Implementation Guide

SEP
Version 2019

ReliaSoft
This license agreement ("License Agreement") sets forth the terms and conditions that govern the distribution and use of the HBM UK Ltd software documentation (the "HBM Software Documentation"), including any and all derivative works. Any unauthorized distribution or use of the HBM Software Documentation is strictly prohibited.

Licensee hereby acknowledges and agrees to the following terms and conditions of this License Agreement for its use of the HBM Software Documentation:

HBM UK Ltd grants the Licensee who accepts and abides by the terms of this License Agreement a non-exclusive, non-transferable, royalty-free, and revocable license to the HBM Software Documentation. Unless otherwise indicated, all HBM Software Documentation are copyrighted and owned by HBM UK Ltd and are the property of HBM UK Ltd. They are licensed to you and for your use only as an individual who has purchased the software ("Licensee"). Notwithstanding this License Agreement, the Licensee shall not have a license to the trademarks, logos, or any other intellectual property of HBM UK Ltd and/or its affiliates.

Licensee may print a single copy of the HBM Software Documentation for his/her reference. Licensee may reprint the HBM Software Documentation, as needed, if the original printed copy is damaged and/or destroyed.

Except as provided above, no part of the HBM Software Documentation, either text or image, may be used for any purpose other than Licensee's own personal use and reference as a learning aid. Therefore, the reproduction, modification, creation of derivative works, storage in a retrieval system, or retransmission, in any form or by any means, electronic, mechanical or otherwise, for reasons other than Licensee's personal use, is strictly prohibited.

Certain photos and images in the HBM Software Documentation are used under non-transferable licenses obtained by HBM UK Ltd and/or its affiliates and are owned by its Licensor(s) ("Licensor"). Images depicting photos of actual persons are licensed to HBM UK Ltd and/or its affiliates and the signed model releases for these images are on file with the Licensor(s). All ownership and intellectual property rights to the HBM Software Documentation are reserved by either HBM UK Ltd and/or its affiliates or its Licensor(s).

DISCLAIMER: THE HBM SOFTWARE DOCUMENTATION IS PROVIDED “AS IS” WITHOUT WARRANTIES OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO: 1) ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT OF INTELLECTUAL PROPERTY; 2) ANY WARRANTY THAT THE HBM COURSE MATERIALS WILL CONFORM TO SPECIFICATIONS; 3) ANY WARRANTY THAT THE WORK WILL BE ERROR FREE OR VIRUS FREE. IN NO EVENT SHALL HBM UK LTD, ITS AFFILIATES, DISTRIBUTORS, CONTRACTORS, AGENTS, AND ITS LICENSOR(S) BE LIABLE FOR ANY DAMAGES, INCLUDING, BUT NOT LIMITED TO, DIRECT, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES, ARISING OUT OF, RESULTING FROM, OR IN ANY WAY CONNECTED WITH THE HBM SOFTWARE DOCUMENTATION, WHETHER OR NOT BASED UPON WARRANTY, CONTRACT, TORT, OR OTHERWISE, WHETHER OR NOT LOSS WAS SUSTAINED FROM, OR AROSE OUT OF THE RESULTS OF, OR USE OF, THE HBM SOFTWARE DOCUMENTATION. LICENSEES AGREE TO WAIVE ANY AND ALL CLAIMS AGAINST HBM UK LTD, ITS AFFILIATES, DISTRIBUTORS, CONTRACTORS, AGENTS, AND ITS LICENSOR(S), AND SHALL INDEMNIFY, DEFEND AND HOLD HARMLESS HBM UK LTD, ITSAFFILIATES, DISTRIBUTORS, CONTRACTORS, AGENTS, AND ITS LICENSOR(S) FOR ANY AND ALL LIABILITIES, CLAIMS, DEMANDS, DAMAGES, EXPENSES OR LOSSES THAT MAY ARISE FROM THE LICENSEE’S USE OR DISTRIBUTION OF THE HBM SOFTWARE DOCUMENTATION, INCLUDING ANY LIABILITIES OR DAMAGES FROM DERIVATIVE WORKS OR OTHER PRODUCTS BASED ON, OR RESULTING FROM, THE USE THEREOF.

This License Agreement is subject to change without notice and does not represent any commitment on the part of HBM UK Ltd and/or its affiliates to the Licensee, including any commitment to maintain or update the HBM Software Documentation. The names of companies, products, people, characters, and/or data mentioned in the HBM Software Documentation are not intended to represent any real individual, company, product or event, unless otherwise noted. Any rights not expressly granted herein are reserved for HBM UK Ltd.

