All information herein is either public information or is the property of and owned solely by Gemalto and/or its subsidiaries who shall have and keep the sole right to file patent applications or any other kind of intellectual property protection in connection with such information.

Nothing herein shall be construed as implying or granting to you any rights, by license, grant or otherwise, under any intellectual and/or industrial property rights of or concerning any of Gemalto’s information.

This document can be used for informational, non-commercial, internal and personal use only provided that:

- The copyright notice below, the confidentiality and proprietary legend and this full warning notice appear in all copies.
- This document shall not be posted on any network computer or broadcast in any media and no modification of any part of this document shall be made.

Use for any other purpose is expressly prohibited and may result in severe civil and criminal liabilities.

The information contained in this document is provided “AS IS” without any warranty of any kind. Unless otherwise expressly agreed in writing, Gemalto makes no warranty as to the value or accuracy of information contained herein.

The document could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Furthermore, Gemalto reserves the right to make any change or improvement in the specifications data, information, and the like described herein, at any time.

Gemalto hereby disclaims all warranties and conditions with regard to the information contained herein, including all implied warranties of merchantability, fitness for a particular purpose, title and non-infringement. In no event shall Gemalto be liable, whether in contract, tort or otherwise, for any indirect, special or consequential damages or any damages whatsoever including but not limited to damages resulting from loss of use, data, profits, revenues, or customers, arising out of or in connection with the use or performance of information contained in this document.

Gemalto does not and shall not warrant that this product will be resistant to all possible attacks and shall not incur, and disclaims, any liability in this respect. Even if each product is compliant with current security standards in force on the date of their design, security mechanisms’ resistance necessarily evolves according to the state of the art in security and notably under the emergence of new attacks. Under no circumstances, shall Gemalto be held liable for any third party actions and in particular in case of any successful attack against systems or equipment incorporating Gemalto products. Gemalto disclaims any liability with respect to security for direct, indirect, incidental or consequential damages that result from any use of its products. It is further stressed that independent testing and verification by the person using the product is particularly encouraged, especially in any application in which defective, incorrect or insecure functioning could result in damage to persons or property, denial of service or loss of privacy.

© 2018 Gemalto. All rights reserved. Gemalto and the Gemalto logo are trademarks and service marks of Gemalto and/or its subsidiaries and are registered in certain countries. All other trademarks and service marks, whether registered or not in specific countries, are the property of their respective owners.

Product Version: SafeNet Authentication Service 3.7 SP2 PCE/SPE
Document Part Number: 007-012949-007, Rev. B
Release Date: February 2018
5 Configuring for MobilePASS Enrollment ................................................................. 42

Step 1 – Creating Certificate Request From IIS .............................................................. 43
Step 2 – Generating Certificate from Microsoft Certificate Authority ................................ 45
  Generating Certificate through Web Enrollment ........................................................ 45
  Issuing Server Certificate from Microsoft Standalone CA ............................................ 46
Step 3 - Importing IIS and Microsoft Root Certificate .................................................... 48
Step 4 - Modifying SAS Self-Enrollment URL to Use SSL ............................................. 52

Step 3 – Adding Additional SAS Sites................................................................................. 40
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.

<table>
<thead>
<tr>
<th>Contact Method</th>
<th>Contact Information</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Address</strong></td>
<td>Gemalto</td>
</tr>
<tr>
<td></td>
<td>4690 Millennium Drive</td>
</tr>
<tr>
<td></td>
<td>Belcamp, Maryland 21017, USA</td>
</tr>
<tr>
<td><strong>Phone</strong></td>
<td>US</td>
</tr>
<tr>
<td></td>
<td>1-800-545-6608</td>
</tr>
<tr>
<td></td>
<td>International</td>
</tr>
<tr>
<td></td>
<td>1-410-931-7520</td>
</tr>
<tr>
<td><strong>Technical Support</strong></td>
<td><a href="https://supportportal.gemalto.com/">https://supportportal.gemalto.com/</a></td>
</tr>
<tr>
<td><strong>Customer Portal</strong></td>
<td>Existing customers with a Technical Support Customer Portal account can login to manage incidents, get latest software upgrades, and access the Gemalto Knowledge Base.</td>
</tr>
<tr>
<td><strong>Documentation</strong></td>
<td>All SAS documentation (Cloud, PCE, SPE, Token and Integration) can be found on the <a href="https://supportportal.gemalto.com/">SafeNet Knowledge Base</a> page.</td>
</tr>
<tr>
<td></td>
<td>All SAS Agents documentation can be found on the <a href="https://supportportal.gemalto.com">SafeNet Authentication Service Downloads</a> page.</td>
</tr>
</tbody>
</table>

