Value-Added Module (VAM)

Epic EPCS VAM Deployment Guide

SECUREAUTH

Copyright Information

Document Revision History

Version	Date	Notes
0.1	16-March-2017	Initial draft
1.0	15-June-2018	First version
2.0	04-December-2019	Second version
3.0	10-September-2020	Third version

For information on support for this module, contact your SecureAuth support or sales representative:

Email: support@secureauth.com inside-sales@secureauth.com

Phone: +1-949-777-6959
+1-866- 859-1526
Website: https://www.secureauth.com/support
https://www.secureauth.com/contact

Contents

Overview

This document details the deployment and configuration of the Epic EPCS Value-Added Module (VAM) on a SecureAuth® Identity Platform (formerly known as SecureAuth IdP) appliance. The addition of the Epic EPCS VAM in your environment enables authentication and authorization of applications on Epic EPCS.

The SecureAuth Epic EPCS VAM enables seamless integration between the Identity Platform multi-factor authentication (MFA) and the Epic Hyperspace platform for the Electronic Prescriptions of Controlled Substances (EPCS) system. Using this integrated package, qualified physicians can write prescriptions quickly and securely while meeting DEA requirements for electronic prescriptions.

-

This guide also includes instructions on installing and configuring the VAM that enables the link between Epic Hyperspace and the Identity Platform.

What's new in version 3.0

The Value-Added Module (VAM) by SecureAuth for Epic ECPS in version 3.0 supports selection of MFA options for authentication, including biometric options. As part of the login worfklow, the user interface shows all the available MFA options to choose from.

Why select the SecureAuth Identity Platform?

SecureAuth's flexible authentication framework allows providers to deploy DEA compliant two-factor authentication (2FA) in ways that are not intrusive on physicians; in many cases SecureAuth can actually optimize workflows by reducing clicks. Its aim is to provide the quickest way to ensure that the accessing physician is the one authorized to approve the prescription, per DEA standards.

Features

- Seamless integration into preexisting Epic e-Prescribing workflows
- Multiple authentication methods that not only meet DEA regulation but make 2FA easy for physicians – such as push-to-accept, fingerprint, and other DEA-compliant methods
- Flexible authentication platform that allows providers to select the 2FA method which best meets their needs

Benefits and use cases

The Epic EPCS VAM enables SecureAuth MFA and modern authentication methods for stronger security.

MFA and adaptive authentication

MFA methods used for EPCS transactions are required to meet **FIPS 140-2 Level 1**. Choose among the following available authentication methods.

- SecureAuth Mobile <Push-To-Accept>
- SecureAuth Mobile App <Soft Token> TOTP
- SecureAuth <Hardware> Token TOTP
- SecureAuth Mobile Fingerprint
- SecureAuth Mobile FaceID

Note: Verify that these methods meet your organization's interpretation of the EPCS Guidelines.

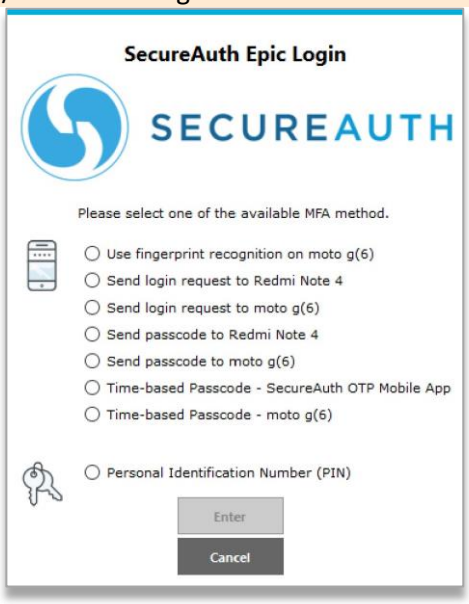
List of MFA options

The following example illustrates a newly added workflow presenting the list of all FIPS 140-2 compliance MFA methods for the user to choose from. To activate this new workflow we have to update `ShowMFA="1"` in SASettings.xml, or else it works as before in version 2.0.