TRADEMARKS:
ReliaSoft, Synthesis Platform, Weibull++, ALTA, DOE++, RGA, BlockSim, RENO, Lambda Predict, XFMEA, RCM++ and XFRACAS are all trademarks of HBM Prenscia Inc.
GlyphWorks, VibeSys, DesignLife and the nCode logos are all trademarks of HBM UK Ltd.

Other product names and services identified in the HBM Software Documentation are trademarks belonging to their respective trademark holders, and are used for illustration purposes. Their use in no way conveys an endorsement or other affiliation with HBM UK Ltd and/or its affiliates.
Contents

1 SEP Architecture
   1.1 Server Requirements
   1.2 Client Requirements

2 Prepare the Database Server - SQL Server or Oracle

3 Prepare the Web Server - IIS
   3.1 IIS Roles and Features
   3.2 SSL Certificate

4 Establish a Service Account for the Application (if Applicable)

5 Install the Web Portal and Activate the License

6 Create the ReliaSoft Database (if Applicable)
   6.1 Plan for Your Implementation
   6.2 Create or Upgrade the Database
   6.3 Assign Roles in SQL Server for Application Service Account

7 Update the SEP Configuration File

8 Perform Post-installation Steps (if Applicable)
   8.1 Set up User Accounts and Permissions
   8.2 Start the Synthesis Service
   8.3 IIS Application Pool Identity
   8.4 Release and Recycle Memory
   8.5 Limits for Uploads, Buffering and Requests
   8.6 HTTPS for Secure Communication
   8.7 Configure Remote ReliaSoft

9 Additional IIS Configuration Changes for Enhanced Security
   9.1 Settings
   9.2 Default web.config Changes

10 FAQs
SEP Implementation Guide

This document provides instructions to implement ReliaSoft SEP by HBM Prenscia for your organization. It covers tasks that typically require IT expertise.

After the website is functional, an application admin can enable and configure various options on the SEP Admin page. For information about these options, see the “SEP Admin Page” topic in the help for any ReliaSoft desktop application.

1  SEP Architecture

SEP is a web-based application that serves the needs of engineering teams of any size. Based on the .NET Framework, it is designed to be n-tier, scalable, distributable, robust and able to be deployed across multiple servers or on a single computer.

1.1  Server Requirements

If you plan to host the database and website on the same server, you will need:

- Windows 2008 R2 or newer
- .NET 4.6
- IIS with support for serving ASP.NET
- SQL Server 2008 or newer OR Oracle 10g or newer (32-bit and 64-bit versions of all, full version only)

