SafeNet Authentication Service

PCE/SPE Installation Guide

gemalto
security to be free

Product Version: SafeNet Authentication Service 3.5 PCE/SPE

Document Part Number: 007-012949-002, Rev. A

Release Date: August 2016

Contents

Preface

Applicability

The information in this document applies to:

- **SafeNet Authentication Service (SAS) – Service Provider Edition (SAS-SPE)**
  The software used to build a SafeNet authentication service.
- **SafeNet Authentication Service (SAS) – Private Cloud Edition (SAS-PCE)**
  A term used to describe the on-premises implementation of SAS-SPE.

Support Contacts

If you encounter a problem while installing, registering or operating this product, please make sure that you have read the documentation. If you cannot resolve the issue, contact your supplier or Gemalto Customer Support. Gemalto Customer Support operates 24 hours a day, 7 days a week. Your level of access to this service is governed by the support plan arrangements made between Gemalto and your organization. Please consult the support plan for further information about your entitlements, including the hours when telephone support is available to you.

Contact Method	Contact Information	
Address	Gemalto, Inc. 4690 Millennium Drive Belcamp, Maryland 21017, USA	
Phone	US	1-800-545-6608
	International	1-410-931-7520
Technical Support Customer Portal	https://serviceportal.safenet-inc.com Existing customers with a Technical Support Customer Portal account can login to manage incidents, get latest software upgrades, and access the Gemalto Knowledge Base.	

System Requirements

For details, see the *SafeNet Authentication Service System Requirements Guide*.

Important Notes

Auto Provisioning Service

NOTE: SAS 3.4 PCE/SPE and later does not support the running of multiple Auto Provisioning Services. Only one service can be run for any given SAS installation.

The Auto Provisioning Service processes all provisioning rules, including:

- Token Provisioning Rules
- Operator Role Provisioning Rules
- Account Manager Role Provisioning Rules

The Auto Provisioning Service manages the creation of provisioning tasks and revocation of previously assigned tokens.

RADIUS Authentication

Information about using SAS with RADIUS authentication is provided in the following documents:

- SAS Agent for FreeRADIUS Configuration Guide
- SAS FreeRADIUS Updater Configuration Guide
- SAS Agent for NPS Configuration Guide

Available at: http://www2.safenet-inc.com/sas/implementation-guides.html

SAS does not Work on FIPS Mode Enabled Machines

SafeNet Authentication Service does not work correctly on FIPS mode enabled machines. Disable '**System Cryptography: Use FIPS compliant algorithms for encryption, hashing and signing**' on the SAS server.

MS SQL in Mixed Mode

Only MS SQL users can be used by SAS to connect to MS SQL databases. Also, mixed mode is required for MS SQL.

MS SQL Collation

It may be possible that the required collation (*SQL_Latin1_General_CP1_CI_AS*) is not applied to the MS SQL database due to default server settings. If any other collation is applied, it needs to be changed.

To check the applied collation, run the following query:
SELECT CONVERT (varchar, DATABASEPROPERTYEX('dbname','collation'));

Note that all the services must be stopped, before executing the following query to alter the collation:

USE master;
GO
ALTER DATABASE <dbname>
COLLATE SQL_Latin1_General_CP1_CI_AS ;
GO

Recommendation

SSL is disabled and TLS is enforced for HTTPS connections to all web servers. This change will not impact most customers.

Customers who use custom applications accessing the Management API, the GrIDSure API, or the Token Validator API directly (not using the Java or .NET Agents) will experience difficulties if their custom applications use SSL only. In these cases, it is recommended to verify that these custom applications behave according to standard practices, and have no issues using TLS when this is enforced by the server.

1

Installing SafeNet Authentication Service

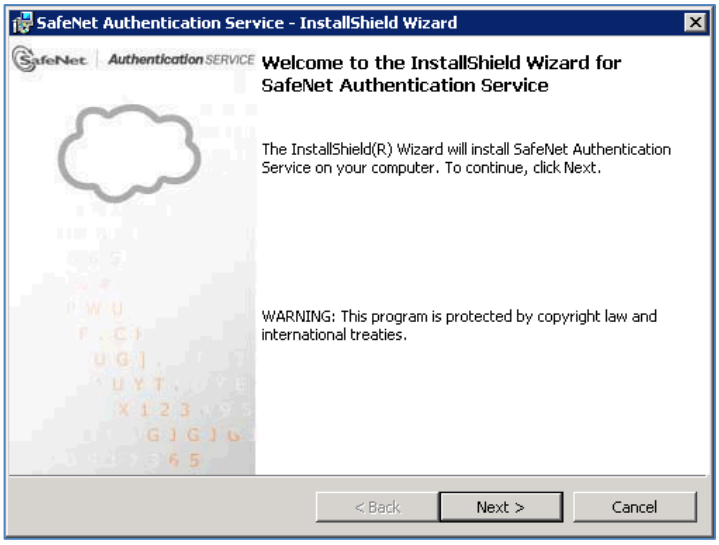
Installing SafeNet Authentication Service on Windows

Prerequisite Microsoft Server Components:

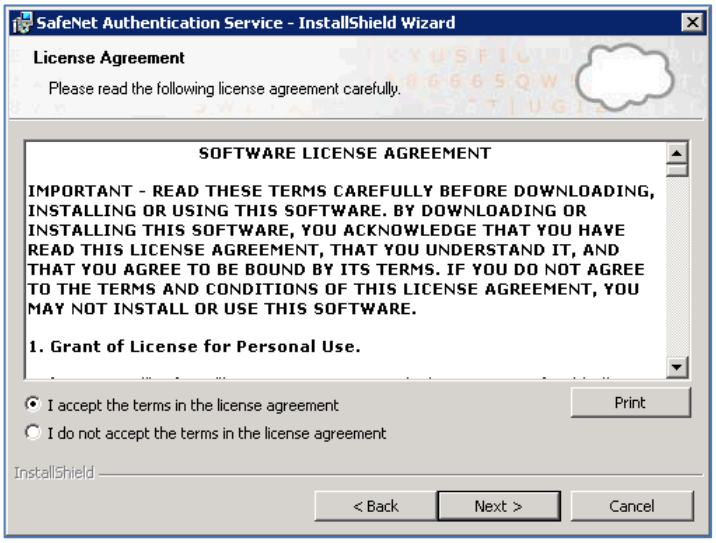
- IIS 7.x/8.x with ASP.NET 3.5

To install SAS on Windows:

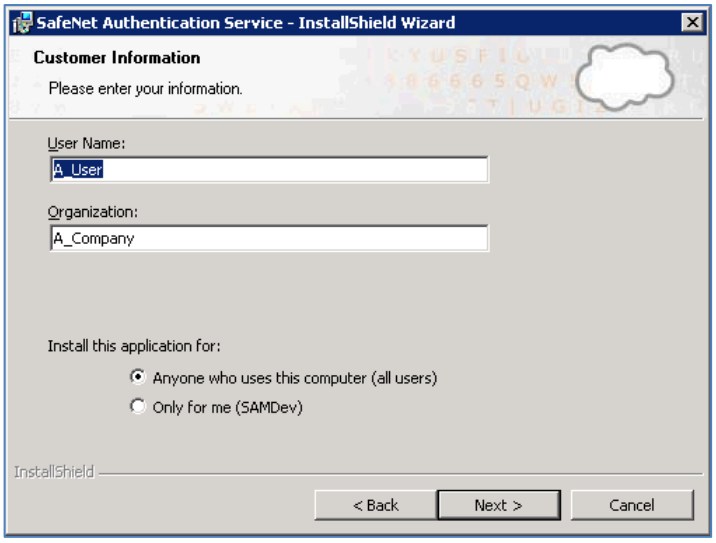
1. Log on to the server on which SAS will be installed using local or domain administrator credentials.
2. Locate and run the following SAS installer, which is located in the directory of the SAS distribution package: BlackShield ID Service Provider Edition x64.exe
3. On the **Welcome to the InstallShield Wizard for SAS** window click **Next**.



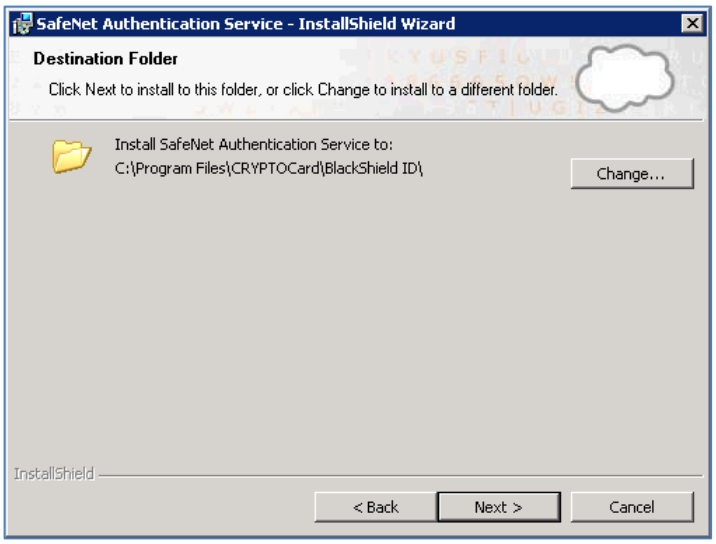
4. On the **License Agreement** window, select **I accept the terms in the license agreement,** and then click **Next**.



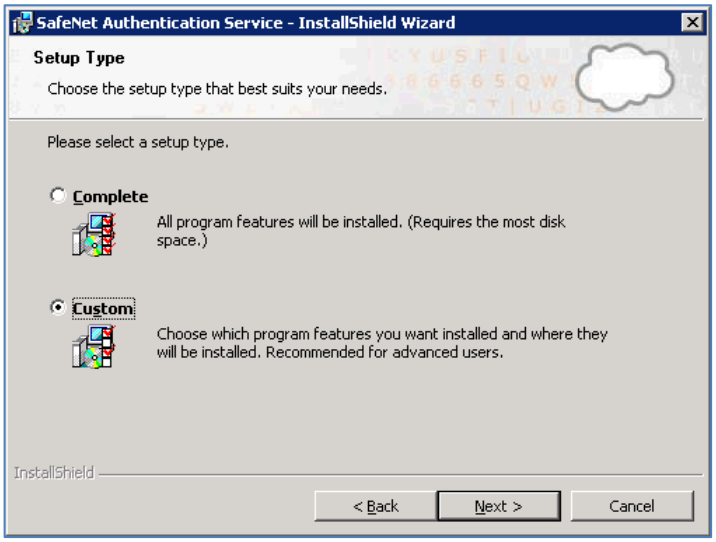
5. On the **Customer Information** window, do the following:
   a. Enter the **User Name** and **Organization**.
   b. Select one of the following options:
      - **Anyone who uses this computer** – Enable access for all users.
      - **Only for me** – Enable access only for the user performing the installation.
   c. Click **Next**.



6. On the **Destination Folder** window, perform one of the following steps:
   - To accept the default installation folder, click **Next**.
   - To change the installation folder, click **Change**, and then browse to locate and select the applicable folder.

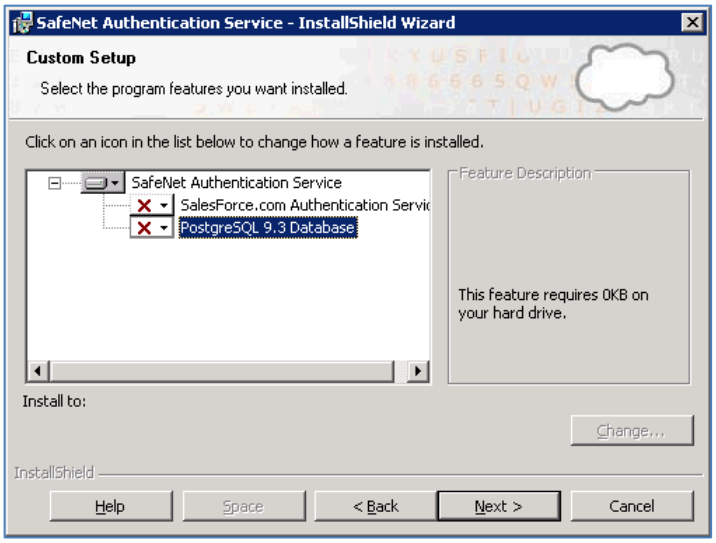


7. On the **Setup Type** window, select **Custom**:

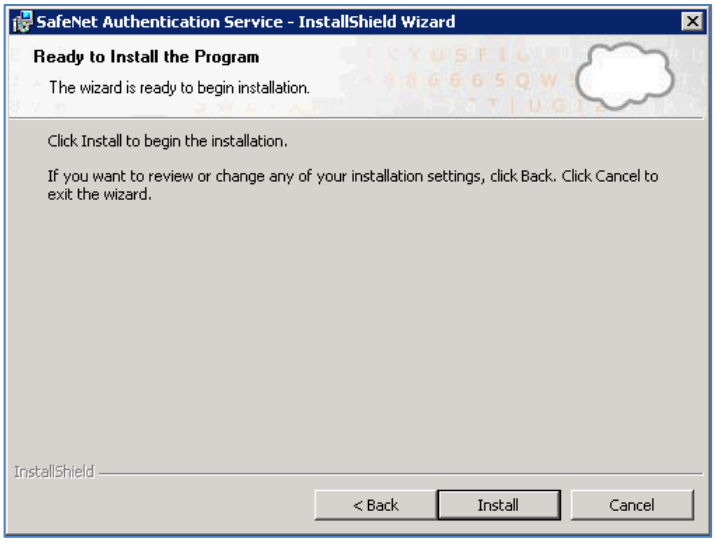


8. If using a database other than PostgreSQL, right click the **PostgreSQL 9.3 Database**, select **This feature will not be available** and then click **Next** to return to the **Setup Type** window.

