# Purpose

This document briefly outlines the steps to install the latest build of the Sentinel Serialization services and client application.

# Scope

This build is the latest work of the development team. Serialization operations are complete. SQL Prelog and Serialization tables have been migrated into the FA database Custom schema. Initial work to handle Leads submitted via Prelog has been added to the table designs. The automatic Lead handling service is operational. The Serialization and Lead client application functions primarily as a status viewer.

Serialization services are provided as Windows Service-hosted WCF services. To facilitate testing with minimum installation, we have provided console hosts applications for all services. The Sentinel mock services are provided only in a console-hosted form.

Since components remain under development, installation using the Windows installer had not been completed. Rather, components requiring installation must be installed using manual processes as noted in the following section.

# Installation

The Serialization feature is installed from a single MSI installation file. Advanced options provide for selecting components for installation. This includes selection of the Serialization services, console hosts for testing, mock services and data, installation of SQL scripts, Serialization and administration client applications. The installer can install individual components or all components at once.

## Service Installation

If Serialization has previously been installed on the target server, stop the service and uninstall it using the Windows Control Panel > Programs and Features applet.

The single, installer file FAS.Sentinel.Serialization.Setup.msi contains all of the Serialization components. Perform the installation with administrator privilege.

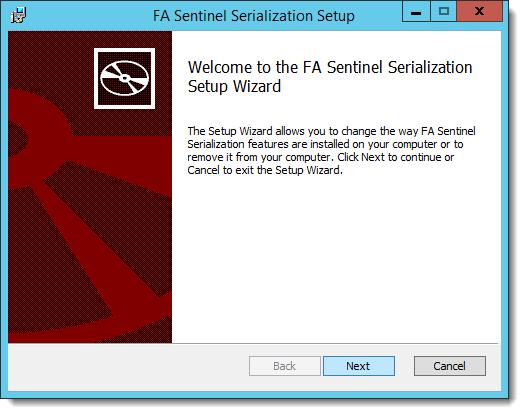


Figure 1 - Serialization Feature Installation

After extracting the installer, the setup requests that you acknowledge the TCSC license agreement.

After reading the agreement, check the box indicating your acceptance and click the Advanced button to permit changing the installation folder.

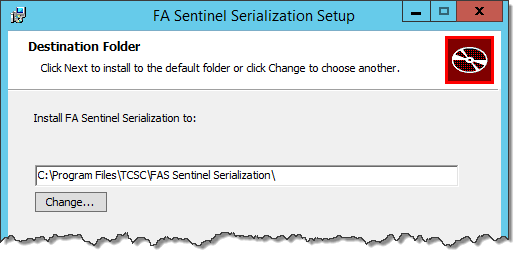


Figure 2 - Select Installation Location

The Serialization services run as 64-bit services. It is customary to install the service under the “Program Files” folder. Click Next to select the installation components.

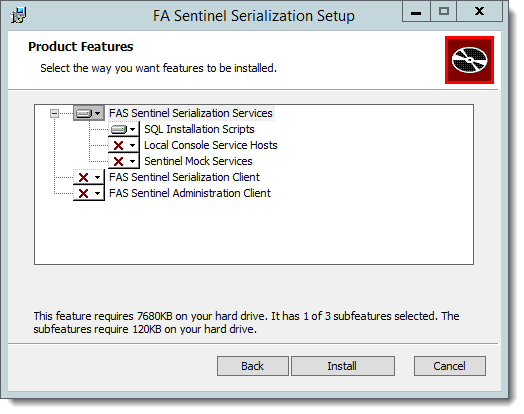


Figure 3 - Installation Feature Selection

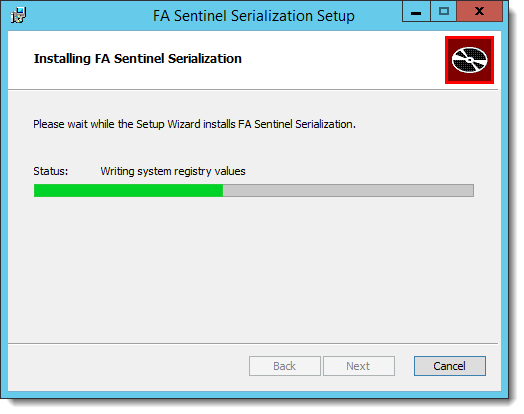


Figure 4 ‑ FAS Serialization Installation Progress

When the installation process completes, if problems occurred during the installation, preserve the Windows Installer log file to assist problem determination.