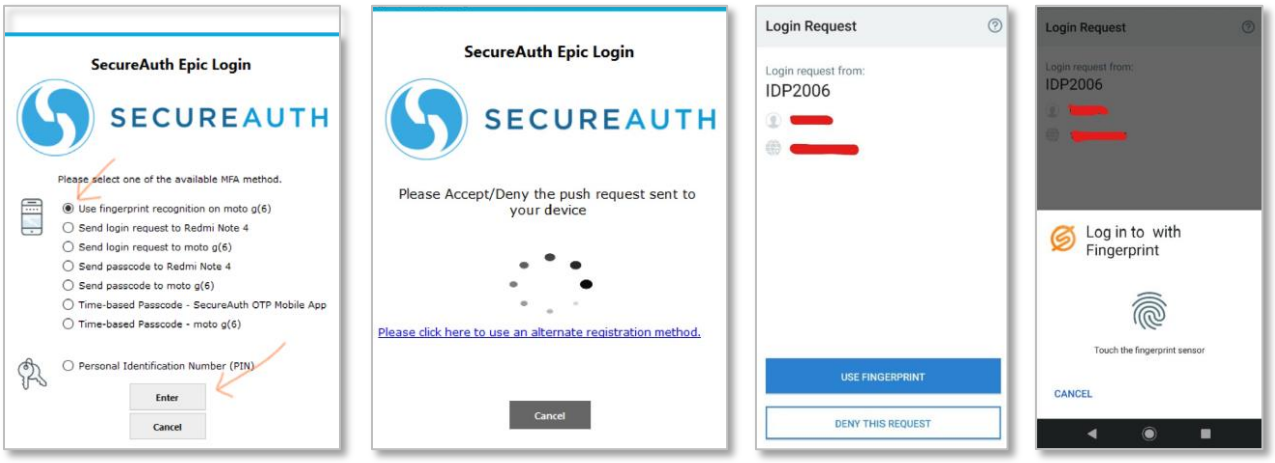
In this example:

1. The physician enters a username and is prompted for a method to receive the required password; if `RequirePassword` key is set to `1` in SASettings.xml, or else it skips to step 3 in the login workflow.
2. The physician enters the password and presses **Enter**.
3. This opens a user interface with list of available MFA options.
4. The physician needs to select a preferred MFA option and presses **Enter**.
5. Based on the selected MFA option, the next page appears to complete 2FA.
6. The physician receives entry to the EPCS system.

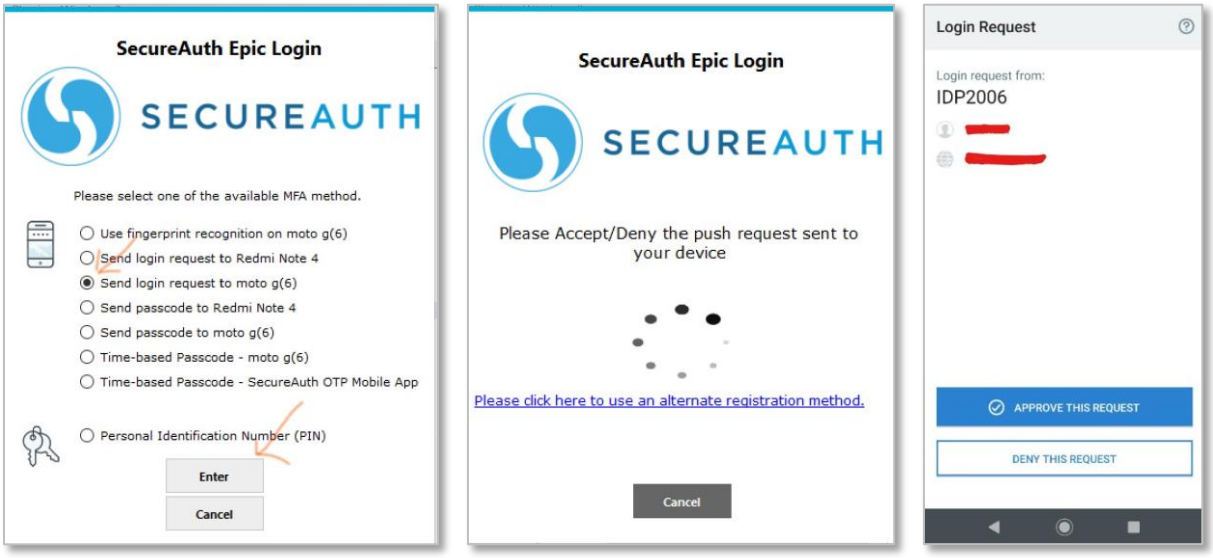
Note: Reading a fingerprint/FaceID is compliant with the FIPS 140-2 standard and can only be handled by those mobile devices that support fingerprint/FaceID reading.