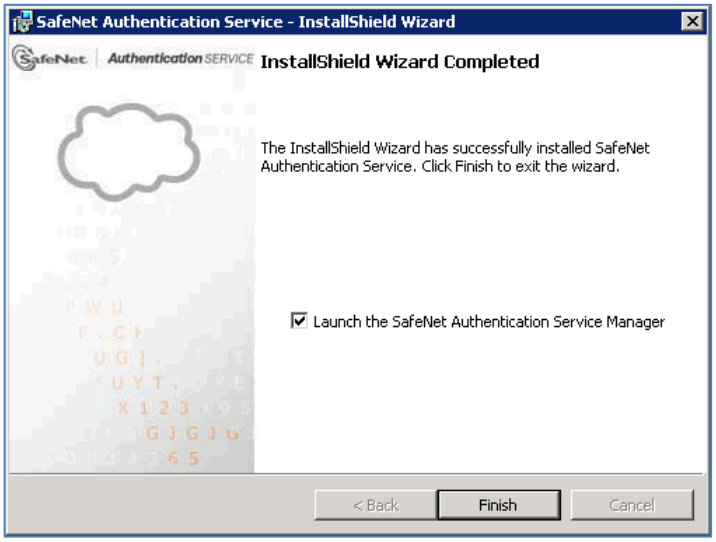
> **NOTE:** We recommend using PostgreSQL 9.3 for test installations or proof-of-concept (POC) installations only.



9. On the **Setup Type** window, click **Next** to proceed.
10. On the **Ready to Install the Program** window, click **Install**.



11. When the installation is complete, click **Finish** to exit the wizard



2
Preparing a Database

SafeNet Authentication Service supports the following databases:

- PostgreSQL
- MS SQL
- MySQL

Preparing the PostgreSQL Database

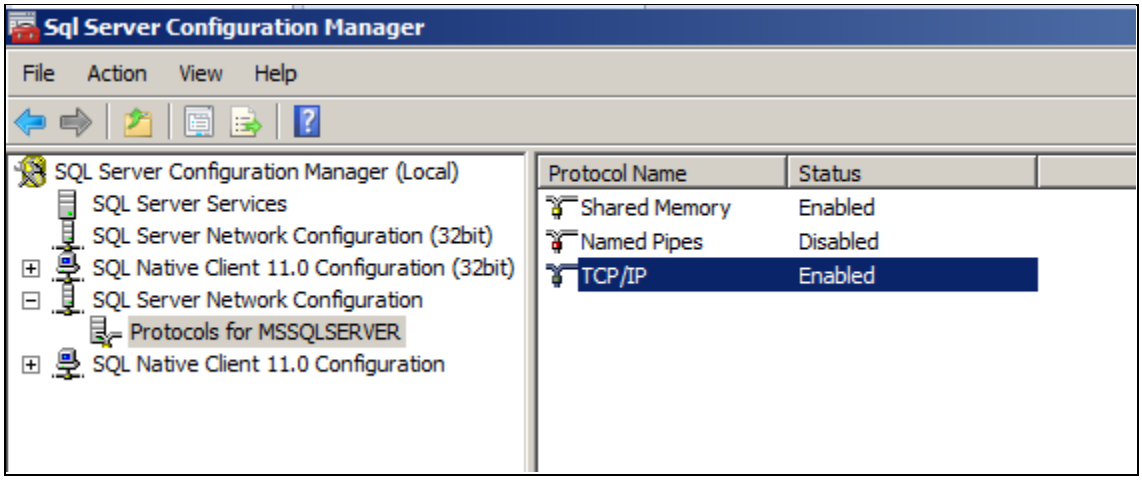
PostgreSQL should be used only for test and proof-of-concept installations. Is not supported in HA configurations. For details, refer to “Step 1 – Configure a Database” on page 27.

NOTE: Use default password to connect to the PostgreSQL database.

Preparing the MS SQL Database

For replication, an active/active (multi-master) configuration needs to be deployed. On MS SQL, this is transactional peer-to-peer replication. For details, refer to “Step 1 – Configure a Database” on page 27. Also, refer to Microsoft documentation.

NOTE: The MS SQL database needs to have TCP/IP enabled, and SQL port set to 1433.



NOTE: The MS SQL database is recommended to be used with the Latin type collation.

Setting Up the MySQL Database

You can configure MySQL database with or without high availability. The detailed information on setting up MySQL database with high availability is given below.

Achieving high availability on database level is essential for maintaining application availability. Databases are the center of today's enterprise and web applications. Just minutes of downtime can often result in significant amounts of revenue loss and unsatisfied customers. Making database highly available is therefore a top priority for all organizations.

MySQL is used with many applications demanding availability and scalability. *Availability* refers to the ability to cope with, and if necessary recover from, failures on the host, including failures of MySQL, the operating system, or the hardware.

High Availability Solution

The MySQL high availability solution supported with SafeNet Authentication Service is MySQL Master-Slave Replication. MySQL Replication is the most popular and cost-effective high availability solution.

MySQL Replication

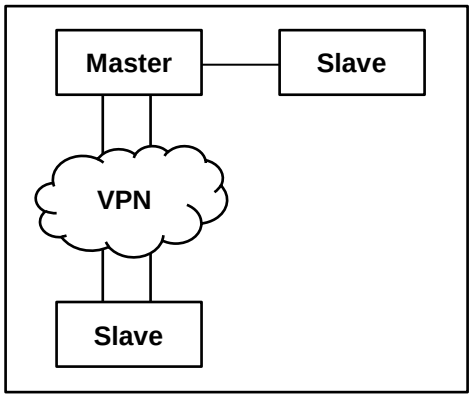
Replication enables data from one MySQL database server (the master) to be copied to one or more MySQL database servers (the slaves). The supported replication mode is *Asynchronous*; slaves do not need to be connected permanently to receive updates from the master. The process of replication is not immediate and there might be some delay due to network latency.

A server involved in a replication setup has one of following roles:

- **Master**: Master MySQL server writes all transactions that change data to a binary log
- **Slave**: Slave MySQL server connects to a master (on start), downloads the transactions from the master's binary log, and applies them to the local server

Binary logs are files that contain details of every transaction that the MySQL server has executed. Slaves contact their master to retrieve newer bits of the binary log, and apply the changes to their local database.

Consider a master-slave setup where a master is connected with one slave from the local network and one slave via a VPN over the Internet.



A setup such as this will result in the two slaves having slightly different data. The locally attached slave may be more up to date, because of added latency and bandwidth restrictions over the VPN connection.

SAS HA Controller Service

SAS HA Controller Service is responsible for setting up and managing MySQL Replication. It configures MySQL servers in master-slave mode. It also makes sure database is highly available to SAS.

If the master MySQL server is not accessible to SAS, after trying 5 times to connect to the master MySQL server the SAS HA Controller Service promotes an appropriate slave MySQL server as a new master.

Deployment Scenario

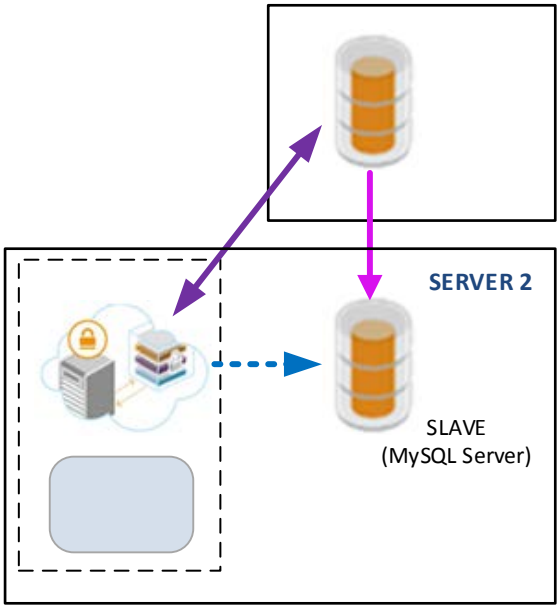
NOTE: On the server where **SAS HA Controller Service** is active, you need to set the firewall rule in a way such that other SAS instances can access it.

In both the small-sized and medium-sized deployment scenarios illustrated below, notice the following:

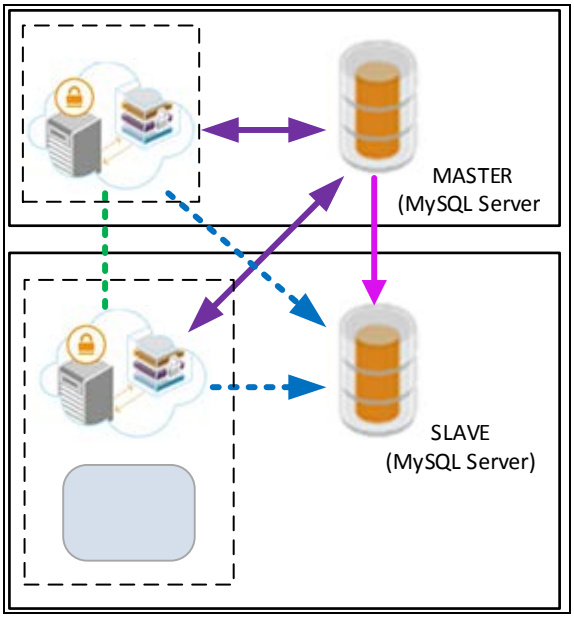
- All SAS instances on different servers interact only with the master database to perform authentication.
- All the update in the master database is replicated to all the slave databases.
- All SAS instances on different servers are aware about all the slave databases. When any of the slave databases is promoted as a master database, all SAS instances now interact with the new master database to perform authentication.
- All SAS instances without the HA Controller Service are connected to the SAS instance with the HA Controller Service. This is required for many purposes. For example, if there is a new master database, all SAS instances now must be informed about it so that they can perform authentication. The intimation about the new master database is done by the HA Controller Service.