1.2  Client Requirements

Once the website has been implemented, users can access it with any browser that supports the following doctype:

```html
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

This includes Internet Explorer, Chrome, Firefox or Safari residing on a Windows operating system, a Mac operating system or a tablet (such as iOS, Android, etc.). If the site is private (e.g., http://InternalServer/SEP), an application administrator may need to provide users with instructions for accessing the website on the internal network from their mobile devices (e.g., via VPN or some other method).
These are the same requirements as for the XFRACAS failure reporting, analysis and corrective action system. Both applications can be deployed together on the same database and web server(s).

2 Prepare the Database Server - SQL Server or Oracle

The ReliaSoft desktop applications, XFRACAS and SEP are all designed to connect with the same data repository on either SQL Server or Oracle.

If you need to establish a new data repository, the following considerations apply for preparing the database server. Later, you will use the admin utility to either create the database or connect to an existing database. (See Section 6 and Section 7.)

If you are setting up for SQL Server

- Make sure you have the latest version of SQL Server running. To do this, run the following query in Query Analyzer: “Select @@version”. This should return a value like “Microsoft SQL Server 2005 - 9.00.3042 (Intel X86)” or “Microsoft SQL Server 2008 R2 - 10.50.1617 (X64),” depending on which SQL Server service pack you have installed.

- Make sure you know the SQL Server Name. This is a local server name or IP address so the IIS machine with the .NET application can connect to the database. These instructions assume that you will use a default instance of SQL Server to host the ReliaSoft database (e.g., SERVERNAME). If not, you must specify the instance when you enter the server name (e.g., SERVERNAME\INSTANCENAME).

If you are setting up for Oracle

For easier support, we recommend installing the SQL Worksheet (available with the Enterprise edition) or Oracle SQL Developer (free to download from the Oracle website).

3 Prepare the Web Server - IIS

To prepare the web server prior to installing SEP, make sure the IIS Web Server role and services are installed. In addition, if applicable, prepare for SSL protection.
3.1  IIS Roles and Features

Install the Web Server (IIS) role (if it is not already installed) and make sure the following role services are also installed. (Instructions follow this table for Windows 2016, 2012 and 2008.)

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Common HTTP Features</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Default Document</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Directory Browsing</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>HTTP Errors</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Static Content</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Health and Diagnostics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>HTTP Logging</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Custom Logging</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Logging Tools</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Request Monitor</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Performance</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Static Content Compression</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Dynamic Content Compression</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Security</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Request Filtering</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>IP and Domain Restrictions</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Windows Authentication</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Application Development</td>
<td></td>
<td></td>
</tr>
<tr>
<td>.NET Extensibility</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>.NET Extensibility 4.5 or 4.6</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>Application Initialization (if Windows Server 2016)</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>ASP</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>ASP.NET</td>
<td></td>
<td>x</td>
</tr>
<tr>
<td>ASP.NET 4.5 or 4.6</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>ISAPI Extensions</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>ISAPI Filters</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Server Side Includes</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>Management Tools</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IIS Management Console</td>
<td>x</td>
<td>x</td>
</tr>
</tbody>
</table>

On Windows Server 2012 or 2016

Note that if you do not already have the required version of the .NET Framework installed, you will need to have the operating system installation media available when you install the Web Server (IIS role). The required file is in the sources/sxs folder.
1. Open the Server Manager.

2. Click the Manage menu, choose Add Roles and Features and proceed through the wizard.

   
a. If the role is already installed, expand the node, review the services that are already installed and select additional services if applicable.
   
b. If the role is not already installed, accept any prompts to install required features and proceed to the Web Server (IIS) > Role Services page where you can select the services you need to install.

4. At the end of the wizard, click Install.

On Windows Server 2008

1. Open the Server Manager.