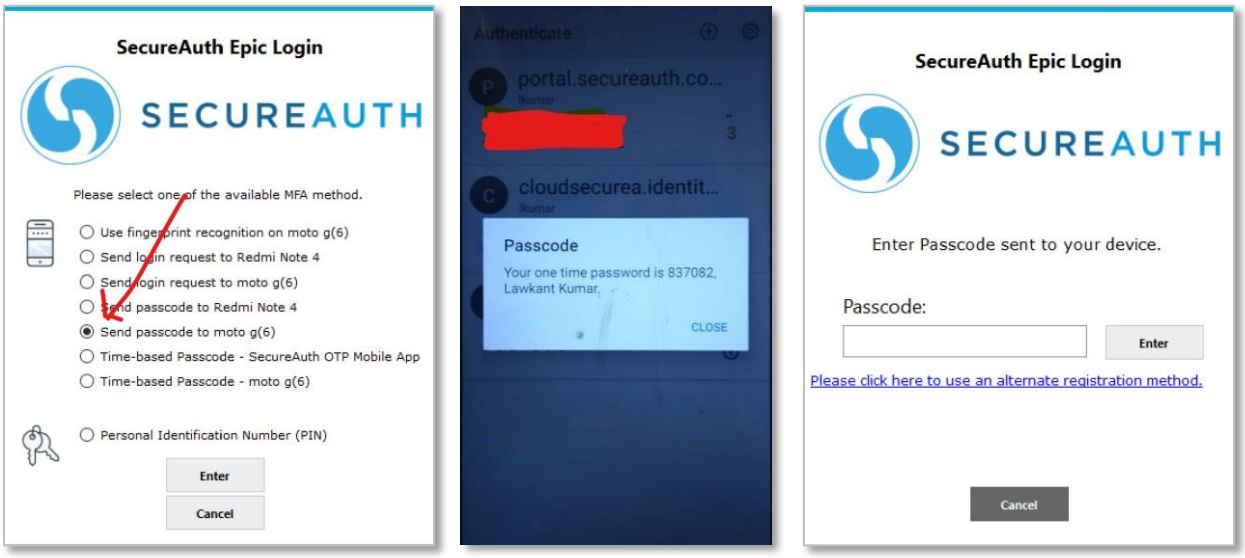
Fingerprint/FaceID: When a physician selects fingerprint or face ID, they will receive a fingerprint/faceID request on their mobile device.