Small-sized Deployments

Scenario 1: Server 1 (Master MySQL DB), Server 2 (SAS, HA Service, and Slave MySQL DB)

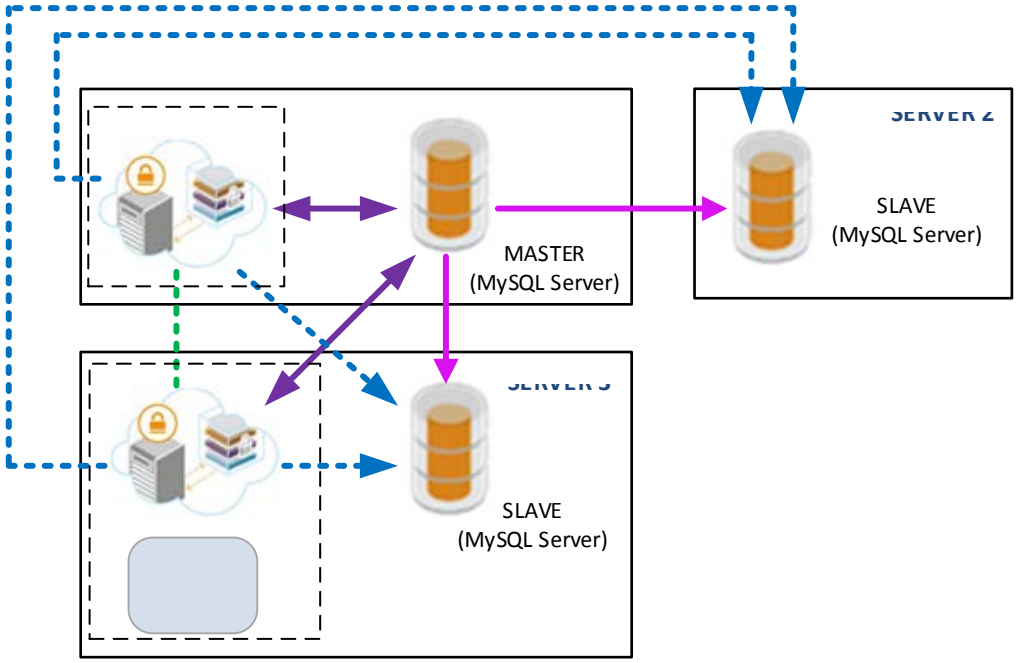


Scenario 2: Server 1 (SAS and Master MySQL DB),
Server 2 (SAS, HA Service, and Slave MySQL DB)

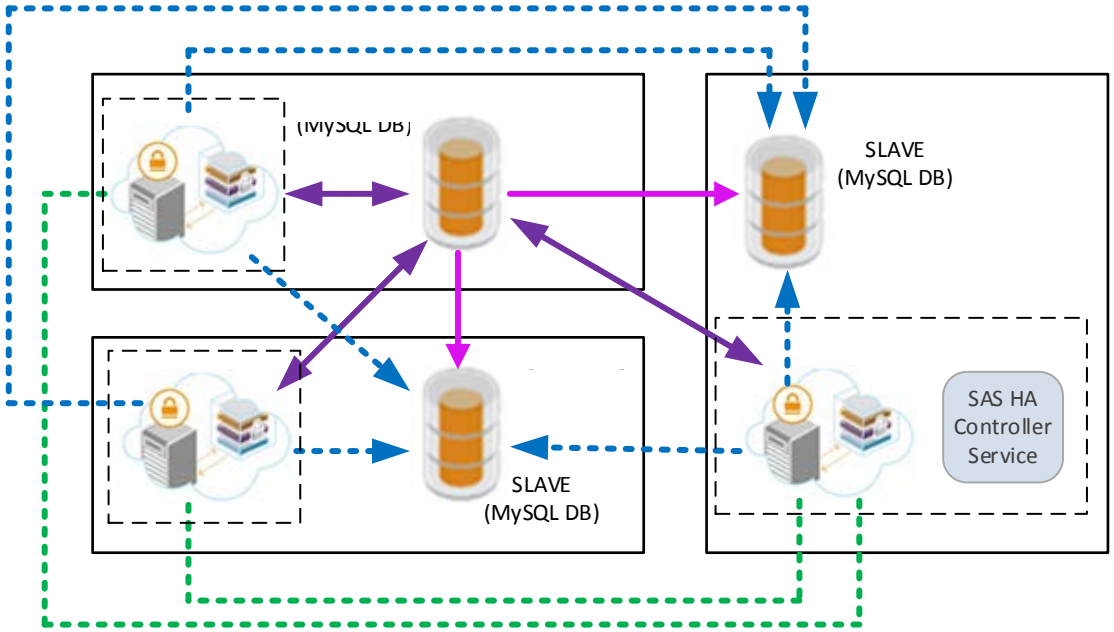


Medium-sized Deployments

Scenario 1: Server 1 (SAS and Master MySQL DB),
Server 2 (Slave MySQL DB)
Server 3 (SAS, HA Service, and Slave MySQL DB)



Scenario 2: Server 1 (SAS and Master MySQL DB),
Server 2 (SAS, HA Service, and Slave MySQL DB)
Server 3 (SAS and Slave MySQL DB)



NOTE: In both small-sized and medium-sized deployment scenarios, ensure that the **SAS HA Controller Service** is not hosted on the same server as master DB. This is due to the fact that when the server hosting master DB goes down, the HA server should be up and running on the other machine so that it can act in time

and initiate to promote a slave as the new master. In case of two slaves, the slave with the most recent updates will be promoted as the new master.

Setting up Highly Available MySQL Database

NOTE: If a setup is running a master-slave configuration and a new slave needs to be entered later, the whole setup can be done again.

Preparing MySQL Servers

It is assumed that one master and two slave MySQL database servers will be on separate machines, and the MySQL database server is installed on each of these machines.

On each of the MySQL server, perform the following steps:

1. From the **Windows Services** application, stop the MySQL service.
2. From **%ProgramData%\MySQL\<MySQL Server 5.7>**, open the **my.ini** file and add the below lines:

```
#########   START   ##########
binlog-format=ROW
log-slave-updates=true
enforce-gtid-consistency
gtid-mode=on
#disable-gtid-unsafe-statements=true # Use enforce-gtid-consistency from 5.6.9+
master-info-repository=TABLE
relay-log-info-repository=TABLE
sync-master-info=1
#datadir=/home/billy/mysql/data1
server-id=100
log-bin=utilBINLog-bin.log
relay-log=RELAYLog.log
report-host=10.164.44.246
binlog-do-db= BlackShield
replicate-do-db= BlackShield
##########   End ##########
```

As described below, specify the values for parameters in **bold** above:

server-id: It is already specified in the my.ini file. You need to specify your own server IDs. Each server in replication must have a unique server ID.

log-bin: Name of a file that will be used as a log file.

relay-log: Name of the file that will be used as a relay log file.

report-host: IP address of the machine where the **my.ini** file exists.

binlog-do-db: Name of the database for which bin log is to be created.

replicate-do-db: Name of the database that is to be replicated (here, **replicate-do-db** and **binlog-do-db** are same).

3. From **%ProgramData%\MySQL\<MySQL Server 5.7>\Data**, delete the **auto.cnf** file. This will result in generating a unique GTID for the MySQL server.

   All MySQL servers in replication must have a unique GTID.

4. From the **Windows Services** application, start the MySQL service.

5. Ensure that the *SAS database user* is created on all the MySQL servers. If not created yet, run the following SQL commands on all the MySQL servers.

   CREATE USER 'SAS DB User'@'IP address of the SAS server' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'SAS DB User'@'IP address of the SAS server';

6. Now you need to create a *replication user* on all the MySQL database servers. This *replication user* will be responsible for communication between master and slave MySQL database servers. Note that username must NOT be same as you provided for the *SAS database user* in the previous step.

   NOTE:

   1. On all the MySQL servers, a MySQL user (termed as replication user) is required. This user must have the replication privileges and access to the **mysql.users** table.
   2. Each slave MySQL database requires a MySQL user to connect to the master MySQL database. So if you have three MySQL databases (one master and two slaves) on separate machines then each database requires MySQL users associated with the other two machines. The user name and password of must be same as the replication user.

   On the master MySQL database server, run the following SQL commands.

   CREATE USER 'ReplicationUser'@'IP address of the machine where SAS HA Controller service is hosted' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the machine where SAS HA Controller service is hosted' WITH GRANT OPTION;

   CREATE USER 'ReplicationUser'@'IP address of the first slave MySQL server' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the first slave MySQL server';

   CREATE USER 'ReplicationUser'@'IP address of the second slave MySQL server' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the second slave MySQL server';

7. On the first slave MySQL database server, run the following SQL commands. Note that username and password must be same as you provided in the previous step.

   CREATE USER 'ReplicationUser'@'IP address of the machine where SAS HA Controller service is hosted' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the machine where SAS HA Controller service is hosted' WITH GRANT OPTION;

   CREATE USER 'ReplicationUser'@'IP address of the master MySQL server ' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the master MySQL server';

   CREATE USER 'ReplicationUser'@'IP address of the second slave MySQL server' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the second slave MySQL server';

8. On the second slave MySQL database server, run the following SQL commands. Note that username and password must be same as you provided in the previous step.

   CREATE USER 'ReplicationUser'@'IP address of the machine where SAS HA Controller service is hosted' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the machine where SAS HA Controller service is hosted' WITH GRANT OPTION;

   CREATE USER 'ReplicationUser'@'IP address of the master MySQL server' IDENTIFIED BY 'password';

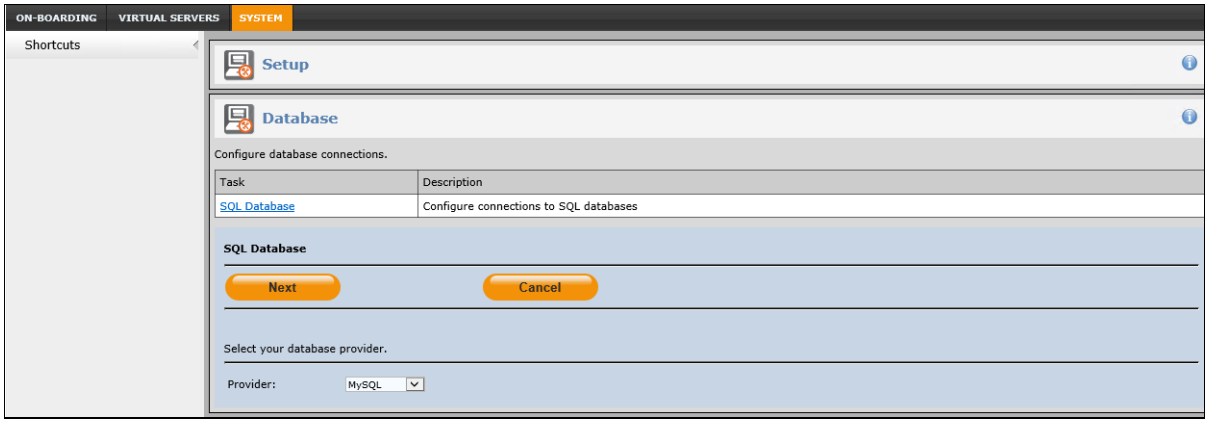
   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the master MySQL server';

   CREATE USER 'ReplicationUser'@'IP address of the first slave MySQL server ' IDENTIFIED BY 'password';

   GRANT ALL PRIVILEGES ON *.* TO 'ReplicationUser'@'IP address of the first slave MySQL server';

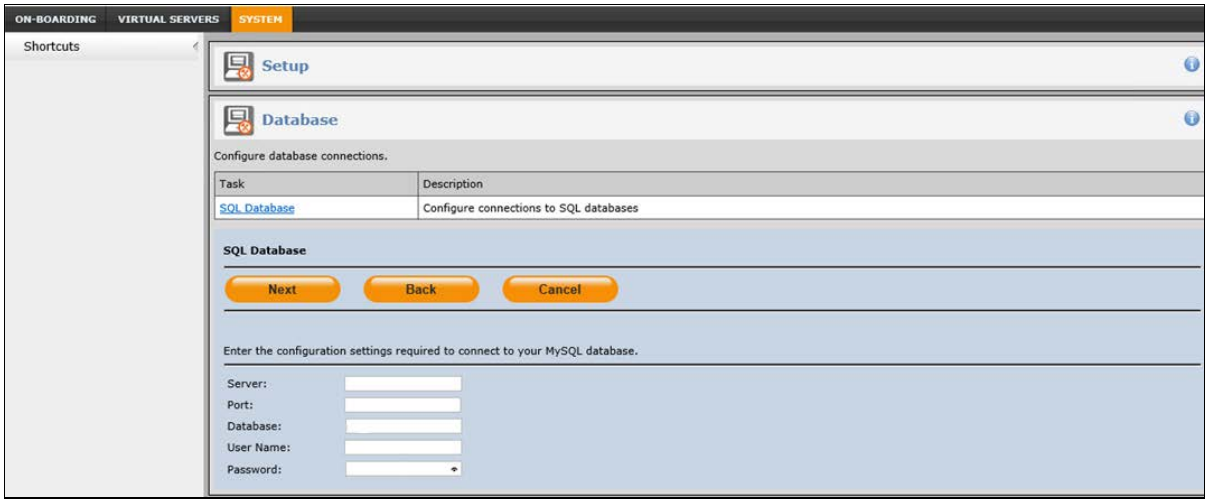
Configuring Database Settings in SafeNet Authentication Service

1. Ensure that in the registry **HKEY_LOCAL_MACHINE\SOFTWARE\CRYPTOCard\BlackShield ID\DAL\HA Service\HAModeEnable** key is set to **FALSE**.
2. Log in to SafeNet Authentication Service system level as an administrator.
3. Click the **SYSTEM** tab. Then, under the **Database** module, click the **SQL Database** link.