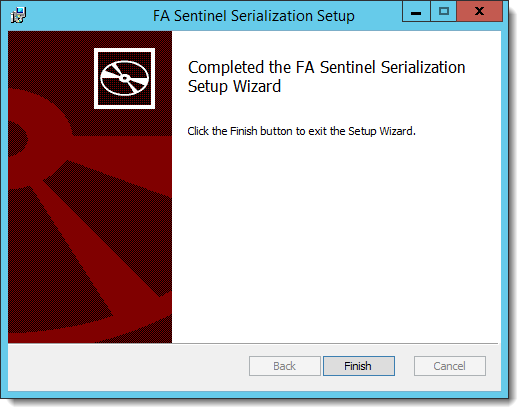


Figure 5 - FAS Serialization Installation Finalization

DO NOT START THE SERVICE until you have completed the remaining configuration steps.

Subsequent installations may add or remove selected components. Using the Windows Control Panel Programs and Features applet, the FAS Sentinel Serialization feature can be removed, repaired or individual components added or removed.

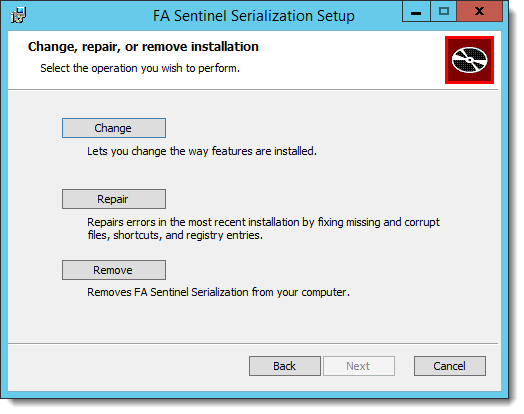


Figure 6 - Change, Repair or Remove Feature

## Installing Components

Serialization operates as a collection of Windows services under the Trusted Subsystem security model. It runs in the same security context as the other FAS application services.

The installation process requires carefully following these steps:

### Using local console hosts

1. Install FAS.Sentinel.Serialization.Setup.msi according to the Service Installation instructions below.
2. Modify the specified SQL installation scripts according to local infrastructure.
3. Use the SQLCMD utility to run the SQL installation script FAS.Sentinel.Serialization\_CommandScript.sql.
4. Configure each console host configuration file according to the local infrastructure.
5. Start each of the console service hosts.
6. Start the Sentinel Mock services console host.

### Using Windows service hosts

1. Install FAS.Sentinel.Serialization.Setup.msi according to the Service Installation instructions below.
2. Use the SQLCMD utility to run the SQL installation script FAS.Sentinel.Serialization\_CommandScript.sql.
3. Configure each service configuration file according to the local infrastructure.
4. Using the sc.exe utility, install and configure each service and configure its Log On and Startup type.
5. Start FAS Sentinel Configuration, Serialization Manager and Request Queue Services.
6. Start the Sentinel Mock services console host.

Failure to complete the above steps may lead to unpredictable and likely undesirable results.

The SQL script Sentinel\_CommandScript.sql is installed in the SQL\_Installation folder under the service folder (the default install location is %ProgramFiles%\TCSC\FAS Sentinel Serialization Services). This script invokes a number of SQL scripts to create tables in the ‘custom’ schema and adds several stored procedures in the target database.

The MSI installer installs the Serialization services but does not register them with the Windows Service Control Manager (SCM).

# SQL Database Provisioning

Installing stored procedures into the FA database requires that the user is logged in with administrator permission for the target database. During MSI installation, if you selected the “SQL Installation Scripts”, the installer copied the SQL installation scripts under the service installation folder (the default install location is %ProgramFiles%\TCSC\FAS Sentinel Serialization Services\SQL\_Installation).

## Modify SQL Installation Scripts

Modify the USE statement in Sentinel\_CommandScript.sql to specify the name of the FA Core database that is to support operation of the Serialization services.

Modify the SQL scripts Sentinel\_RepositoryObjectContentChunk\_Synonym.sql and Sentinel\_RepositoryObjectVersion\_Synonym.sql, replacing [yourORdatabase] with the name of the FA Object Repository database that is to support operation of the Serialization services.

## Provision FAS.Serialization Tables

After modifying the command script, change to the SQL\_Installation\Script\_Components folder and run the following command to add the Serialization tables to the FA Core database.

sqlcmd –E -S <Server> -i ..\Sentinel\_CommandScript.sql -m10 -r1

Sentinel\_CommandScript.sql is the name of the SQLCMD script that invokes other SQL scripts in the folder in which it is run. These scripts install views, synonyms and stored procedures into the Custom database schema.

When the SQLCMD utility runs, it displays its progress to the console.