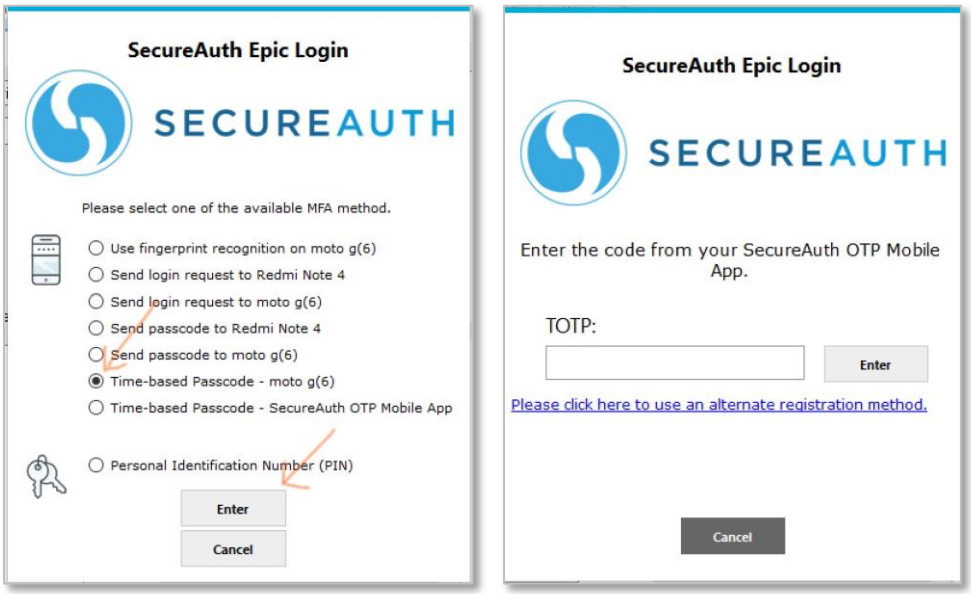
- **Send Login Request (Push to Accept)**: When a physician selects send login request, they will receive a login request on their mobile device.



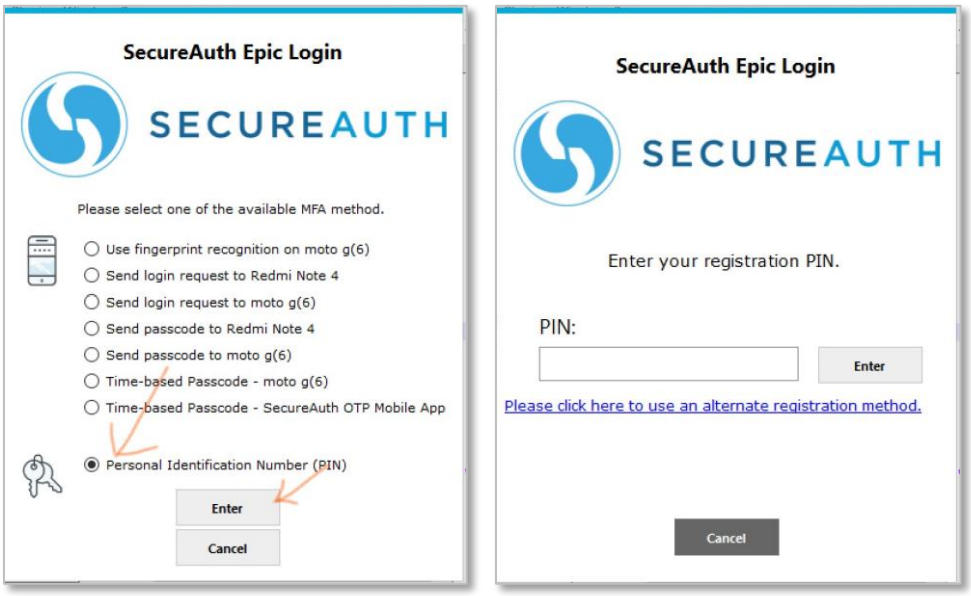
Send passcode to mobile app: When a physician selects send passcode to mobile app, they will have to enter a passcode sent to mobile app in the next screen and press **Enter**.



- **Time-based passcode (TOTP)**: When a physician selects time-based passcode, they will have to enter a timebased passcode in the next screen and press **Enter**.



Personal Identification Number (PIN): When a physician selects personal identification number (PIN), they will have to enter their PIN in the next screen and click **Enter**.



Soft-token EPCS with second factor

The following example illustrates a commonly deployed 2FA method using mobile devices featuring FIPS 140-2compliant one-time passcode (OTP) tokens. This method is easy for physicians to use, and because of its FIPS 140-2 compliance, it meets the DEA's requirement for 2FA.

In this example:

1. The physician enters a user name and is prompted for a method to receive the required passcode The physician receives the passcode on their registered device.
2. The physician enters the received passcode and presses **Enter**.
3. The physician is granted access to the EPCS system.