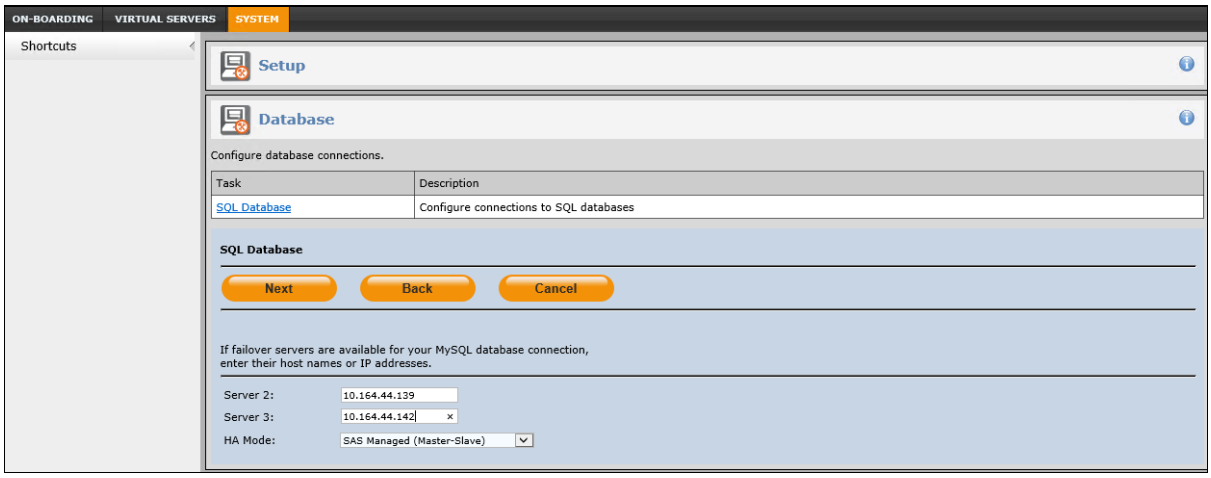
4. In the **Provider** field, select **MySQL**, and then click **Next**.
5. Complete the following fields, and then click **Next**:

Server	Enter the IP address of the **master** MySQL server.
Port	Enter the port number on which the **master** MySQL server is listening for replication.
Database	Enter the name of the **master** MySQL database that is to be created. It should be the same as specified in the **my.ini** file.
User Name	Enter the user name to be used by SAS to connect to the master MySQL database. This is the same user that you created in “Preparing MySQL Servers ” on page 18.
Password	Enter the password associated with the user name.



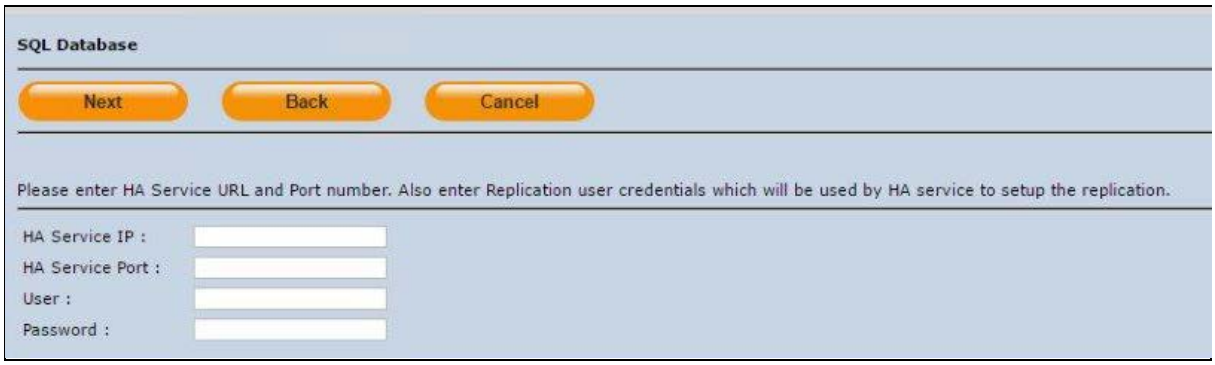
6. In the **HA Mode** field, select **SAS Managed (Master-Slave)**. Then, complete the following fields, and click **Next**. SafeNet Authentication Service will use the same credentials given in the previous step to connect to the slave MySQL servers.

Server 2	Enter the IP address of the first slave MySQL server.
Server 3	Enter the IP address of the second slave MySQL server.



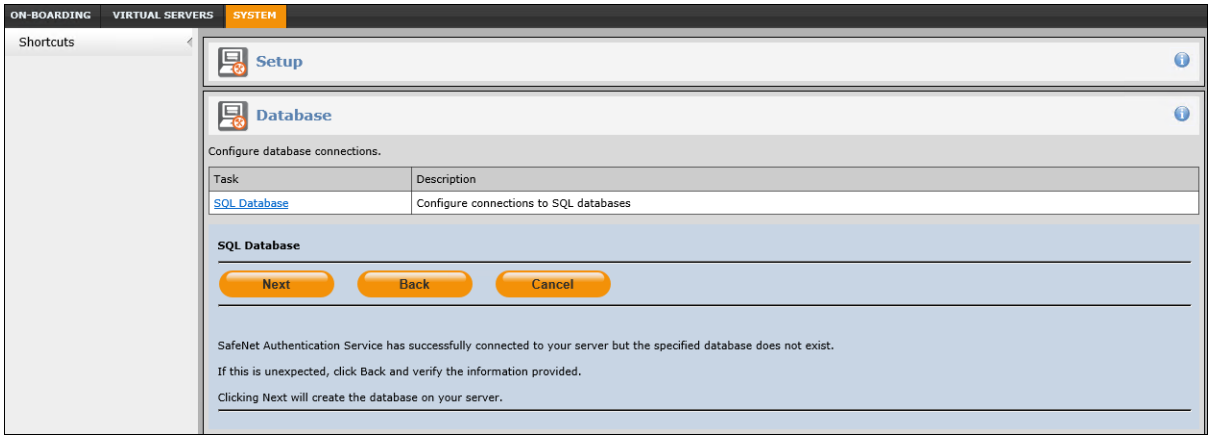
7. Complete the following fields, and then click **Next**:

HA Service IP	Enter the IP address of the machine where **SAS HA Controller Service** is installed.
HA Service Port	Enter the port number on which **SAS HA Controller Service** is listening.
User	Enter the name of the user that will be used by **SAS HA Controller Service** for replication setup.
Password	Enter the password associated with the user.

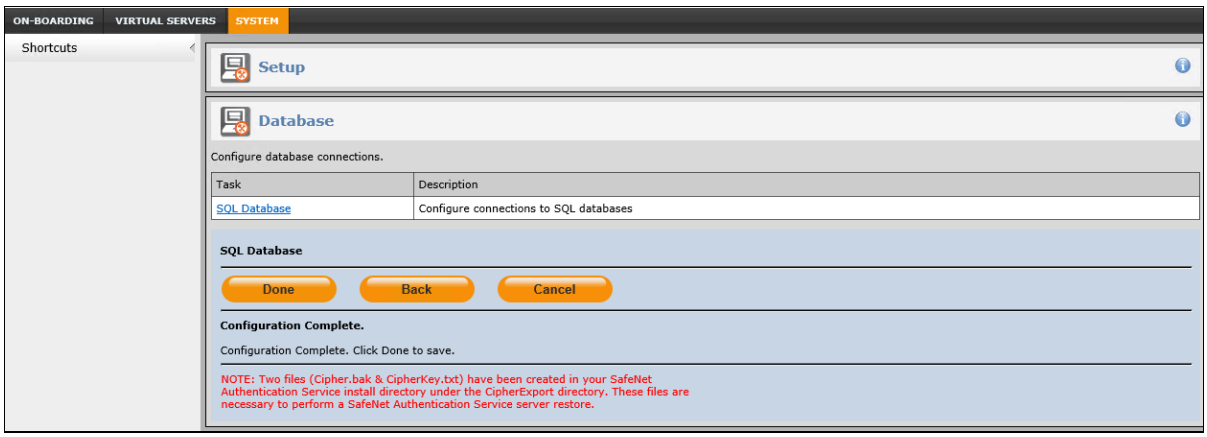


NOTE: Generally, the machine where **SAS HA Controller Service** is installed is the primary SAS server. While adding additional SAS sites, when you export and import the primary SAS server, the information regarding the MySQL HA Service URL is also copied.

8. A success message is displayed. Click **Next**.



9. Database configuration is complete. Click **Done**.



Automatic Switching Masters During Failover

When the master MySQL database becomes unavailable, one of the slave MySQL databases (the one that is most updated) is automatically promoted to master. You need to manually troubleshoot the previous master database. Once it is online, you can add it as a slave (see “Promote to Slave” on page 24).

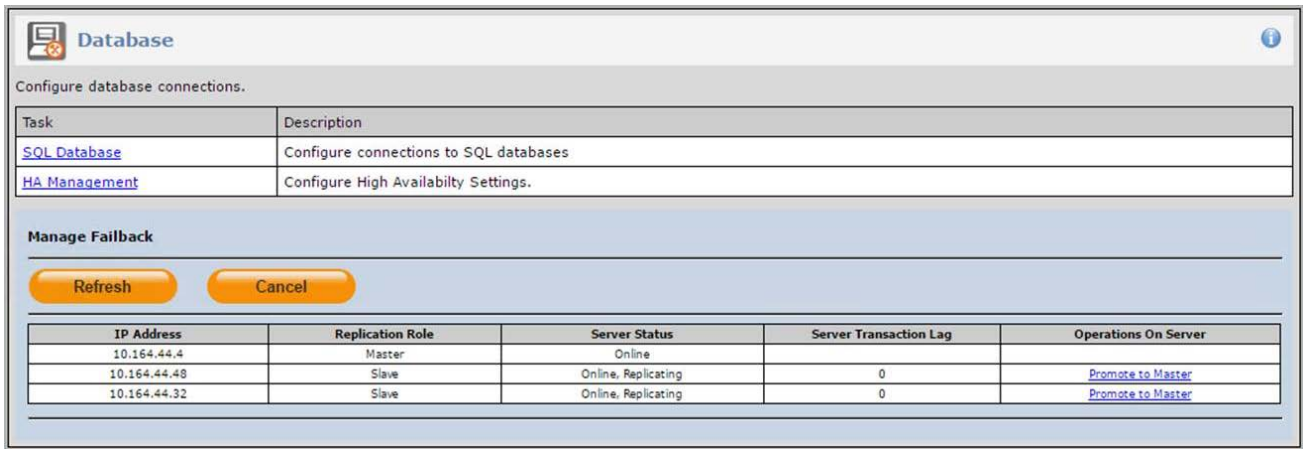
Administration Activities

The administration activities that you can perform are detailed in the sections below.

Promoting a Slave MySQL Server to Master

Due to any reason (for example, to use upgraded hardware for high performance), you may require to promote any of the slave MySQL servers to master.

1. Log in to SafeNet Authentication Service system level as an administrator.
2. Click the **SYSTEM** tab. In the **Database** module, click the **HA Management** link. The details of the master and slave databases are displayed.



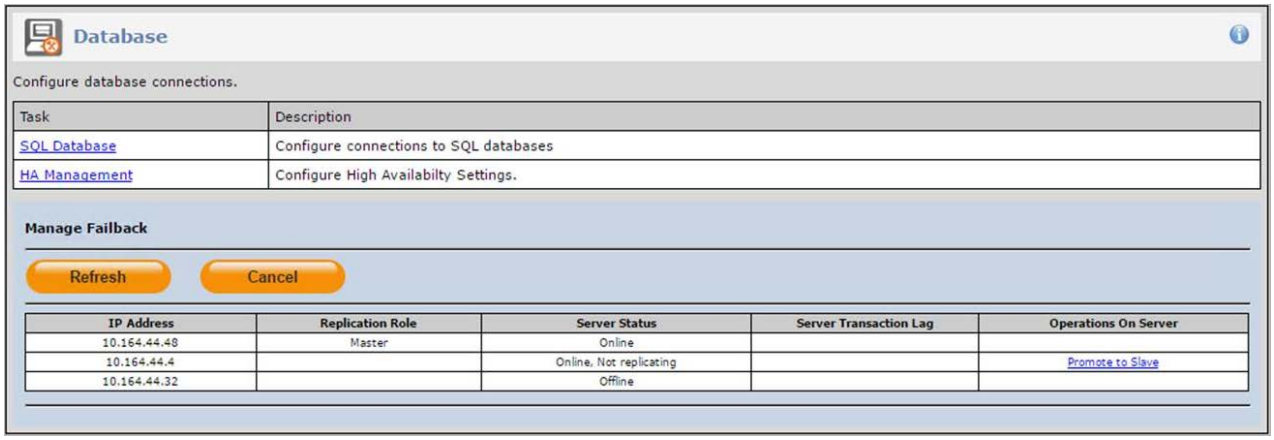
3. To promote any of the slave MySQL Servers to a master, click the respective **Promote to Master** link.

   If a slave MySQL server that you are trying to promote to master is not up to date, a message is displayed and *promoting to master* activity is discarded.