2. If the Web Server (IIS) role is not installed, view the Roles page and, under Roles Summary, click Add Roles. Follow the wizard to install the role and services.

3. If the Web Server (IIS) role is already installed, view the Roles > Web Server (IIS) page. Under Role Services, review the services that are already installed. If you need to add service(s), click Add Role Services and follow the wizard.

3.2 SSL Certificate

If you want the website to use HTTPS for secure communication (SSL/TLS), you must have a digital certificate. Later, you will use this certificate to create the binding for the site. (See Section 8 on page 7.)

If you don’t purchase a third-party certificate, you can create your own via another method, such as by generating a self-signed certificate via IIS Manager or using the Active Directory Certificate Services role installed on the server.

4 Establish a Service Account for the Application (if Applicable)

If the database is on SQL Server, we recommend establishing a service account (e.g., “SynUser”) that SEP will use to connect to the ReliaSoft database as well as any other external databases that may be used to create “custom connection” dashboards in the Synthesis Data Warehouse (SDW). This account must meet several requirements:
It must be an Active Directory account.

Ideally, it should have a password that does not expire (recommended).

If the database in on SQL Server, a user should be assigned to a public server role, with at least the `db_datareader` and `db_datawriter` roles for the ReliaSoft database. *(If the database does not yet exist, you will need to add the roles via SQL Server after you create it in Section 6.)*

It should have at least the `db_datareader` role for any other SQL Server databases that will be used by SDW custom connections

If you plan to use this account as a database connection for Open buttons, it must also meet the requirements for an “impersonation user” for all computers that will run desktop applications—specifically, it must be on a trusted domain and it cannot be a local admin, domain admin or member of any Windows admin group (see Section 8.7 on page 12). Alternatively, you can establish a separate impersonation account for the application admin to use for Open buttons only. For details on other options to configure the database connection for Open buttons, see the “SEP Admin Page” topic in the help for any ReliaSoft desktop application.

### 5 Install the Web Portal and Activate the License

After you have prepared the database and web server(s), you can log in to the web server as an administrator and perform the following steps. User Account Control (UAC) can either be left on or turned off for this installation.

1. Run the SEP setup (e.g., `SEP19.exe`) and follow the steps in the wizard to create the website and install the activation and admin tools.

2. From **Start**, search for “SEP 2019 Activation” then run the product activation tool and follow the steps to activate your license.

The license will be registered to a specific e-mail address, which will receive the notification required to activate it. This will be the same address for all stages of license usage from development/staging to production. Choose an address that can be accessed by someone who changes the hardware on the server. If the hardware changes for any reason, the license must be reactivated to get SEP back up and running.

### 6 Create the ReliaSoft Database (if Applicable)

If you already have a database that the website will use, skip ahead to Section 7.

#### 6.1 Plan for Your Implementation

To create a new ReliaSoft database, you’ll need to run the SEP admin utility (which resides on the web server) from a Windows user account that has the following permissions:
For SQL Server implementations, you must be able to create objects under the default database owner (dbo) schema. Be prepared to specify the server and database names.

For Oracle implementations, you must be able to create a database. Be prepared to specify a port, host, service name, schema and password.

6.1.1 Created Admin Accounts

During this process, the admin utility automatically creates two new user accounts for you—one for SEP and ReliaSoft desktop applications and another for XFRACAS—that use your Windows login and provide full admin permissions for those applications. The XFRACAS account is a special, IT/admin-only account for tasks such as updating database tables, rolling out new permissions to other admin users, performing bulk data imports (so imported records are not assigned to a specific user), etc. It is invisible to regular users and does not count against the number of users allowed by your XFRACAS license.

6.1.2 Future Upgrades and Ongoing IT / Maintenance Tasks

We recommend performing upgrades and ongoing IT/admin tasks from the same Windows account that you used to create the database (which, by default, has the database permissions required for all ReliaSoft applications). If you cannot identify a single person in your organization who will be available to perform these tasks—both now and in the future—we recommend establishing a shared service account for this purpose.

Note that, even if you created the database from a personal user account, you can still create a shared account to use for future upgrades:

- For SEP and desktop applications, use the admin utility to create additional accounts that are assigned to the “admin” security group.
- For XFRACAS, use the website’s Admin tools (Admin > Configure > Security > Users) to change the domain\username of the account that was created automatically.

Instructions for upgrading SEP are provided in the Install Update guide that comes with your upgrade package.

