) you should use a signed SSL certificate

To generate a certificate with OpenSSL, perform the following.

1. Prepare required infrastructure:
   
   ```
   mkdir db
   mkdir db\new
   mkdir db\private
   echo. 2>db\index
   echo 01>./db/serial
   echo unique_subject = no>./db/index.attr
   ```

2. Create “ca” configuration file:

   ```
   [req]
   distinguished_name = req_distinguished_name
   prompt = no
   RANDFILE = ./db/private/.rand

   [req_distinguished_name]
   countryName = US
   stateOrProvinceName = Washington
   localityName = Seattle
   organizationName = DataSunrise
   organizationalUnitName = IT
   commonName = DataSunrise
   emailAddress = support@datasunrise.com
   ```

3. Create “cfg” configuration file:

   ```
   [req]
   distinguished_name = req_distinguished_name
   prompt = no
   RANDFILE = ./db/private/.rand

   [req_distinguished_name]
   countryName = US
   stateOrProvinceName = Washington
   localityName = Seattle
   organizationName = DataSunrise
   organizationalUnitName = IT
   commonName = sunrise
   emailAddress = support@db.local

   [ext]
   extendedKeyUsage = 1.3.6.1.5.5.7.3.1

   [ca]
   default_ca = CA_default
   ```
Starting DataSunrise for the first time

```plaintext
dir = ./db                  # top dir
database = $dir/index      # index file.
new_certs_dir = $dir/new   # new certs dir
certificate = $dir/ca.cer   # The CA cert
serial = $dir/serial       # serial no file
private_key = $dir/private/ca.pem # CA private key
RANDFILE = $dir/private/.rand # random number file
default_days = 365         # how long to certify for
default_crl_days = 30      # how long before next CRL
default_md = sha512
policy = policy_any        # default policy
certificate = ca_default   # Certificate display option
email_in_dn = no           # Don't add the email into cert DN
name_opt = ca_default      # Subject name display option
cert_opt = ca_default      # Certificate display option
#copy_extensions = none     # Don't copy extensions from request
[policy_any]
countryName = supplied
stateOrProvinceName = optional
organizationName = optional
organizationalUnitName = optional
commonName = supplied
emailAddress = optional
```

4. Generate a root certificate ./db/ca.cer and a key ./db/private/ca.pem:

```plaintext
@ECHO OFF
SET RANDFILE=./db/private/.rand
openssl genrsa -des3 -out ./db/private/ca.pem 2048
openssl rsa -in ./db/private/ca.pem -out ./db/private/ca.pem
openssl req -new -x509 -days 3650 -key ./db/private/ca.pem -out ./db/ca.cer -config ca
openssl x509 -noout -text -in ./db/ca.cer
```

5. Generate and sign a certificate for the server or proxy:

```plaintext
@ECHO OFF
SET friendlyName=CA-signed certificate for ARMOR
SET RANDFILE=./db/private/.rand
SET /P serial=<./db/serial
openssl genrsa -des3 -out ./db/private/%serial%.pem 2048
openssl rsa -in ./db/private/%serial%.pem -out ./db/private/%serial%.pem
openssl req -new -key ./db/private/%serial%.pem -nodes -config cfg -out req
openssl ca -config cfg -extensions ext -inf files req
openssl pkcs12 -export -in ./db/new/%serial%.pem -inkey ./db/private/%serial%.pem
-name "%friendlyName%" -out ./db/private/%serial%.pfx
MOVE ./db\new\%serial%.pem .\db\new\%serial%.cer
```

6. Generated certificates will be saved in the db\new folder. Generated keys and pfx files (packed keys and certificates) will be saved in the db\private folder.

It is required that the CN (canonical name) used in the certificate, should be available at the client/proxy side and a client or proxy uses this name to connect to a server/proxy. It is required because otherwise they would not pass certificate check even if a client/proxy would determine the root certificate as trusted. You can achieve this by adding the CN to "hosts" file or by adding a corresponding entry to DNS (if administering AD).
4.6.2.4 Installing SSL Certificate for MS SQL Server Proxy
To install SSL certificate for SQL Server proxy, perform the following:

1. Run certmgr.msc (or add it via Microsoft Management Console)
2. Locate SSL certificate (Refer to subs. 4.10.2.1, step 2) (Personal / Certificates folder).
3. Export certificate with closed key to *.pfx file
4. Retrieve private key from *.pfx by executing the following command:
   ```bash
   openssl pkcs12 -in certname.pfx -nocerts -out key.pem -nodes
   ```
   **Note:** replace SERVER_HOST with actual SQL Server host name.
5. Specify the certificate and the key in GUI: Configuration —> Databases —> Certificates —> Certificate and Private Key tabs respectively.