Promoting a MySQL Server to Slave

If the MySQL server is in the **Online, Non-Replicating** status and you want to add it to the replication topology, you need to promote it to slave.

1. Click the **SYSTEM** tab. In the **Database** module, click the **HA Management** link. The details of the master and slave databases are displayed.



2. This slave MySQL server will now be shown as online but not replicating. Click the **Promote to Slave** link for this server.

Adding a Slave MySQL Server

If you want to add a slave MySQL server, you need to do the following:

1. In registry, set **HAModeEnable** to **False**.
2. Turn off all services of SAS.
3. On the master MySQL database, run this command. Then, copy the value and keep it for later use.

   show global variables like 'gtid_executed';
4. In the MySQL workbench, set the following parameter to OFF.

   set-gtid-purged – Add 'SET @@GLOBAL.GTID_PURGED' to the output
5. Export the master MySQL database.
6. On the existing slave MySQL server, delete the MySQL database. Then, run **stop slave** command on this MySQL server.
7. Ensure that the **my.ini** file settings on both the slave servers are correct, and DB and replication users are properly created. For more information, refer to "Preparing MySQL Servers" on page 18.
8. Create a database with the same name as master database on both the MySQL servers to be added as slave.
9. On both the slave MySQL servers, import the master MySQL database you exported earlier.
10. Run the **RESET MASTER** command on both the salve servers:
11. Run the following command on both the slave servers after replacing **gtid_executed_value** with the value you copied in step 3:

    set global GTID_PURGED='gtid_executed_value';

12. Configure database settings in SafeNet Authentication Service. For more details, refer to “Configuring Database Settings in SafeNet Authentication Service” on page 20.
13. Turn on all services of SAS.

Removing a Slave MySQL Server

If you want to remove a slave MySQL server, you need to do the following:

1. In registry, set **HAModeEnable** to **False**.
2. Turn off all services of SAS.
3. Configure database settings in SafeNet Authentication Service. For more details, refer to “Configuring Database Settings in SafeNet Authentication Service” on page 20.
4. Turn on all services of SAS.

Troubleshooting

Hardware Failures of the Machine Hosting MySQL Server / Slave out of Replication

Turn off all services of SAS.

1. On the master MySQL database, run this command. Then, copy the value and keep it for later use.

   show global variables like 'gtid_executed';
2. In the MySQL workbench, set the following parameter to OFF.

   set-gtid-purged – Add ‘SET @@GLOBAL.GTID_PURGED’ to the output
3. Export the master MySQL database.
4. Rectify the hardware problem of the machine or get a new machine. Make sure the machine IP address is same as earlier.
5. Ensure that the my.ini file settings on the slave machine are correct, and DB and replication users are properly created. For more information, refer to “Preparing MySQL Servers” on page 18.
6. Start slave servers with the **skip_slave_start** option.
7. Create a database with the same name as master database.
8. Import the master MySQL database you exported earlier.
9. Run the **RESET MASTER** command on the slave server:
10. Run the following command on the slave server after replacing **gtid_executed_value** with the value you copied earlier:

    set global GTID_PURGED=’gtid_executed_value’;
11. Turn on all services of SAS.
12. Using the HA Management UI in SafeNet Authentication Service, promote this machine as a slave. For more details, refer to “Promote to Slave” on page 24.

You have 1 master and 2 slave MySQL servers. Both the slave MySQL servers are out of replication.

Add both the slave MySQL servers in the replication topology. Refer to “Adding a Slave MySQL Server” on page 24.

You have 1 master and 2 slave MySQL servers. Now, one slave server is replicating and the other slave server is out of replication.

1. On the replicating slave MySQL server, perform the following steps:
   a. Run the **stop slave** command.
   b. Run the following command, and then copy the value and keep it for later use:

      show global variables like 'gtid_executed';
   c. In the MySQL workbench, set the following parameter to OFF.

      set-gtid-purged – Add ‘SET @@GLOBAL.GTID_PURGED’ to the output
   d. Export MySQL database.
   e. Run the **start slave** command.
2. On the non-replicating slave MySQL server, perform the following steps
   a. Run the **stop slave** command.
   b. Delete the existing database.
   c. Create a database with the same name as previous database.
   d. Import the MySQL database you exported in step 1.d.
   e. Run the **RESET MASTER** command on this MySQL server:
   f. Run the following command on this MySQL server after replacing **gtid_executed_value** with the value you copied in step 1.b:

      set global GTID_PURGED=’gtid_executed_value’;
   g. Using the HA Management UI in SafeNet Authentication Service, promote this machine as a slave. For more details, refer to “Promote to Slave” on page 24.

When all the machines (SAS, SAS HA Controller Service, and all MySQL servers in the replicating topology) are powered off.

1. Power on the master MySQL machine and make sure that the MySQL service is started.
2. Power on rest of the MySQL machines (slaves) and start the MySQL service on these machines.
3. Start the SAS HA Controller Service, if it is installed on the machine other than the SAS server.
4. Start the SAS server.
5. In the HA Management user interface, if the slave servers are shown as online but not replicating, click the **Promote to Slave** link for these servers.

3
Configuring SafeNet Authentication Service

This section covers only the mandatory SAS configuration items. Advanced and optional configurations are detailed in the *SAS Service Provider Administrator Guide*.

- Open a web browser and then browse to **http://127.0.0.1/console** or **http://ip_address_of_server/console**.
- When prompted for credentials, enter the administrator username and password.

You will be automatically redirected to the **SYSTEM** tab where all system configuration options can be configured.

SAS configuration requires the following steps:

- Step 1 – Configure a Database
- Step 2 – Install the License
- Step 3 - Configure Email Settings
- Step 4 - Configure Self-Enrollment Policy Settings
- Step 5 – Configure Operator Email Validation URL Settings
- Step 6 - Create the Service Provider Account
- Step 7 - Create an Operator Step
- Step 8 – Define Auth Nodes

Step 1 – Configure a Database

This step connects SAS to a database server, and creates the database and tables it will use.

To configure a database:

1. On the **System** tab, click **Database > SQL Database**.
2. Select the required database from the list and then click **Next**.
3. Enter the host name, database name (default - Blackshield), user Name, and password to be used by SAS to connect to the database, and then click **Next**.
4. [Optional] If configuring a failover database server, enter similar connection information as in step 3, and then click **Next**.

> **NOTE:** If configuring MySQL database, refer to "Configuring the MySQL Database" on page 14.

5. On the **Connection Confirmation** window, click **Next** to continue, or correct any connection failure issues. Database creation may take up to a minute.
6. Copy the two Cipher files (**Cipher.bak** and **CipherKey.txt**) to a secure location and delete them from the server.

> **NOTE:** The database cannot be copied or restored to a different server or to this server in the event of significant hardware changes without the **Cipher.bak** and **CipherKey.txt** files.

7. Click **Done** to complete database installation.

Step 2 – Install the License

The license determines the number of authentication methods that can be assigned or active, and the types of tokens available.

To install the license:

1. On the **System** tab, click **Setup > Licenses.**
2. Use the **Browse** button to locate the license file (**.blc** extension).

   If this product is being provided for evaluation, use the 45-day evaluation license (30-0001457.001.blc) located in the **software/license** folder.
3. Paste the activation key into the **Activation Key** field. If this product is being used for evaluation, use the activation key (ActivationKey.txt) that comes with the 45-day evaluation license selected in the previous step.
4. Click **Import** to complete license installation.

Step 3 - Configure Email Settings

SAS uses email to send administrator validation, enrollment, and other messages.

To configure email settings:

1. On the **System** tab, click **Communications > E-mail Settings**.
2. Enter the fields as follows:

Field	Description
From Address	Enter an account name and email address Default: SafeNet Authentication Service Mailer (admin@localdomain.mail)
SMTP Server	Enter the location of the SMTP server

Field	Description
Port	Enter the Port Number of the SMTP server Default: 25
SMTP User	Enter an SMTP user name (if required)
SMTP Password	Enter an SMTP password (if required)
Use SSL	If your SMTP server supports SMTP over STARTTLS and you wish to send messages between SAS and the SMTP server over an encrypted channel, select this option.

3. Click **Apply** to commit the configuration.
4. To test the configuration, enter a valid email address in the **Test To Address** field and then click **Test**.

Step 4 - Configure Self-Enrollment Policy Settings

Self-enrollment is necessary for the provisioning and auto-provisioning of tokens to users. It is through this service that users will install software tokens or activate hardware tokens. This configuration determines the base URL included in enrollment messages sent to users.

To configure self-enrollment policy settings:

1. On the **System** tab, click **Communications > Self-Enrollment Policy**.
2. Verify or modify the default hyperlink to reflect the location of the self-enrollment website. The default location is **http:///selfEnrollment**. To require SSL for all self-enrollment processes, change the default **http://URL** to **https://URL**.

   This requires the configuration of a certificate on IIS. For details see “Configuration for MobilePASS Enrollment” on page 35.

Step 5 – Configure Operator Email Validation URL Settings

Login to the SAS management interface requires a validated email address as the UserID. This configuration determines the base URL to be sent to Operators through which they will validate their email and gain access to the management interface.

To configure the operator email validation URL:

1. On the **Systems** tab, click **Communications Module > Operator E-mail Validation URL**.
2. Verify or modify the default hyperlink to reflect the location of the Operator validation website. The default location is **http:///console/Default.aspx**. To require SSL for all self-enrollment processes, change the default **http://URL to https://URL**.

   This requires the configuration of a certificate on IIS. For details see “Configuration for MobilePASS Enrollment” on page 35.

Step 6 - Create the Service Provider Account

The Service Provider account is the organization and authentication server hosting the authentication service, and includes basic information such as company name and address. Depending on licensing, the Service Provider may be permitted to create additional accounts, all of which can be managed through SAS, but each of which appears and behaves as a unique, stand-alone enterprise authentication server. This functionality can be used to support multiple LDAPs for subsidiary organizations. Contact your supplier for additional information.

To create the Service Provider account:

1. On the **On-Boarding** tab, click **Create account**.
2. In the **Account** field, enter a unique company name.
3. Optionally, enter address information in the corresponding fields.
4. Click **Save**.

Step 7 - Create an Operator

The next step in the configuration is to create an Operator account that will be used to manage the server. The localhost administrator or root account will not be used after this point other than to reconfigure the database or install additional licenses. Apart from these functions, the Operator account has access to all functionality in the management interface.

To create an Operator:

1. On the **On-boarding** tab, click **Create Operator** and then click **Add**.
2. Enter the Operator information in the fields and click **Next**.

   The minimum requirement is **First Name**, **Last Name**, **UserID**, and **E-Mail address**. The **UserID** and **E-Mail address** must be unique.

   When through, click **Next**.

> **NOTE:** While static passwords are allowed, it is strongly recommended that all operators use two-factor authentication for log on to the management interface. The authentication methods available for provisioning to the Operator are presented in the list, along with the quantity in inventory as determined by licensing.

3. Click **Done**.

   An enrollment message is delivered to the email address entered previously.
4. The Operator should do the following:
   a. Click the hyperlink in the self-enrollment email and then follow the instructions to self-enroll.

      Immediately following completion of self-enrollment, the Operator will receive a second message containing the Operator email validation link.

   b. Click the Operator email validation link, enter the UserID (email address), and a password or one-time password, depending on the authentication method enrolled.

      If validation and authentication are successful, the Operator is logged in to the management interface.

Step 8 – Define Auth Nodes

An Auth Node must be created for any SAS Agent to allow authentication requests to SAS.

To define Auth Nodes:

1. On the **Virtual Servers** tab, select **Comms > Auth Nodes, and then** click **Add**.
2. Enter the fields as follows:

Field	Description
Agent Description	Enter a description for the agent.
Hostname	Enter the hostname of the server
Low IP Address In Range	Enter the lowest IP address in the range **Note:** If you are specifying a single IP address, enter the IP address in the **Low IP Address**. The **High IP Address** can be left empty.
High IP Address In Range	Enter the highest IP address in the range.

NOTE: If more than one IP address is required in the **Auth Node** section, expand the **Services** module and then modify the value in **Auth Nodes: Max. Auth Nodes field**.

This completes the basic configuration settings. All other configuration must be performed by the Operator account created above.

4

Configuring SafeNet Authentication Service for High Availability

A site is defined as an instance of the SAS server. The number of permitted sites is determined by the license installed on the primary SAS server.