**Note:** the installation script Sentinel\_CommandScript.sql is not a SQL query. It contains instructions for the SQLCMD utility to run individual SQL scripts in the SQL\_Installation\Script\_Components folder. The SQLCMD utility must be run in the folder containing the SQL scripts that were installed by the Serialization\_Setup process. These are idempotent scripts that and not remove existing data.

## Load Mock Data

To support testing of the client and services when actual data is not readily available, the installation process optionally provides mock data in the SQL\_Installation\Script\_Components\MockData folder. This script is not idempotent and simply adds a set of sample data in support of the Serialization services.

# Service Configuration

After Serialization software is installed, the service must be configured to the local infrastructure. For initial testing, the installer (optionally) provides console host applications to run the services without requiring that they be installed and enabled in the Windows Service Control Manager.

## Service Security

The FA Serialization services operate in compliance with the Trusted Subsystem security model. As such, they must run in the security context of the FA Services service account. If using the console hosts option, starting the console host in the administrator security context usually is sufficient.

Using the Windows service hosting option requires that you use the Services Management Console. Open the Properties dialog for each of the services. On the General tab, ensure that “Startup type” is “Automatic”. On the Log On tab, select “This account” and configure the FA Services account name and password. DO NOT START THE SERVICE until you have updated the application configuration files.

## Service Configuration Files

Ensure that each service configuration file contains settings compatible with the target environment. If using the console host option, configure the appropriate Console.exe.config file. If using the Windows service host option, configure the appropriate WindowsService.exe.config file.

|  |  |
| --- | --- |
| File name | Setting |
| FAS.Configuration.Console.exe.config | connectionStrings |
|  | TCSCLogging logPath |
| FAS.Sentinel.Serialization.Console.exe.config | connectionStrings |
|  | TCSCLogging logPath |
|  | LabReportPacketTypeId |
|  | ServiceSecurityAccountName |
| FAS.Sentinel.Serialization.RequestQueueConsole.exe.config | connectionStrings |
|  | TCSCLogging logPath |
|  | LabReportPacketTypeId |
|  | ServiceSecurityAccountName |
| FAS.Sentinel.Serialization.MockServices.Console.exe.config | TCSCLogging logPath |

Review the following settings and make the appropriate changes.

### TCSCLogging logPath

Serialization records its actions in a log file to assist with problem determination. The configuration file specifies the log file location, the amount and type of information recorded, and the maximum size of each log file segment. Serialization obtains configuration settings from the FAS.Serialization.Sentinel.WindowsService.exe.config file in the configuration/TCSCLogging element.

1. Specify the location of the log files as the value of the logPath attribute.
2. Set the log file name with the logName attribute
3. Control the type and volume of logging detail using the level attribute. The level set during installation is “Verbose”. This setting generates the greatest, most detailed amount of logged data. Upon completion of testing, the recommended operational level is “Error.”

By default, each service’s log file is configured to output to the C:\Logs folder.

### connectionStrings

The serialization services noted need to connect to the FA Core database. Inspect each configuration file and locate the configuration/connectionStrings element. Change the SERVER and DATABASE properties of the ConfigurationDbContext, and FaContext settings to correspond to the local installation. Your changes should match the following pattern:

<connectionStrings>  
 <add name="FaContext"  
 connectionString="Data Source=**<SERVER>**;Initial Catalog=**<DATABASE>**;Integrated Security=True"  
 providerName="System.Data.SqlClient" />  
 . . .  
</connectionStrings>

### General Application Settings

The following settings are found in the configuration/appSettings element and configure the FA edge of the Serialization interface.

LabReportPacketTypeId value="<Guid>" /> This setting is the unique identifier of the lab report Packet Type. The following SQL script extracts the value from the FA Core database.

SELECT [Id]

,[Name]

,[Description]

FROM .[pkt].[PacketType] PT

WHERE PT.Name = 'Lab Report'

ServiceSecurityAccountName value="FBI UID" This setting is the default value to be used as the UsernameToken when a valid FBI UID is not provided in the Position Code field of the FA employee details.

## Service Installation

The three (3) serialization services can be installed using the Windows SC utility. Note that the named options have a space following the equal sign “=”.

The SC utility creates a subkey and entries for a service in the registry and in the Service Control Manager database. The command syntax is:

sc [<ServerName>] create <ServiceName> binpath= <BinaryPathName> [obj= <AccountName>]

Where

<ServerName> Specifies the name of the remote server on which the service is located. The name must use the Universal Naming Convention (UNC) format (e.g. \\MyServer). If running SC locally, you can omit this parameter.