SecureAuth Epic Login
SECUREAUTH
Type your Passcode and press Enter
User name:
SecureAuthUser
Passcode:
Enter
Cancel

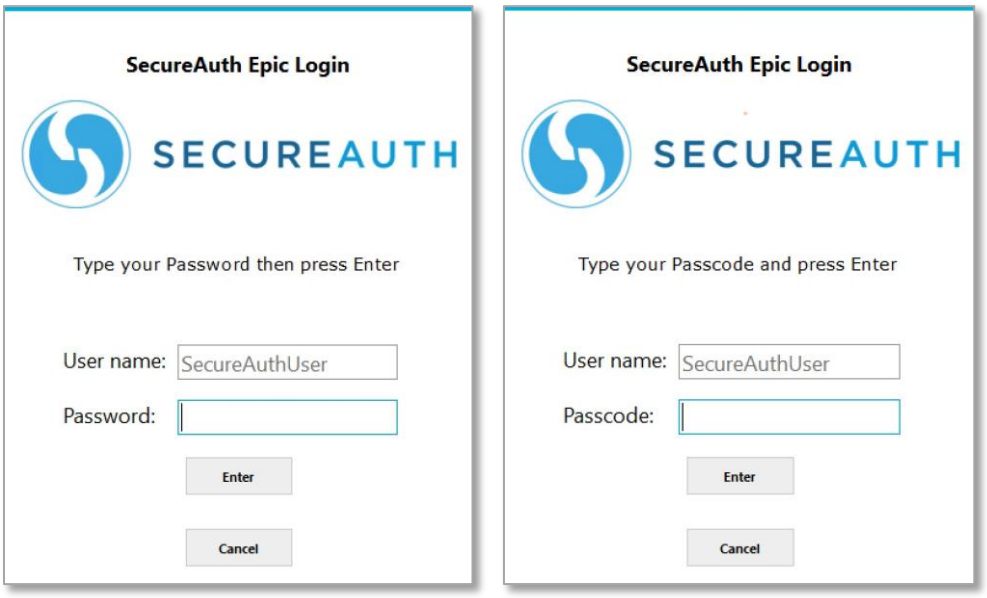
Soft-token EPCS with password

In a variation on the first example, the process requires the physician to perform soft-token EPCS second factor as shown in the following illustration examples (using both a password and a passcode).

The physician uses their mobile application with fingerprint ID (if supported by FIPS 140-2) to unlock their onetime passcode.

1. The physician enters a user name and is prompted for a fingerprint.
2. The physician authenticates with a fingerprint on the mobile device.

Note: Reading a fingerprint is compliant with the FIPS 140-2 standard and works only on mobile devices that support fingerprint reading.



The physician is granted access to EPCS data.

Push-to-Accept EPCS

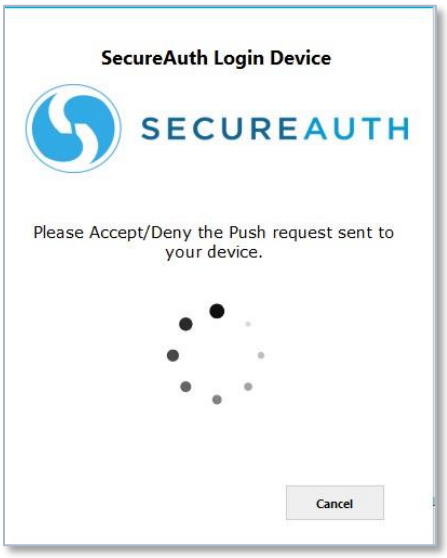
To comply with DEA requirements while providing the quickest possible access, the Identity Platform also features push-to-accept with TouchID for EPCS authentication. Many providers are moving to push-to-accept with TouchID because it not only reduces the number of clicks and character entries a physician must perform, but also incorporates all three of the authentication factors identified by the DEA:

- Something you *know* (username/password).
- Something you *have* (mobile device).
- Something you *are* (fingerprint for TouchID).