System Requirements

For details, see [SafeNet Authentication Service System Requirements Guide](https://supportportal.gemalto.com/).
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](http://www2.safenet-inc.com/sas/implementation-guides.html)

SAS on FIPS Mode Enabled Machines

SafeNet Authentication Service does not work correctly in 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.
Installing SafeNet Authentication Service on Windows

**Prerequisite Microsoft Server Components:**
- IIS 10.x with ASP.NET 3.5 for Microsoft Windows Server 2016 (64bit),
- IIS 7.x/8.x with ASP.NET 3.5 for prior versions of supported Microsoft Windows Server.
- MySQL Connector 6.9.9 (download, here)

**Notes:**
- The MySQL Connector is required only if the database in use is MySQL.
- To install MySQL Connector 6.9.9, you must uninstall any previous MySQL Connector version(s).

**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:
   1. Enter the **User Name** and **Organization**.
   2. 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.
   3. 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.

![Destination Folder Window](image)

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

![Setup Type Window](image)
8. If using a database other than PostgreSQL, right click the **PostgreSQL 9.6 (PostgreSQL 9.6.4) Database**, select **This feature will not be available** and then click **Next** to return to the **Setup Type** window.

   **NOTE:** We recommend using PostgreSQL 9.6 (PostgreSQL 9.6.4) 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
Preparing Database

SafeNet Authentication Service supports the following databases:

- PostgreSQL
- MS SQL
- MySQL

Preparing PostgreSQL Database

PostgreSQL should be used only for test and proof-of-concept installations. Is not supported in HA configurations. For details, refer Step 1 – Configure a Database on page 34.

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

Preparing 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 Step 1 – Configure a Database on page 34. Also, refer Microsoft documentation.

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

You can configure MySQL database with or without high availability. The detailed information on setting up MySQL database with High Availability (HA) 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 HA solution supported with SafeNet Authentication Service is MySQL Master-Slave Replication. MySQL Replication is the most popular and cost-effective HA 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 High Availability 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:** Only one SAS HA Controller Service (High Availability service) should be up and running for an SAS MySQL HA installation. Also, 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 of 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 on Windows

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.
Preparing Database

**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.

   ```sql
   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.

   ```sql
   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';

Setting up Highly Available MySQL Database on Linux

Perform the following steps to set up MySQL HA with Linux environment:

1. Using the Terminal, stop the MySQL service by running the following command:

   ]# service mysqld stop

2. Open the /etc/my.cnf file and add the below lines within [mysqld]:

```
# START
binlog-format=ROW
log-slave-updates=true
enforce-gtid-consistency=true
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
```
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. Delete the `auto.cnf` file available at the following location:
   ```bash
   /var/lib/mysql
   ```
   This will result in generation of a unique GTID for the MySQL server. All MySQL servers that are in replication must have a unique GTID.

4. Using the **Terminal**, start the MySQL service by running the following command:
   ```bash
   # service mysqld start
   ```

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.
   ```sql
   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:**

a. 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.
b. 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.

```sql
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.

```sql
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.

```sql
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';

**NOTE:** The procedure help is provided to support SAS setup with MySQL HA on Linux (machines) and should primarily be managed by customer's database administrators. Since Linux is not an officially supported operating system with SAS, best effort support can be provided from the application end, in case of any issues.