Regardless of the architecture, establishing multiple SAS sites follows the same implementation process. The primary SAS server must be installed, configured, and capable of processing authentication requests prior to configuring additional SAS sites. Changes or additions must be configured on the primary SAS site prior to configuring any other SAS site(s), including:

- Database connection
- Export of the site key file and configuration file
- Import of the site key file and configuration file into the replica site
- Ensure that the Primary server SQL database is using host names or IP addresses:

To configure SAS sites (not the database) for high availability, ensure that the database currently used by SAS can be reached by all additional SAS sites. Ensure that the required ports are open from the additional SAS sites to the database server. For more details, see the *SafeNet Authentication Service System Requirements Guide*.

Before a replica SAS site can be configured, ensure that SAS is installed on the secondary server. A site file and file key must be generated and exported from the primary SAS server. This is done from the System Level.

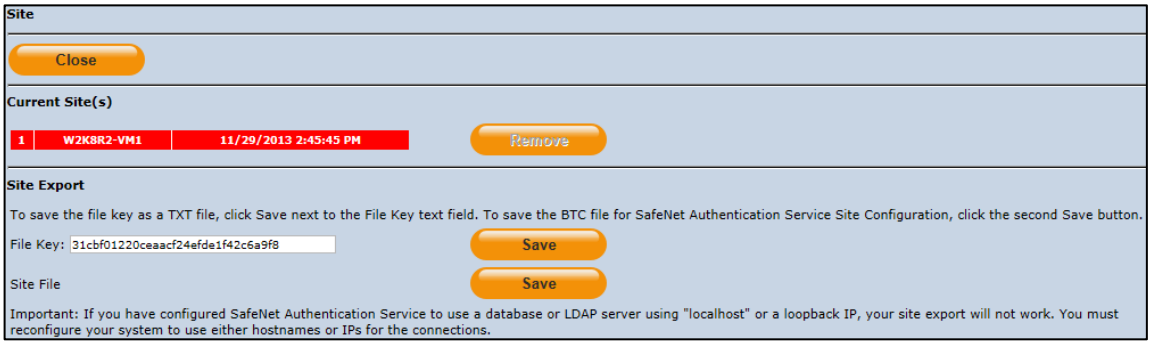
Adding a secondary SAS site requires the following steps:

- Step 1 – Export a SAS Site
- Step 2 – Import the SAS Site
- Step 3 – Add Additional SAS Sites

Step 1 – Export a SAS Site

To export a SAS site:

1. Log on locally to the primary SAS server.
2. Select the **System** tab.
3. Click the **Setup** module**.**
4. Click the **Site** link.

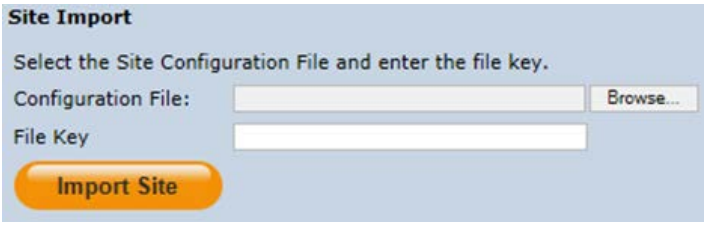


5. To save the file key, click the **Save** button in the **File Key** section and save the file to a secure location.
6. To save the site file, click the **Save** button in the **Site File** section and save the file to a secure location
7. Copy the file key and site file to the replica SAS site.

Step 2 – Import the SAS Site

To import the SAS site:

1. On the replica SAS server, log on using a local administrator account.
2. On the **System** tab, expand the **Setup** module, and then click the **Site** link.
3. Under the **Site Import** section, click **Browse** to locate and select the SAS **BSC** file



4. Open the **FileKey.txt** file and copy the key within the file.
5. Paste the key into the **File Key** field, and then click **Import Site.**

Step 3 – Add Additional SAS Sites

To add additional SAS sites:

Repeat the steps described above, Step 1 - Export SAS Site and Step 2 - Import SAS Site.

NOTE: Before reconfiguring database in secondary SAS, you need to complete these steps:

1. Set **HAModeEnable** to FALSE in the registry.
2. Set **ServiceURL** to "" (blank) in the registry.

NOTE: While trying to import user data from a primary SAS instance to secondary SAS machine(s), if the site import setup is lost, please import the complete site again from the primary SAS instance.

You may be directed to database configuration page on the secondary SAS machine(s) if the setup is lost. Please never configure the database here, or else it will make modifications on the primary SAS instance, and all the user data will be lost.

5
Configuring for MobilePASS Enrollment

To enroll MobilePASS tokens on Windows Desktop systems, a certificate must be generated and associated with SAS in Microsoft IIS.

Requirements:

- A Certificate Authority capable of issuing a web server certificate.
- A SAS server installed and configured (stand-alone or domain)

NOTE: SAS can also work with a certificate that is issued from a publicly trusted third-party root certificate authority (for example, Verisign, Comodo, GoDaddy). This is recommended for installations where self-enrollment is published to the internet.

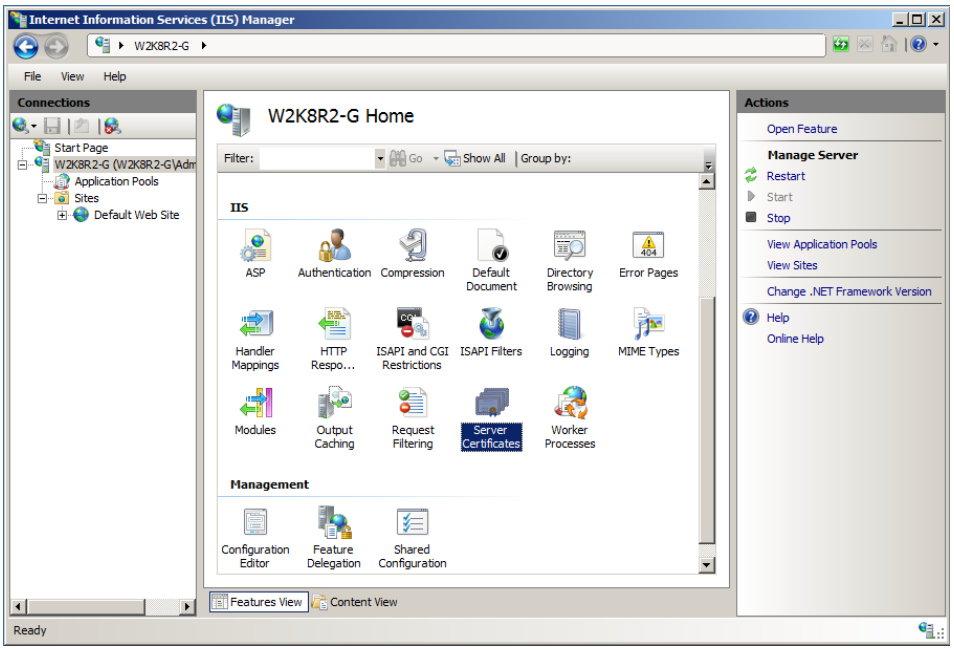
To configure SAS for MobilePASS enrollment, do the following:

- Step 1 – Create a Certificate Request from IIS
- Step 2 – Generate a Certificate from a Microsoft Certificate Authority
- Step 3 - Importing the IIS and Microsoft Root Certificate
- Step 4 - Modify the SAS Self-Enrollment URL to use SSL

Step 1 – Create a Certificate Request from IIS

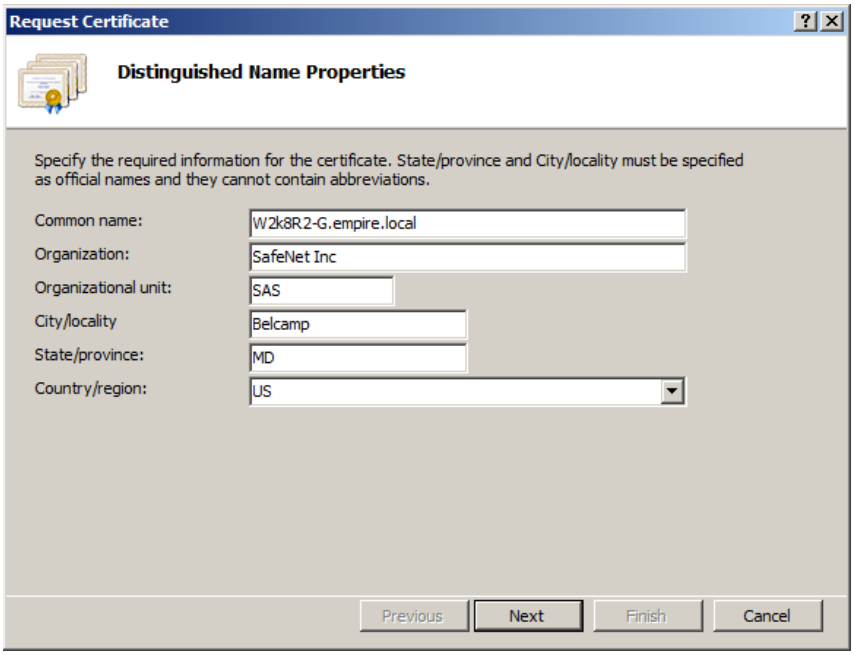
To create a certificate request from IIS:

1. On the SAS server, select **Information Internet Services (IIS).**
2. In the left pane, click the **server name**.



(The screen image above is from Microsoft. Trademarks are the property of their respective owners.)

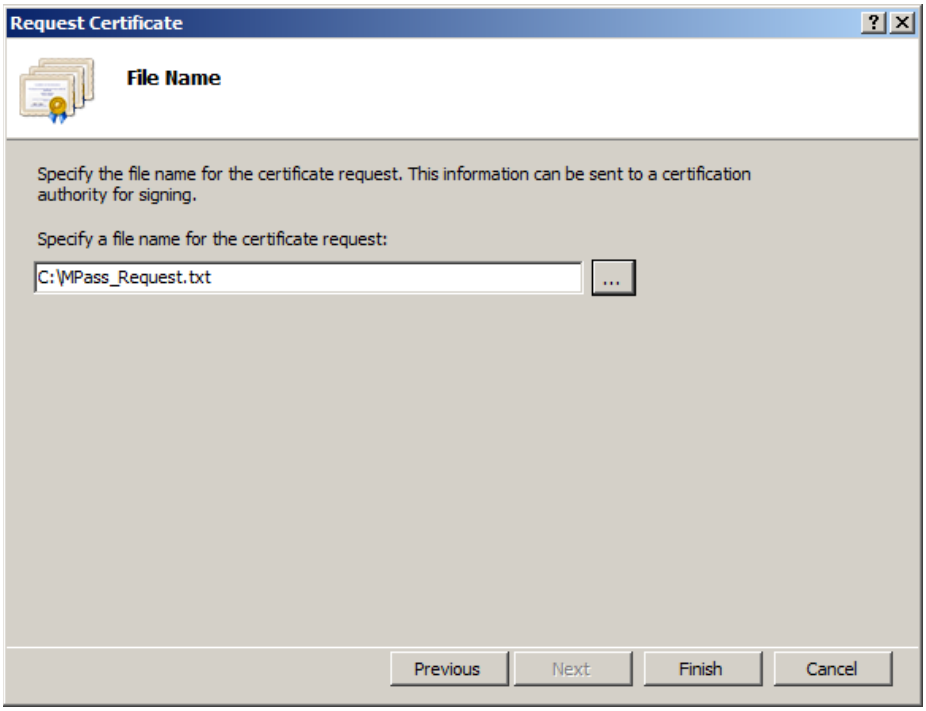
3. In the middle pane, scroll down and select **Server Certificates**
4. In the right pane of the **Server Certificates window**, click **Create Certificate Request.**



5. On the **Distinguished Name Properties** window, enter the relevant information for your organization and then click **Next**.

> **NOTE:** The Common name must be configured with the SAS full qualified domain name (or wildcard).

6. On the **Cryptographic Service Provider Properties** window, select **Cryptographic service provider**.
7. Select **Bit length**.
8. Click **Next**.
9. On the **File Name** window, click the browse button (…) and then select a location to save the certificate request text file.



10. Enter a name for the certificate request.
11. Click **Open** and then click **Finish**.

    By default the request (.txt file) is saved in the System32 folder.

> **NOTE:** When the CA is installed on the same server as SAS, the Root certificate of your CA is listed automatically in the certificates list on your IIS server.