<ServiceName> This setting specifies the “short name” of the service as may be seen in response to a net start command.

obj= <AccountName>.Specifies the name of the FA Service Account under which the service will run.

password= <Password> This is the password for the FA Service Account.

The following are examples

sc create FAS\_Configuration binpath= "C:\Program Files\TCSC\FAS Sentinel Serialization Services\FAS.Configuration.WindowsService.exe" obj= <FA\_ServiceAccount> DisplayName= "FAS Sentinel Configuration Service" password= <FA service password>

sc \\FASAppServer create FAS\_RequestQueue binpath= "C:\Program Files\TCSC\FAS Sentinel Serialization Services\FAS.Sentinel.Serialization.RequestQueueService.exe" obj= <FA\_ServiceAccount> DisplayName= "FAS Sentinel Request Queue Service" password= <FA service password>

C:\>sc create FAS\_Serialization binpath= "C:\Program Files\TCSC\FAS Sentinel Serialization Services\FAS.Sentinel.Serialization.WindowsService.exe" obj= <FA\_ServiceAccount> DisplayName= "FAS Sentinel Serialization Service" password= <FA service password>

The Sentinel Mock service is not configured to run as an installed Windows service. It is intended to run under the console host found in the Mock Services folder.

# Application Folder Structure

The FAS Sentinel Serialization Setup MSI creates a folder structure under the Program Files folder according to the selections made during installation. The installer places the files required to operate all services and their Windows and console hosts in the FAS Sentinel Serialization Services folder. Under this folder, the installer places the SQL\_Installation files and the Mock data source files.

Selecting the Serialization Client during installation places the files required to run the client in the FAS Sentinel Serialization folder. The installer also creates the registry entries required to install the controls and libraries used in the user interface. The Serialization Client may be tested on a machine other than the one running the services.

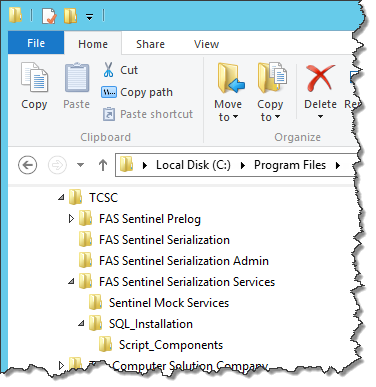


Figure 7 - Installed Folder Structure

# Ports in use

The four (4) serialization services are configured to listen on the following ports. These port assignments may be changed to meet local requirements.

| Scheme | Port | Service |
| --- | --- | --- |
| netTCP | 8755 | Configuration service |
| netTCP | 8001 | Sentinel.SerializationManagement service |
| https | 8733 | Sentinel.Mock Case service |
| https | 8733 | Sentinel.Mock Lead service |
| https | 8733 | Sentinel.Mock Serialize service |
| https | 8733 | Sentinel.Mock Search service |
| https | 8733 | Sentinel.Mock CreateFD1057 service |

The Request Queue service client endpoints must be configured to use the same addresses exposed by the Mock services. In addition, the Request Queue service and Mock services must be configured to use the SentinelWSSecurityBinding custom binding that supports the Data Power WS-Security UsernameToken authentication mode.

# Running the application

The FA – Sentinel Serialization services communicate among themselves as illustrated in Figure 8 - Serialization Service Communication.

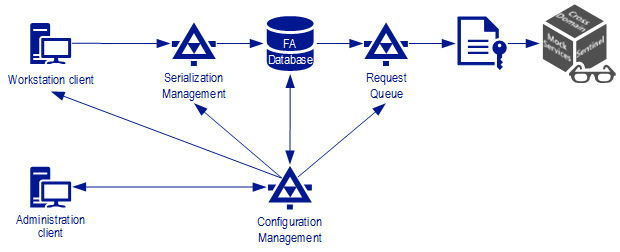


Figure 8 - Serialization Service Communication

Although the services are resilient to communication failures, it is best to start the services in order to minimize such failures.

Start the console hosts in the following order:

1. Mock Services
2. Configuration
3. Serialization
4. Request Queue

Following startup of the service command hosts you can start the Serialization client application.

During startup the Serialization client application requests the FA Employee entry for the currently logged in user. If the FAS Serialization service cannot be accessed or FA Employee entry cannot be accessed or if the Windows user name is not in the table, the Serialization client raises a message and shuts down.

The FAS.Sentinel.Serialization.RequestQueueService will start properly but fail when attempting to communicate with the Sentinel Mock service if the Mock Services have not been installed with the proper SSL certificates. Test the Sentinel Mock services using a normal browser before attempting to run the Request Queue service.