### 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.

![Database Configuration Screen](image)

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

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Server</strong></td>
<td>Enter the IP address of the <strong>master</strong> MySQL server.</td>
</tr>
<tr>
<td><strong>Port</strong></td>
<td>Enter the port number on which the <strong>master</strong> MySQL server is listening for replication.</td>
</tr>
<tr>
<td><strong>Database</strong></td>
<td>Enter the name of the <strong>master</strong> MySQL database that is to be created. It should be the same as specified in the <strong>my.ini</strong> file.</td>
</tr>
<tr>
<td><strong>User Name</strong></td>
<td>Enter the user name to be used by SAS to connect to the <strong>master</strong> MySQL database. This is the same user that you created in <strong>Preparing MySQL Servers</strong> on page 18.</td>
</tr>
<tr>
<td><strong>Password</strong></td>
<td>Enter the password associated with the user name.</td>
</tr>
</tbody>
</table>
NOTE: To connect MySQL database, you will require MySQL Connector. If it is already not downloaded and installed on your machine, the following screen is displayed, with a download link and steps to guide you during the installation. During installation, ensure that you choose either Typical or Complete installation.

To connect to a MySQL database the MySQL .NET Connector needs to be installed.
1. Click on the link below to download the MySQL .NET Connector installer (Version 6.9.9).
2. Launch the MySQL .NET Connector installer and complete the installation.
3. Restart this application and all background services.
4. Return to this configuration screen and finish the configuration.

NOTE: If MySQL Connector 6.4.4 is already installed, then you have to manually remove it in order to install MySQL Connector 6.9.9. On installing the MySQL Connector, the Administrator will be logged out of the application, and will have to login again to complete the rest of the configurations.

6. 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.

<table>
<thead>
<tr>
<th>Server 2</th>
<th>Enter the IP address of the first slave MySQL server.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server 3</td>
<td>Enter the IP address of the second slave MySQL server.</td>
</tr>
<tr>
<td>HA Mode</td>
<td>Select Manual replication (Master-Master) option.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>HA Service IP</th>
<th>Enter the IP address of the machine where SAS HA Controller Service is installed.</th>
</tr>
</thead>
<tbody>
<tr>
<td>HA Service Port</td>
<td>Enter the port number on which SAS HA Controller Service is listening.</td>
</tr>
<tr>
<td>User</td>
<td>Enter the name of the user that will be used by SAS HA Controller Service for replication setup.</td>
</tr>
<tr>
<td>Password</td>
<td>Enter the password associated with the user.</td>
</tr>
</tbody>
</table>
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 27).

Administration Activities

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

Promoting Slave MySQL Server to Master

Due to any reason (for example, to use upgraded hardware for high performance), you may require promoting 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 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 Slave MySQL Server

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

1. In the 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.
   ```sql
   show global variables like 'gtid_executed';
   ```
4. In the MySQL workbench, set the following parameter to **OFF**.
   ```sql
   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 [Preparing MySQL Servers](#) on page 18.
8. Create a database with the same name as a master database on both the MySQL servers to be added as a 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 slave servers:
11. Run the following command on both the slave servers after replacing `gtid_executed_value` with the value you copied in step 3:
   ```sql
   set global GTID_PURGED='gtid_executed_value';
   ```
13. Turn on all services of SAS.
Removing Slave MySQL Server

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

1. In the registry, set `HAModeEnable` to `False`.
2. Turn off all services of SAS.
4. Turn on all services of SAS.

Setting up FreeRADIUS API

In case of a fresh SAS installation with MySQL database, RADIUS API encounters an issue with MySQL database (MySQL EF6 DLL in GAC missing). It is a limitation of MySQL Connector 6.9.9.

Ensure that the following steps are already performed:

1. After installing SAS, install MySQL 6.9.9 Connector.
2. Configure SAS with MySQL database.

Follow the steps:

For Windows Server 2008 R2 SP1

1. Copy the following text in a text file and save the file in the `.ps1` file format:

   ```
   $config_text = @"$
   <?xml version="1.0"?>
   <configuration>
   <startup useLegacyV2RuntimeActivationPolicy="true">
   <supportedRuntime version="v4.0.30319"/>
   <supportedRuntime version="v2.0.50727"/>
   </startup>
   </configuration>
   "$
   $config_text| Out-File $pshome\powershell.exe.config
   $config_text| Out-File $pshome\powershell_ise.exe.config
   ```

2. Save it and rename the file extension to `.ps1`.

3. Run as Administrator in the PowerShell.
NOTE: For Windows Server 2008 R2 SP1, the administrator also needs to follow the steps in the For Windows Server 2012, 2012 R2 and 2016 section.

For Windows Server 2012, 2012 R2 and 2016

1. Copy the following text in a text file and save the file in the .ps1 file format:

```
#Note that you should be running PowerShell as an Administrator
[System.Reflection.Assembly]::Load("System.EnterpriseServices, Version=4.0.0.0, Culture=neutral, PublicKeyToken=b03f5f7f11d50a3a")
$publish.GacInstall("C:\Program Files (x86)\MySQL\MySql Connector Net
6.9.9\Assemblies\v4.5\MySql.Data.Entity.EF6.dll")
# If installing into the GAC on a server hosting web applications in IIS,
you need to restart IIS for the applications to pick up the change.
iisreset
```

2. Run the .ps1 file, as an Administrator in the PowerShell.

3. Reset IIS.

Points to Remember

1. Default location:
   System Directory:\Program Files (x86)\MySQL\MySql Connector Net
   6.9.9\Assemblies\v4.5

2. If someone changes the directory location while installing the MySQL Connector, the above path also needs to be updated in the script.

3. Open the PowerShell script and change the path to where your DLL resides.

Setting up MS SQL Database

You can configure MS SQL database with or without high availability. MS SQL is used with many applications demanding availability and scalability. If you need to work with AlwaysOn Availability Groups in MS SQL, follow the steps to configure and set it up with the SAS solution.

NOTE: The procedure’s help reference is provided to support SAS setup with the respective HA model and should primarily be managed by customer’s database administrators. Best effort support can be provided from the application end in case of any issues.

Troubleshooting

For Hardware Failures of 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.
   ```sql
   show global variables like 'gtid_executed';
   ```
2. In the MySQL workbench, set the following parameter to OFF.
   ```sql
   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 [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 the 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:
    ```sql
    set global GTID_PURGED='gtid_executed_value';
    ```
11. Turn on all services of SAS.
12. Using the HA Management User Interface (UI) in SafeNet Authentication Service, promote this machine as a slave. For more details, refer [Promote to Slave](#) on page 27.

**If 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 [Adding a Slave MySQL Server](#) on page 28.

**If you have 1 master and 2 slave MySQL servers. Now, one slave server is replicating and the other slave server is out of replication.**
3. On the replicating slave MySQL server, perform the following steps:
   1. Run the `stop slave` command.
   2. Run the following command, and then copy the value and keep it for later use:
      ```sql
      show global variables like 'gtid_executed';
      ```
   3. In the MySQL workbench, set the following parameter to OFF.
      ```sql
      set-gtid-purged – Add ‘SET @@GLOBAL.GTID_PURGED’ to the output
      ```
   4. Export MySQL database.
   5. Run the `start slave` command.
13. On the non-replicating slave MySQL server, perform the following steps
1. Run the `stop slave` command.
2. Delete the existing database.
3. Create a database with the same name as the previous database.
4. Import the MySQL database you exported in step 1.d.
5. Run the **RESET MASTER** command on this MySQL server:

6. 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';
   ```

7. Using the HA Management UI in SafeNet Authentication Service, promote this machine as a slave. For more details, refer **Promote to Slave** on page 27.

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

   I. Power on the master MySQL machine and make sure that the MySQL service is started.
   
   II. Power on rest of the MySQL machines (slaves) and start the MySQL service on these machines.
   
   III. Start the SAS HA Controller Service, if it is installed on the machine other than the SAS server.
   
   IV. Start the SAS server.
   
   V. 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.
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:

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

Step 1 – Configuring 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.
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.

   **NOTE:** After configuring SAS with a database, if an Administrator plans to configure the SAS again with a different database, the steps must be followed:
   1. Configure SAS with the required database.
   2. Once the configuration is complete, perform an IISRESET operation.
   3. Login to SAS, and provide the license again.

### Step 2 – Installing 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 - Configuring 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:
### Step 4 - Configuring 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 42.

### Step 5 – Configuring 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`.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>From Address</td>
<td>Enter an account name and email address</td>
</tr>
<tr>
<td></td>
<td>Default: SafeNet Authentication Service Mailer (<a href="mailto:admin@localdomain.mail">admin@localdomain.mail</a>)</td>
</tr>
<tr>
<td>SMTP Server</td>
<td>Enter the location of the SMTP server</td>
</tr>
<tr>
<td>Port</td>
<td>Enter the Port Number of the SMTP server</td>
</tr>
<tr>
<td></td>
<td>Default: 25</td>
</tr>
<tr>
<td>SMTP User</td>
<td>Enter an SMTP user name (if required)</td>
</tr>
<tr>
<td>SMTP Password</td>
<td>Enter an SMTP password (if required)</td>
</tr>
<tr>
<td>Use SSL</td>
<td>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.</td>
</tr>
</tbody>
</table>
This requires the configuration of a certificate on IIS. For details, see Configuration for MobilePASS Enrollment on page 42.

**Step 6 - Creating 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 - Creating 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 logging 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:
   
   1. 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.
2. 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 – Defining 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:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent Description</td>
<td>Enter a description for the agent.</td>
</tr>
<tr>
<td>Hostname</td>
<td>Enter the hostname of the server</td>
</tr>
<tr>
<td>Low IP Address In Range</td>
<td>Enter the lowest IP address in the range</td>
</tr>
<tr>
<td><strong>Note:</strong> If you are specifying a single IP address, enter the IP address in the <strong>Low IP Address</strong>. The <strong>High IP Address</strong> can be left empty.</td>
<td></td>
</tr>
<tr>
<td>High IP Address In Range</td>
<td>Enter the highest IP address in the range.</td>
</tr>
</tbody>
</table>

**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.
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:

1. The database currently used by SAS can be reached by all additional SAS sites.
2. The required ports are open from the additional SAS sites to the database server. For more details, see **SafeNet Authentication Service System Requirements Guide**.
3. The SAS HA Controller Service is enabled on only one of the servers. To ensure the same, refer note section of **Step 3 – Add Additional SAS Sites** on page 37.

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 at the System Level.

**Adding a secondary SAS site requires the following steps:**

- **Step 1 – Export an SAS Site**
- **Step 2 – Import the SAS Site**
- **Step 3 – Add Additional SAS Sites**
Step 1 – Exporting SAS Site

To export an 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 – Importing 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 Choose File 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.

NOTE: To connect MySQL database, you will require MySQL Connector. If it is already not downloaded and installed on your machine, the following screen is displayed, with a download link and steps to guide you during the installation. During installation, ensure that you choose either Typical or Complete installation.
NOTE: If MySQL Connector 6.4.4 is already installed, then you have to manually remove it in order to install MySQL Connector 6.9.9. On installing the MySQL Connector, the Administrator will be logged out of the application, and will have to login again to complete the rest of the configurations.

**Step 3 – Adding 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 perform the following steps to disable the HA Controller service on secondary SAS.

1. Set `HAModeEnable` to FALSE in the registry.
2. Set `HAServiceURL` to "" (blank) in the registry.

Both the above values can be found at the HA Service registry key, available at the following path:

```
HKEY_LOCAL_MACHINE\SOFTWARE\CRYPTOCard\BlackShield ID
```

**WARNING:** 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.
**NOTE:** SAS servers can be distributed over different data centers with different time zone configurations. This may create an issue during the site import process.

Before site import, it is advised to ensure that the display time on all SAS machines is synchronized.
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.
- An SAS server installed and configured (stand-alone or domain)

**NOTE:** SAS can also work with a certificate that is issued by 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 – Creating 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**.

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.

![Request Certificate](image)

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 – Generating Certificate from Microsoft Certificate Authority

The certificate can be generated through one of the following:

- Web Enrollment (see Generating a Certificate through Web Enrollment, below).
- The SAS server (see Issuing the Server Certificate from the Microsoft Standalone CA on page 42).

Generating 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.


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 Server Certificate from 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 43) 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 IIS and Microsoft Root Certificate

Importing Certificate to SAS Server:

To import the certificate to the SAS server:

1. On the SAS server, select Start > Run, type 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 Certificate to 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.

![](image1.jpg)

**Binding Certificate to 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 details as follows:
   1. In the **Type** field, select https.
   2. In the **IP address** field select the server name or fixed IP.
   3. In the **SSL certificate** field select the domain SSL certificate.

![](image2.jpg)
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 - Modifying SAS Self-Enrollment URL to Use SSL

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

2. Login to SAS as an Operator.
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.