6.2 Create or Upgrade the Database

1. Log in to Windows with an appropriate account for your implementation. (Alternatively, you can run the admin utility as that account in step 2).

2. From Start, search for “SEP 2019 Admin” and open the admin utility.

3. Click either New Enterprise Repository or Upgrade Enterprise Repository and enter the details required to create or upgrade the database.

6.3 Assign Roles in SQL Server for Application Service Account

Finally, if you created a new database on SQL Server, you must make sure the application service account (i.e., the account that the application will use to connect to the database) has the required roles assigned in SQL Server. For requirements, see Section 4 on page 4.
7  Update the SEP Configuration File

After you have installed the website, activated the license and established a database, the next step is to update the configuration file on the web server.

From Start, search for “SEP 2019 Admin” and open the admin utility. Then click Update SEP Configuration File.

1. On the Connection tab:

   - **Connection Info** - Enter the required details for the database that the application will connect to. If you used the admin utility to create the database, the connection info will appear in the fields automatically.

     **NOTE:** If you are configuring SEP for use with an Oracle database, you must specify a valid service name. Entering an SID may result in slow performance.

     Select **Encrypt Connection String** if you want to hide the connection string information within the web configuration file.

   - **User Impersonation (SQL Server)** - If the database is on SQL Server, enter the credentials that SEP will use to connect. For requirements, see Section 4.

     Select **Encrypt Impersonation Identity** if you want to hide the credentials within the web configuration file.

2. On the Settings tab:

   - **Request timeout** sets how long IIS waits for a request to the application to finish processing. Typically, this will not need to be changed for an SEP implementation.

   - For a SQL Server implementation, select **Encrypt communication** if you want to encrypt the connection between the application and the database.

     Select **Trust server certificate** if the server has a self-signed certificate.

     **NOTE:** To encrypt the connection for an Oracle implementation, you must set the encryption type to either “requested” or “required” for the Oracle database. For more information, please consult the Oracle documentation (e.g., https://docs.oracle.com/cd/B19306_01/network.102/b14268/asoconfig.htm#i1007808).

   - If a Secure Socket Layer (SSL) certificate has been implemented for SEP, select **Yes** for **HTTP Cookies Require SSL** if you also want the browser cookies to require SSL (an additional level of security).

8  Perform Post-installation Steps (if Applicable)

After installation, you may need to configure additional settings to fit your particular implementation.

To make changes to address OWASP security concerns, see Section 9.
8.1 Set up User Accounts and Permissions

After the database has been created, you can use any of the desktop applications or the ReliaSoft Admin tool to create user accounts and set access permissions. You must create an account for anyone who will be able to edit or view data in the ReliaSoft desktop applications or SEP. (User accounts for XFRACAS are managed separately.)

- In the ReliaSoft Admin tool on the web server, click the Manage Synthesis Users button.
- In the ReliaSoft desktop applications (e.g., Weibull++, XFMEA, etc.), first open the database and then choose File > Manage Repository > Users and Security.

If your organization uses Microsoft Active Directory, you can save time by importing user information from the directory to create the user accounts.

For more information, consult the “Security Options” topics in the desktop application help files (e.g., http://help.synthesisplatform.net/weibull_alta19/security_options.htm). After the accounts have been created, an application admin can use the SEP Admin page to specify which users can access SEP. (See the “SEP Admin Page” topic in the help for any ReliaSoft desktop application.)

8.2 Start the Synthesis Service

The Synthesis Service is an optional utility that is installed with SEP. If the application admin(s) want to use the service to send alerts for actions based on calendar date (e.g., when the action is due in X days), you will need to make sure the service is running on the web server. (For instructions on how to configure the settings, see the “SEP Admin Page” topic in the help for any ReliaSoft desktop application.)

To start the Synthesis Service on the SEP web server:

1. From Start, search for “Services” and open the Services window.
2. In the list of local services, right-click SynthesisService and select Properties.
3. On the Log On page, enter the credentials for an account that the service can run as. We recommend to use an account that does not expire, such as the application service account discussed in Section 4 on page 4.
4. On the General page, set the Startup type to either “Automatic” or “Automatic (Delayed Start).” Then click Start.

The current status of the service (Running, Not Running or Not Found) will be displayed on the SEP Admin page. By default, this service is configured to not run from 8 to 10 p.m. each day that can be used for routine database maintenance and backups (StopProcessingTime = 20:00:00 and MinutesToHoldProcessing = 120). To change these settings, edit serviceConfig.xml on the web server. By default, this file is installed in the “bin” folder for the SEP website (e.g., C:\inetpub\wwwroot\SEP\bin).