An example of the process required to perform push-to-accept EPCS 2FA is shown in the following illustration example:

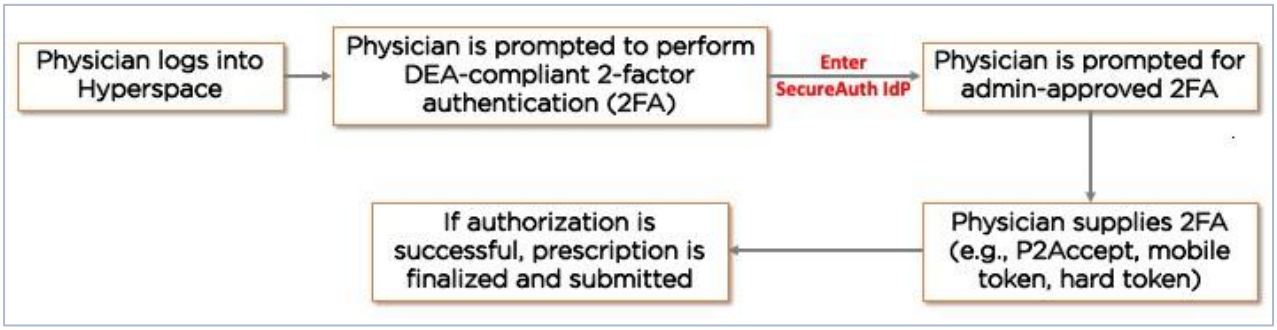
1. The physician opens the application on their mobile device or computer.

2. The application prompts the physician for authentication. At the same time, the secondary device is activated.
3. The physician pushes a button on the secondary device to accept the request for entry.
4. The application then allows physician access to the required data.



Note: If Push-to-Accept is used with accounts that have multiple registered mobile devices, a page appears with a list of mobile devices from which the user can select, as shown below.

The process flow using the Epic EPCS VAM is shown below.



Configuration tasks to support this VAM

This section covers the required configurations to integrate this VAM with the Identity Platform.

Prerequisites

This document is based on the development of the Epic EPCS VAM using the following systems:

- Epic EPCS version 8.4 and 8.7 installed and running on Windows
- Identity Platform version 9.2 or later

Internal developer tested environments

SecureAuth VAM team developed a test tool for testing and validating the result internally, EPIC Hyperspace was not available internally.

Configure the Identity Platform

Configuring the Identity Platform for use with Epic EPCS involves the creation of a realm dedicated to handling the necessary API instructions.

Note: Configuring the Identity Platform for use with Epic EPCS should be handled, at least initially, by the SecureAuth deployment staff and should not be the client's responsibility.

1. Designate a realm in your Identity Platform appliance to provide API access to the SecureAuth Epic EPCS VAM.

   For more information about creating a new realm, see SecureAuth IdP Realm Guide.

2. Go to the **Data** tab and configure the required fields for a data store integration.

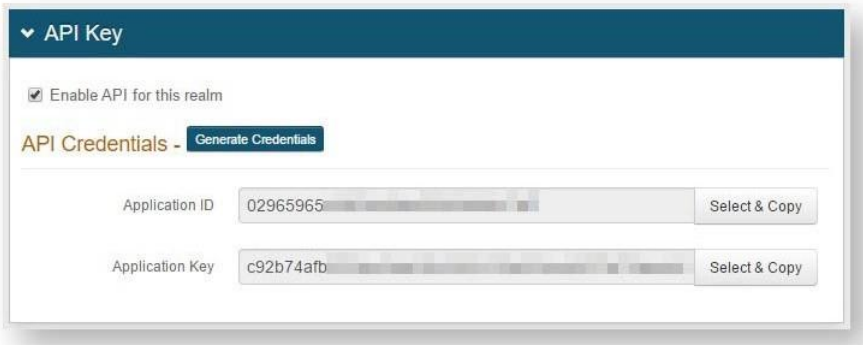
   A data store integration is required for the Identity Platform to pull user profile information during the login process. For more information about configuring the **Data** tab, see Data Tab Configuration.