Step 2 – Generate a Certificate from a Microsoft Certificate Authority

The certificate can be generated through one of the following:

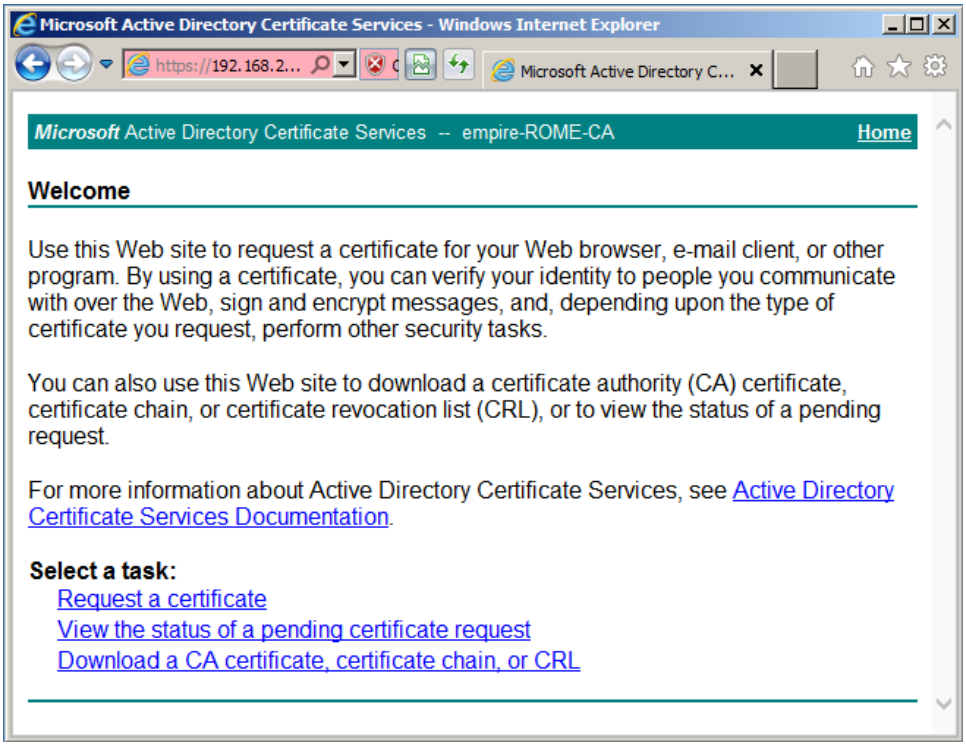
- Web Enrollment (see “Generating a Certificate through Web Enrollment” on page 39).
- The SAS server (see “Issuing the Server Certificate from the Microsoft Standalone CA” on page 40).

Generating a Certificate through Web Enrollment

To generate a certificate from a Microsoft certificate authority:

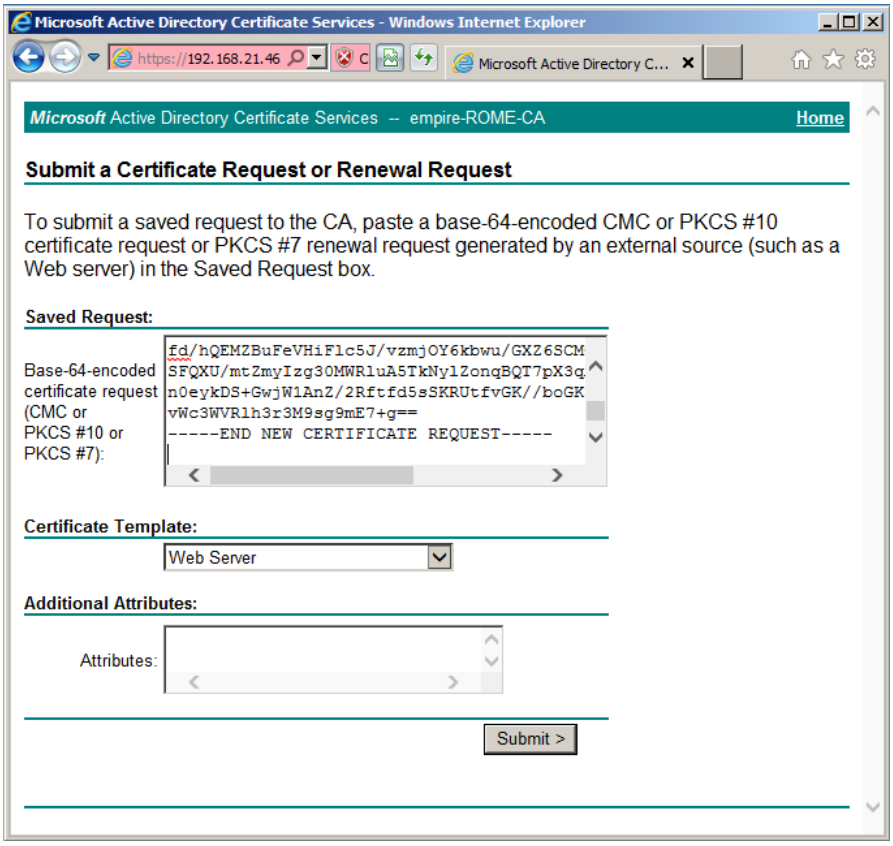
1. On the SAS Server, open a browser and go to **https:// IP address of CA/certsrv**.

> **NOTE:** If SAS is not part of a domain, the site will prompt for a login.



2. On the **Welcome** page, click **Download a CA certificate, certificate chain, or CRL**. If a pop-up message is displayed, click **No**.
3. Ensure **DER** is selected, and then click **Download a CA certificate**.
4. Save the file to the desktop as **MS Root CA.cer.**
5. Return to the previous web page and click **Request a certificate**.

6. On the **Request a Certificate** window, click **Advanced certificate request**.
7. Click **Submit a certificate request by using a base-64-encoded CMC or PKCS #10 file**, or submit a renewal request by using a base-64-encoded PKCS #7 file link.
8. On the SAS server, open the certificate request text file that was created and copy all contents of the file.



9. On the **Submit a Certificate Request or Renewal Request** window, go to the **Saved Request** section and paste the content copied from the certificate request file.
10. In the **Certificate Template** section, select **Web Server**.
11. Click **Submit**.
12. On the **Certificate Issued** window, select the **DER encoded**, and click **Download certificate**.
13. Name the certificate file **IIS_Cert.cer** and save it to a location where it can be accessed by SAS.

Issuing the Server Certificate from the Microsoft Standalone CA

As an alternative to the web enrollment, you can use the SAS server to request a certificate from the Standalone CA.

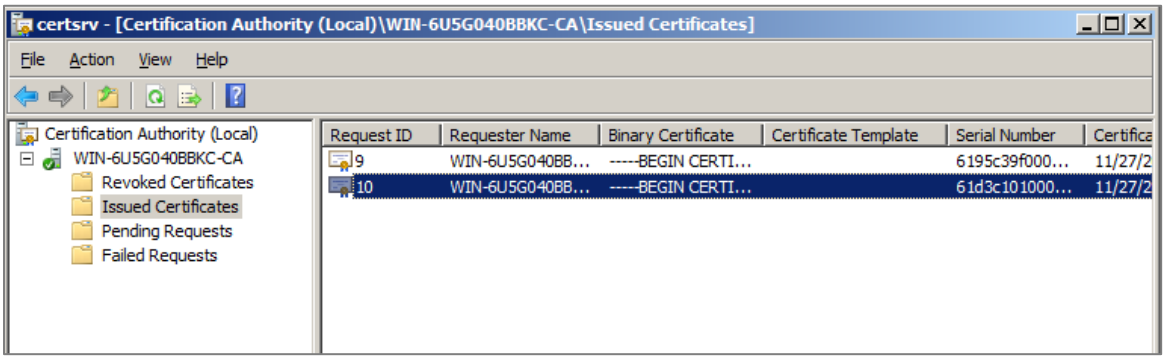
To issue the Server Certificate from the Microsoft Standalone CA:

1. From **Administrative tools**, select **Certification Authority**.
2. Highlight the **CA server**.
3. Right click the server and select **All Tasks > Submit New Request**.
4. Select the certificate request file you saved previously (see “Step 1 – Create a Certificate Request from IIS” on page 36) and click **Open (view all files).**

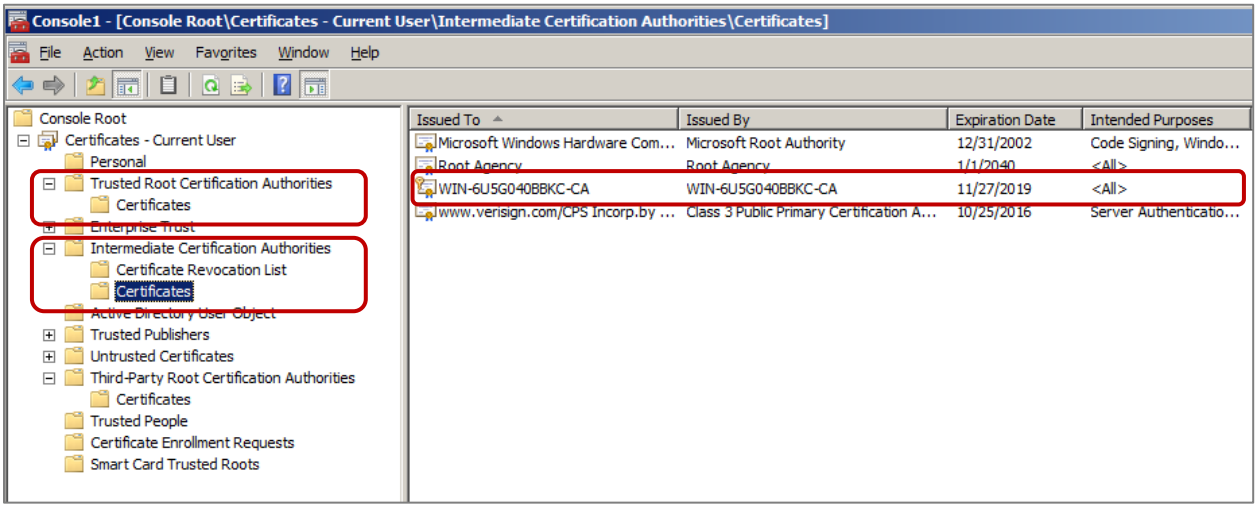
The request is listed in the **Pending requests** folder of the CA console.

5. Right click the certificate request and select **All Tasks > Issue.**

   The certificate is displayed in the **Issued Certificates** folder in the CA console.



> If the certificate is not issued, check the **Failed Requests** folder. A certificate request will fail if the root certificate is not added to the trusted certificate store and to the intermediate certificate store.

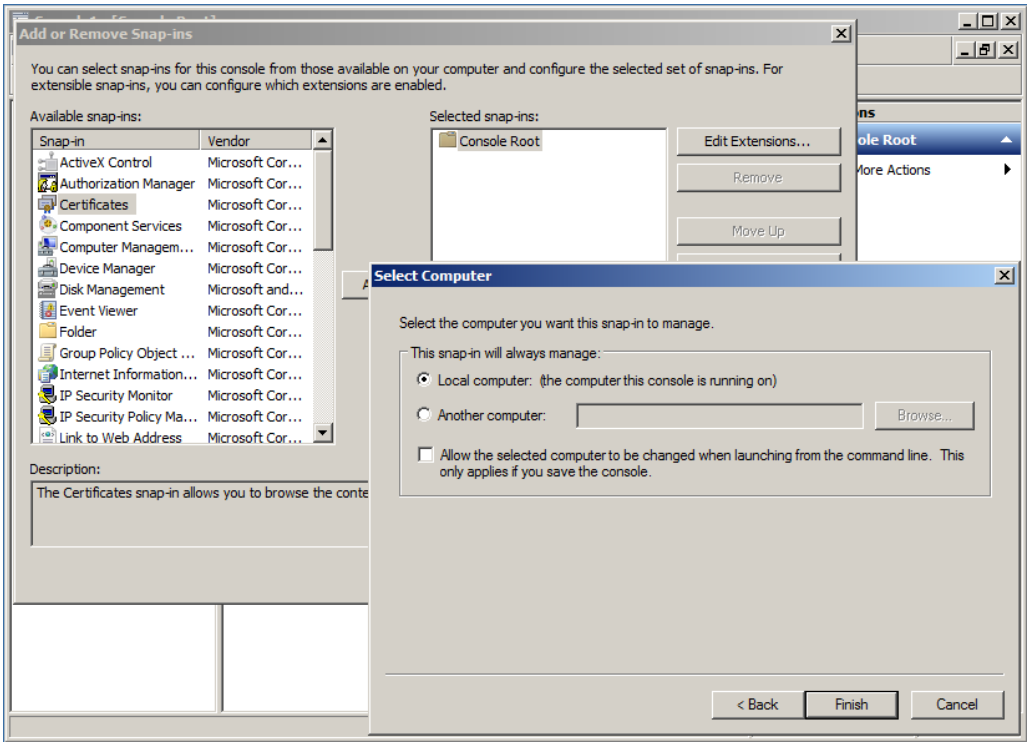


Step 3 - Importing the IIS and Microsoft Root Certificate

Importing the Certificate to the SAS Server:

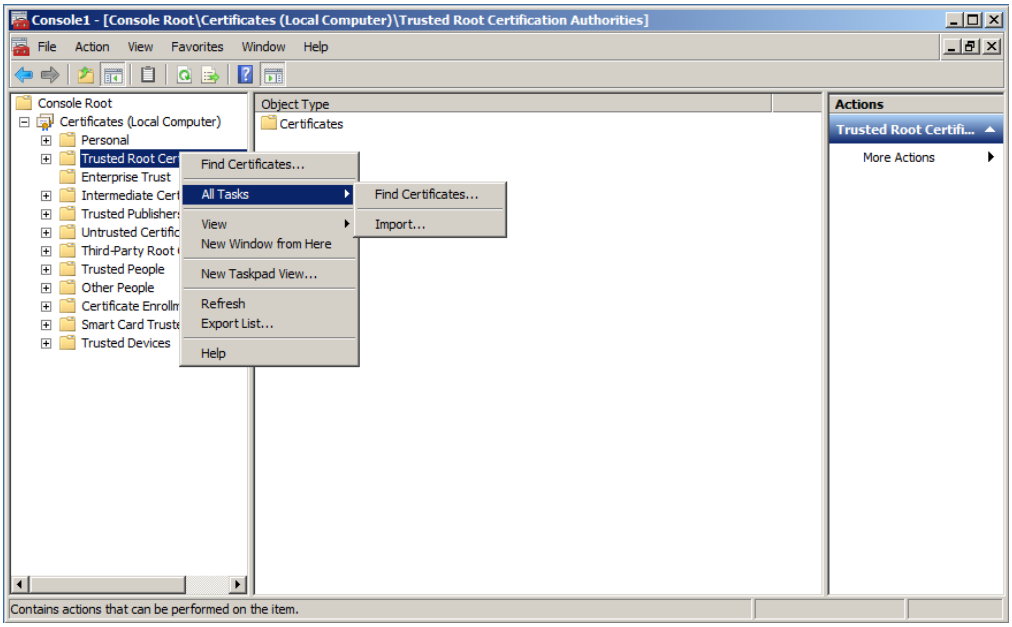
To import the certificate to the SAS server:

1. On the SAS server, select **Start > Run, t**ype **mmc**, and then press the **Enter** key.
2. On the **MMC window,** click **File > Add/Remove Snap-in**.



3. On the **Add or Remove Snap-ins** window, in the left pane, click **Certificates** and then click **Add**.
4. On the **Certificates snap-in** window, select **Computer account** and then click **Next**.

5. Select **Local computer (the computer this console is running on)**, click **Finish**, and then click **OK**.



6. In the left pane, expand the **Certificates** section, and then right-click **Trusted Root Certification Authorities.**
7. Click **All Tasks > Import.**
8. On the **Certificate Import Wizard** window, click **Next** to continue.
9. Click **Browse and then locate** the root certification authority **.CER** file. Select the file, click **Open**, and then click **Next**.
10. Ensure the option **Place all certificates in the follow store** is selected, and that **Certificate Store** is set to **Trusted Root Certification Authorities.**
11. Click **Next** and then click **Finish** to complete the wizard.
12. When prompted, click **OK** to confirm that the certificate was imported successfully.

Importing the Certificate to the IIS Server

To import the certificate to the IIS server:

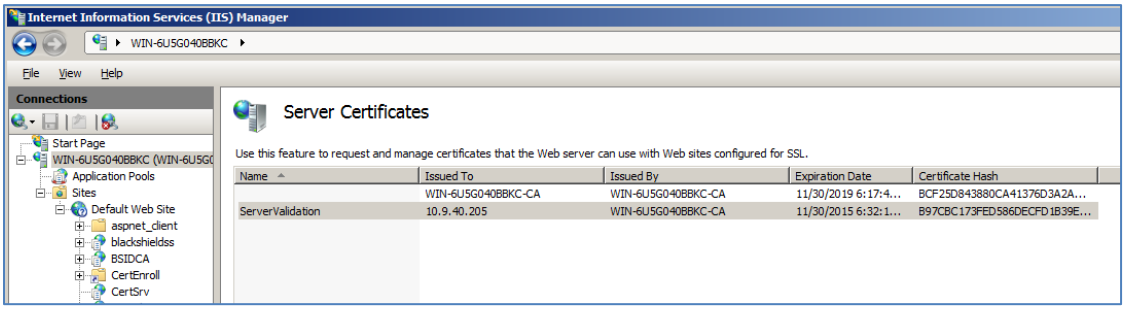
1. In the **Certification Authority** console, right click the issued certificate and select **Open**.
2. Verify that the certificate is set to **Ensures the identity of a remote computer**.
3. Select the **Certification Path** tab and verify that the certificate is **OK (root certificate is trusted).**
4. Select the **Details** tab and click **Copy to File**.

   The certificate **Export Wizard** opens.
5. Click **Next**.
6. Select the file format to be DER encoded and click **Next**.
7. Enter a name and path to the certificate file. By default it will be exported to the System32 folder.
8. Click **Next** and then click **Finish**.

   A message confirms that the export was successful.

9. In the **IIS Manager** console, highlight the IIS server and select **Server Certificates**.
10. In the **Actions** pane, select **Complete Certificate Request.**
11. Select the certificate file you have exported from the CA (.CER) and click **Open**.
12. Enter a friendly name for the certificate and click **OK**.

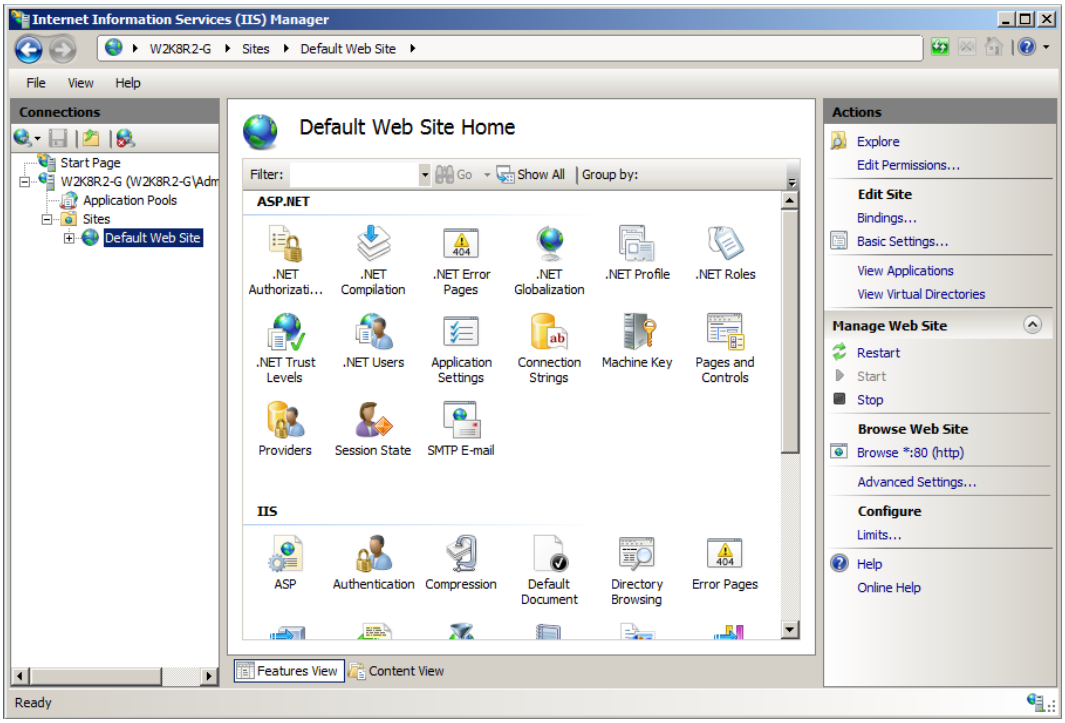
    The certificate is imported to the IIS server.



Binding the Certificate to the Website for SSL Communication

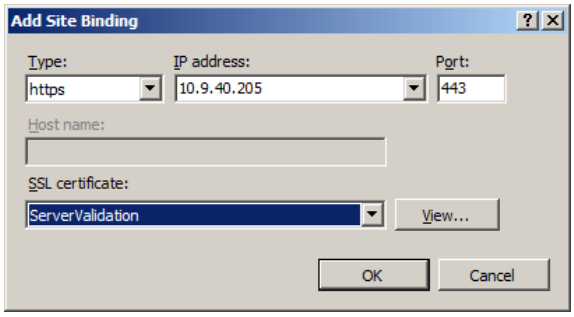
To bind the certificate to the Website for SSL communication:

1. In the left pane, expand **Sites**, click **Default Web Site, and then click Bindings.**



2. On the **Site Bindings** window, click **Add.**
3. In the Add Site Binding window enter the fields as follows:
   a. In the **Type** field, select https.
   b. In the **IP address** field select the server name or fixed IP.

c. In the **SSL certificate** field select the domain SSL certificate.



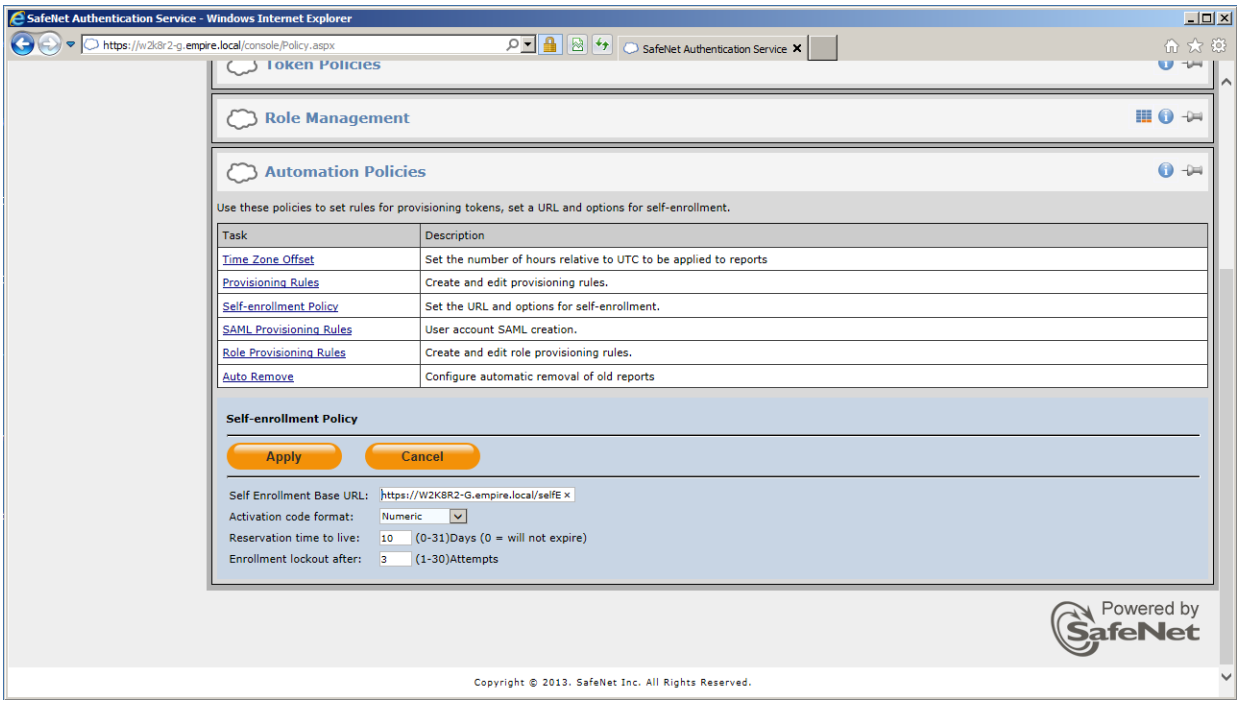
4. Click **OK** to bind the certificate to the website.

> **NOTE:** Any desktop system that is enrolling MobilePASS tokens and is not part of the domain will require the MS Root Certificate to be imported into the Trusted Root Certification Authorities.

Step 4 - Modify the SAS Self-Enrollment URL to use SSL

To modify the SAS self-enrollment URL to use SSL:

1. On the SAS server, browse to **http://localhost/console**.
2. Log in to SAS as an Operator.
3. Click **Virtual Server > Virtual Server Account Name > Policy > Automation Policies > Self-enrollment Policy.**



4. In the **Self-Enrollment Base URL field,** select the certificate DNS name that was bonded in IIS to the website:

   For example, change **http://W2k8R2-G/selfEnrollment** to **https://W2k8R2-G.empirel.local/selfEnrollment**.
5. When finished, click **Apply**.