8.3 IIS Application Pool Identity

The application admin(s) may determine that SEP needs to display SDW dashboards based on custom connections to external Access databases (or to external SQL Server databases if the ReliaSoft database is on Oracle). If so, you may wish to set a service account (e.g., “SynUser”) as the IIS application pool identity (see Section 4). This is not recommended if your website is public.

1. In the Connections pane of the IIS Manager, click Application Pools.

2. Right-click the website’s application pool and choose Advanced Settings on the shortcut menu.

3. For the Identity property, click the ... button to open the Application Pool Identity window. Select the Custom account option and click Set to open the Set Credentials window. Enter the account credentials (domain\username) for the service account and click OK.

Note that for Access databases with the *.accdb file type, the SDW dashboard can only be displayed if the database was created with the same version of Microsoft Office (32-bit or 64-bit) that is installed on the web server (for SEP) or on the individual user's computer (for ReliaSoft desktop applications).

To ensure that the SDW dashboard will display regardless of which version of Microsoft Office is installed, use the *.mdb file type instead.

8.4 Release and Recycle Memory

For large systems or systems with a high transactional load, an “Out of Memory” error can occur when the request for pages exceeds the system’s capability to release and recycle the memory with the default IIS settings. The settings provided below will force IIS to recycle the memory usage and handle the memory usage better so that the “Out of Memory” error does not occur. Note that being too aggressive with regard to how memory is recycled can slow down the response of the system. Typically, memory is recycled when the application pool and the server are not busy. Forcing memory recycling to happen more often can take up processor cycle time when the application is still busy, thus slowing down system performance. The following settings have been tested to prevent the error occurring while making the minimum possible impact on performance.

1. In the Connections pane of the IIS Manager, click Application Pools.

2. Right-click the system’s application pool and choose Recycling on the shortcut menu.

3. In the Application Pool Recycling Settings window that appears, specify the following settings and then click Next:

   o In the Fixed Intervals area, select Regular time intervals and enter 1740.
   o In the Memory Based Maximums area, select Private memory usage and enter 1,024,000.

4. Select to log the following events and click Finish:

   o Regular time intervals
   o Private memory usage
   o Unhealthy ISAPI
8.5 Limits for Uploads, Buffering and Requests

You may need to modify some limits for uploads, buffering and requests to suit your needs for the website (e.g., if you are running very large reports and find that they do not respond, if you are unable to upload large files, etc.).

1. In the Connections pane of the IIS Manager, click the SEP site. Under Management, double-click Configuration Editor.

2. In the Section drop-down list, choose system.webServer/asp. Under limits:
   - The maxRequestEntityAllowed value sets the maximum file size that can be uploaded to the server. This is set during installation to 4,194,304 bytes (or ~4 MB).
   - The bufferingLimit value sets the size of the buffer that holds the response sent back to the client. This is set during installation to 4,194,304 bytes (or ~4 MB).

3. In the Section drop-down list, choose system.webServer/caching.
   - The maxResponseSize value sets the maximum file size that can be returned to the client. This is set during installation to 262,144 bytes (or ~262 KB).

4. In the Section drop-down list, choose system.web/httpRuntime.
   - The maxRequestLength value specifies the limit for the input stream buffering threshold, in kilobytes. This is set to 4096 KB by default.

5. Return to the Connections pane, click the SEP site. Under IIS, double-click Request Filtering.

6. In the Actions panel, click the Edit Feature Settings link.
   - The Maximum allowed content length value specifies the maximum length of content in a request, in bytes. This is set to 3000000 bytes (or ~30 MB) by default.

8.6 HTTPS for Secure Communication

8.6.1 Enable HTTPS

If you want to use HTTPS for secure communication (SSL/TLS) and you already have a certificate for the website (as discussed in Section 3 on page 2), do the following:

1. In the Connections pane of the IIS Manager, open the Sites node under the server name. Click the Default Web Site.

2. In the Actions area on the right side of the window, click the Bindings link and then click the Add button in the Site Bindings window that appears.

3. Add a site binding of type https and specify your digital certificate. Close the Site Bindings window.
4. Return to the **Connections** pane and click the **SEP** site.

5. Under IIS, double-click the **SSL Settings** icon. Select **Require SSL** and **Ignore**, then click **Apply**.

### 8.6.2 Enable TLS 1.2 Protocol for HTTPS (on Windows 2008 or 2012)

If you are using HTTPS for secure communication and you wish to enable TLS 1.2, the same protocol must be enabled for the database server, the web server and .NET on the web server. *(Also note that if you are using a digital certificate, it must be SHA-256 or higher.)*

If your web server and database server are both Windows 2016, the TLS 1.2 protocol will be enabled by default. If either server is Windows 2008 or 2012, add the following registry keys:

```
Windows Registry Editor Version 5.00

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL 2.0]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\SSL 2.0\Client]
  "DisabledByDefault"=dword:00000001

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.0]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.0\Client]
  "Enabled"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.0\Server]
  "Enabled"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.1]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.1\Client]
  "DisabledByDefault"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.1\Server]
```
"DisabledByDefault"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.2]

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.2\Client]

"DisabledByDefault"=dword:00000000

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurityProviders\SCHANNEL\Protocols\TLS 1.2\Server]

"DisabledByDefault"=dword:00000000

8.7 Configure Remote ReliaSoft

Remote ReliaSoft allows you to configure SEP to run ReliaSoft desktop applications on one or more remote servers, eliminating the need to install and update software on each client computer. This section describes the minimum requirements for setting up the server(s) and generating a connection file for Remote ReliaSoft.

8.7.1 Requirements

- A Windows server that can be configured with Microsoft Remote Desktop Services (RDS) and RDP RemoteApp, or multiple RDS servers and a “broker” that distributes the requests. When planning your hardware requirements, you can estimate the following typical memory requirements per application per session. For example, a server with 32 GB RAM could support approximately 30 simultaneous ALTA users, or 100 simultaneous Lambda Predict users, and so on.

<table>
<thead>
<tr>
<th>Application</th>
<th>Estimated Memory Requirement per Session (MB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALTA</td>
<td>1000</td>
</tr>
<tr>
<td>RGA</td>
<td>1000</td>
</tr>
<tr>
<td>Weibull++</td>
<td>900</td>
</tr>
<tr>
<td>BlockSim</td>
<td>500</td>
</tr>
<tr>
<td>RENO</td>
<td>500</td>
</tr>
<tr>
<td>XFMEA</td>
<td>400</td>
</tr>
<tr>
<td>RCM++</td>
<td>400</td>
</tr>
<tr>
<td>RBI</td>
<td>300</td>
</tr>
<tr>
<td>MPC</td>
<td>300</td>
</tr>
<tr>
<td>Lambda Predict</td>
<td>300</td>
</tr>
</tbody>
</table>

- Sufficient RDS license seats (purchased from your preferred Microsoft vendor).

- ReliaSoft desktop applications installed and kept up-to-date on the RDS server(s). Locally hosted licensing is recommended (token-based or floating). If you do not already have a locally hosted license...

8.7.2 Set Up the RDS Server(s) and Create a Connection File

Note that this document provides partial instructions focused on the settings that will affect your ability to generate a connection file that can be used within SEP. If you are not familiar with the steps required to install and configure RDS, you may need to consult documentation/support provided by Microsoft.

1. Install the ReliaSoft desktop applications and configure the locally hosted licensing.

2. In the Windows Server Manager, make sure the required RDS roles are installed:

<table>
<thead>
<tr>
<th>Windows 2016 or 2012</th>
<th>Windows 2008</th>
</tr>
</thead>
<tbody>
<tr>
<td>Remote Desktop Session Host</td>
<td>Remote Desktop Session Host</td>
</tr>
<tr>
<td>Remote Desktop Connection Broker</td>
<td>Remote Desktop Services Manager</td>
</tr>
<tr>
<td>Remote Desktop Web Access</td>
<td>Remote App Manager</td>
</tr>
</tbody>
</table>

3. In Remote Desktop Session Host, configure your Remote Desktop license (CALs) and any other settings that are appropriate for your implementation.


   o In Windows 2016 or 2012, in the Windows Server Manager under Remote Desktop Services, create a Session Collection. In the collection, use Publish RemoteApp Programs and follow the wizard. In the Properties window, make sure the **Parameters** setting is set to “Allow any command line parameters.”

   o In Windows 2008, follow the wizard in the RemoteApp Manager. Make sure the **Arguments** setting is set to “Unrestricted.”

5. Create and save an *.rdp file that contains the connection information for the ReliaSoft Launcher.

   o In Windows 2016 or 2012, open RDWeb (e.g., https://servername/RDWeb) in a web browser (other than Internet Explorer) and save the connection file.

   o In Windows 2008, click the **Create .rdp File** link in the RemoteApp Manager.

   **Note:** SEP will use the *.rdp file that you provide as a template, and it will add settings that allow the ReliaSoft Launcher to open a specific ReliaSoft application, project and analysis. If your RDS server is configured to use digital certificates, you may need to edit the file (e.g., in Notepad or another text editor) before uploading to SEP. Specifically, if a certificate was added to the end of the file, it must be removed.

8.7.3 Enable Remote ReliaSoft

Remote ReliaSoft is now ready to be enabled on the SEP Admin page. (See the “SEP Admin Page” topic in the help for any ReliaSoft desktop application.)
9 Additional IIS Configuration Changes for Enhanced Security

This section provides recommendations to address issues that may be identified if you choose to scan your web server for Open Web Application Security Project (OWASP) security concerns.

For some of the issues listed here, you will need to install the URL Rewrite tool, available at http://www.iis.net/downloads/microsoft/url-rewrite.

9.1 Settings