4.6.2.5 Disabling Ephemeral Keys-Based Encryption
DataSunrise sniffer does not support processing of traffic encrypted with [EC]DHE protocol based on ephemeral keys.
To enable DataSunrise to process traffic of SQL Server 2014 or higher, it is required to disable [EC]DHE on the database server by using IIS Crypto utility: [https://www.nartac.com/Products/IISCrypto](https://www.nartac.com/Products/IISCrypto). Use the guide below.

**Note:** to disable [EC]DHE-based ciphers for SQL server's crypto provider, you can use an alternative method described here: [https://support.microsoft.com/en-us/kb/245030](https://support.microsoft.com/en-us/kb/245030)

1. Run IIS Crypto
2. Uncheck **ECDH** and **Diffie-Hellman** check boxes in the **Key Exchanges** subsection. Click **Apply**.
3. Restart DB's server for changes to take effect.
This section describes the most common issues DataSunrise users face.

1. I installed database server, client database and the firewall on one host. I’m trying to run DataSunrise in Sniffer mode, but it is not listening for the traffic.
   - In this case DataSunrise can’t capture traffic sent from host machine to that same host machine. You should use DataSunrise Proxy mode only or install database server and database client on separate hosts.

2. I’m trying to add a new Oracle database via Configuration menu, but connection is failing because of a “Couldn’t load oci.dll” error.
   - Probably you installed 32-bit version of Oracle Database Instant Client or did not set system variables correctly. You need to install 64-bit version of Oracle Database Instant Client on the server DataSunrise is installed on, and add its home directory path to the %ORACLE_HOME% system variable. Then you need to add the same directory path to the %PATH% system variable and reboot the server or restart DataSunrise service.

   For example (PATH):
   
   ```
   C:\Oracle\app\oracle\product\11.2.0\server\
   ```

3. DataSunrise running on a host can’t capture data packets between database client running on the same host and database server running on an Oracle VirtualBox virtual machine.
   - If you’re using VirtualBox 5.0.2, for instance, DataSunrise will likely fail to capture data packets between database client running on the host and database server running on the guest OS. This problem can occur under various network connection settings such as NAT, bridged and host-only. However, if you run the DB client on the guest OS and DB server — on the host, DataSunrise would be able to capture network packets.
     This issue is caused by VirtualBox’s 5.0.X virtual network adapter (VirtualBox NDIS Bridged Network Driver). Try to install an older version of VirtualBox and check if DataSunrise captures data packets between the host and the guest OS.

4. I’m trying to enter the web interface after DataSunrise was updated, but it displays "Internal System Error" message.
   - Most likely, you kept web interface tab opened in your browser while updating the firewall. Log out the web interface if necessary and press Ctrl + F5 to reload the page.

5. When I’m trying to run DataSunrise in sniffer mode, it displays a message: “Can’t to parsing SSL connection in sniffer mode”.
   - In order to run the firewall in sniffer mode, you should disable SSL support in your client application settings (SSL Mode -> Disable). You can also switch application’s SSL Mode to “Allow” or “Prefer”, but disable SSL support in database server settings first.

6. When connecting to Aurora DB, MySQL ODBC driver stops responding.
   - Most probably, you’re using ODBC driver version 5.3.6, which is known to cause freezes from time to time. Install MySQL ODBC driver version 5.3.4.

7. I forgot the password to the GUI.
   - You can set new administrator password. Use Windows CLI to run DataSunrise’s `appbackendService.exe` file with `set_admin_password` parameter (run CLI as administrator). For example:

   ```
   >appbackendService.exe set_admin_password=new_password
   ```

   To apply new password, restart `DATA_SUNRISE_SECURITY_SUITE` system service.

8. I’m using MS SQL Server database. I’m creating a target database profile, but can’t properly configure the database connection.
• In the DB connection details, specify the credentials (Default login and Password fields) used for SQL Server authentication and not for Windows authentication. To specify the database server’s host (Host field), use actual DB server’s IP address or host name instead of server’s SPN.

9. I’m using MS SQL Server database. When connecting through DataSunrise proxy, I get an error "Cannot connect to...". For example "Cannot connect to vsunrise.db.local,1435". This error disappears at the next connection.

• This error occurs when the database server and DataSunrise proxy are located on the same host and the client first connects to the server directly and then through the proxy. Client’s CSP confuses proxy and the server and tries to restore the first SSL session through the proxy connection. But the proxy and the server operate on different processes and cannot share SSL sessions.

You can disable session caching on the client side:

• Open Registry Editor. Click Start, Run, type regedt32 and click OK
• Click to select the following key in the registry:

  \[HKEY_LOCAL_MACHINE\][System][CurrentControlSet][Control][SecurityProviders]\[SCHANNEL]\  

• In the Edit menu, click Add Value, type "ClientCacheTime" in the Value Name box, select "REG_DWORD" for Data Type, and then click OK.
• Exit Registry Editor.

Having added ClientCacheTime to the registry, reboot your server.