The tasks performed in the IIS Manager should be done at the default website level (i.e., in the Connections pane, open the Sites node under the server name and click Default Web Site). Alternatively, the model web.config code (see page 16) summarizes the changes made in the IIS Manager.

Note: If you make these changes directly in the web.config file in the root folder for your SEP website, you can skip the steps shown below in italics.

Web Server Default Welcome Page

From the wwwroot directory, remove iisstart.htm, welcome.png and the asp_client folder.

Clickjacking: X-Frame-Options Header Missing

1. In the IIS Manager Home page, double-click HTTP Response Headers.

2. In the Actions area, click Add. Enter X-Frame-Options as the name, and SAMEORIGIN as the value.

OPTIONS Method Is Enabled

1. In the IIS Manager Home page, double-click Request Filtering.

2. On the HTTP Verbs tab, click Allow Verb in the Actions area and enter Options in the Deny Verb window.

Microsoft IIS Version Disclosure

1. In the following Registry key, create a dWORD entry, DisableServerHeader, and set its value to 1:

   HKLM\SYSTEM\CurrentControlSet\Services\HTTP\Parameters

2. In the IIS Manager Home page, double-click URL Rewrite.

3. In the Actions area, click View Server Variables, then click Add and enter RESPONSE_SERVER in the text box.

4. Add an outbound rule to rewrite the RESPONSE_SERVER server variable as blank.
a. In the **Actions** area, click **Back to Rules** and then click **Add Rule(s)**.
b. In the **Add Rule(s)** window, click **Blank rule** in the **Outbound rules** category and click **OK**.

c. Create the outbound rule using the following settings:
   
   - **Name**: Response Server
   - **Precondition**: None
   - **Matching scope**: Server Variable
   - **Variable name**: RESPONSE_SERVER
   - **Variable value**: Matches the Pattern
   - **Using**: Regular Expressions
   - **Pattern**: .+
   - **Action type**: Rewrite
   - **Action Properties**:
     - **Value**: &lt;leave this field empty&gt;
     - **Replace existing server variable value**: Selected

---

**ASP .NET Version Disclosure**

1. In the IIS Manager Home page, double-click **URL Rewrite**.

2. In the **Actions** area, click **View Server Variables**, then click **Add** and enter **RESPONSE_X-ASPNET-VERSION** in the text box.

3. **Add an outbound rule to rewrite the RESPONSE_X-ASPNET-VERSION server variable as blank**.

   a. In the **Actions** area, click **Back to Rules** and then click **Add Rule(s)**.
   b. In the **Add Rule(s)** window, click **Blank rule** in the **Outbound rules** category and click **OK**.

c. Create the outbound rule using the following settings:
   
   - **Name**: x-ASPNet
   - **Precondition**: None
   - **Matching scope**: Server Variable
   - **Variable name**: RESPONSE_X-ASPNET-VERSION
   - **Variable value**: Matches the Pattern
   - **Using**: Regular Expressions
   - **Pattern**: .+
   - **Action type**: Rewrite
   - **Action Properties**:
     - **Value**: &lt;leave this field empty&gt;
     - **Replace existing server variable value**: Selected
X-Powered-By Header

1. In the IIS Manager Home page, double-click **HTTP Response Headers**.

2. Select the **X-Powered-By** header and click **Remove**.

3. In the IIS Manager Home page, double-click **URL Rewrite**.

4. In the **Actions** area, click **View Server Variables**, then click **Add** and enter **RESPONSE_X-POWERED-BY** in the text box.

5. **Add an outbound rule to rewrite the RESPONSE_X-POWERED-BY server variable as blank.**

   a. In the **Actions** area, click **Back to Rules** and then click **Add Rule(s)**.
   
   b. In the **Add Rule(s)** window, click **Blank rule** in the **Outbound rules** category and click **OK**.

   c. Create the outbound rule using the following settings:

      - **Name**: X-Powered
      - **Precondition**: None
      - **Matching scope**: Server Variable
      - **Variable name**: RESPONSE_X-POWERED-BY
      - **Variable value**: Matches the Pattern
      - **Using**: Regular Expressions
      - **Pattern**: .+
      - **Action type**: Rewrite
      - **Action Properties**:
        - **Value**: <leave this field empty>
        - **Replace existing server variable value**: Selected

Custom Errors

1. In the IIS Manager, open the Configuration Editor.

2. In the **Section** drop-down list, choose **system.web/customErrors**.

3. Set **Mode** to **RemoteOnly**.

9.2  Default web.config Changes

```xml
<configuration>
  <system.web>
    <customErrors mode="RemoteOnly"/>
  </system.web>
</configuration>
```
10  FAQs

Can we implement replication for a ReliaSoft database?

XFRACAS, SEP and the ReliaSoft desktop applications cannot be deployed with bi-directional database replication (peer-to-peer replication or merge replication). The applications are designed for use with a single back-end database; they do not handle conflict detection and resolution.

It may be possible to use a ReliaSoft database with uni-directional replication (transactional replication or snapshot replication). However, this is likely to affect the performance of the application(s) and you must test on your own to evaluate the impact in your particular situation. **This type of use is not recommended or supported by ReliaSoft.**
For the purpose of disaster recovery, we recommend to establish a regular schedule for database backups and transaction log backups. These backups can be stored in a location that is protected from potential failure of the application's database server. If an issue occurs, you can restore the most recent database backup (e.g., nightly) and then restore subsequent transaction logs up to the point right before the failure.