3. Select the **API** tab.

4. In the **API Key** section do the following:

   a. Select the **Enable API Key for this realm** check box.

   b. Click **Generate Credentials** to create a new Application ID and Application Key, which are unique to this realm.



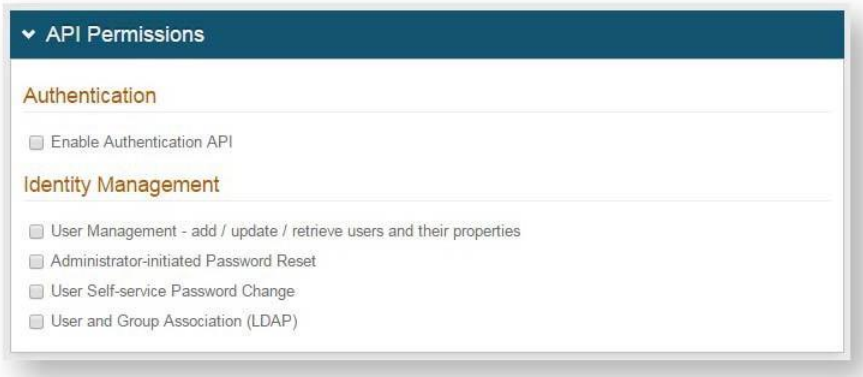
Note: The API key looks as if it consists of 64 random characters, but it is actually composed of 32 *two*character hexadecimal values. This is important when using the API key to produce the required HMAC hash.

c. Copy and paste each of the credentials into a text editor.

These values will be required in the HTTP Header configuration.

5. In the **API Permissions** section, do the following:

a. Select the **Enable Authentication API** check box.



b. If required, select any of the **Identity Management** tools to include in the API as explained in the following table.

Option	Description
User management – add, update, and retrieve users and their properties	Use this tool to add new user profiles, retrieve and update existing user profiles. Updating a user profile includes setting and/or clearing property values in the user profile.
Administrator-initiated password reset	Use this tool to allow an administrator to send a new password to the end user when requested through an application. Use case scenario: End user forgets their password to an application and requests a new password.

User self-service password change	Use this tool to allow end users to enter their current or temporary password and create a new password. Use case scenario: Used in conjunction with the administrator-initiated password reset option – end user enters a current password sent by the administrator, and then enters a new password.
User & group association (LDAP)	Use this tool to enable associations between existing users and groups within the LDAP data store.

6. **Save** your changes.

Install and configure Epic EPCS VAM

1. Copy the provided ZIP file to a location on the system running Hyperspace.
2. Do one of the following:
   - If running the Hyperspace thick client from a workstation, install the **SALoginDevice** on the workstation itself, for example Win10.
   - If using a VDI server, such as Citrix, for access to Hyperspace, install the **SALoginDevice** on all Citrix servers.
3. Extract the SecureAuth folder to the C:\ drive. This folder can be installed in another location if write access is provided to the logs folder.
4. Open the SecureAuth folder and run **RegisterSALoginDevice.bat** as an Administrator.

   This adds **SecureAuthLoginDevice.dll** to the system code base.

   Note: By executing this .bat file, the SecureAuth Epic EPCS VAM is automatically registered, enabling you to bypass regsvr32 for this DLL.

5. Open **SASettings.xml** using a text editor and modify the following settings:

Setting	Description
SecureAuthAPIUrl	The URL used to access the Authentication API realm defined in the SecureAuth configuration. It must be accessible by HTTPS and the certificate used to serve the SSL connection must be trusted by the Epic Hyperspace server.
EPCSUrl	References the local host running Epic Hyperspace. Do not change this URL.
AppID	The Application ID provided in the Identity Platform appliance configuration for the Authentication API.
AppKey	The Application Key provided in the Identity Platform appliance configuration for the Authentication API.

Retry	Indicates the number of failed attempts by the user to enter a correct OTP before the Epic EPCS VAM returns to Epic and fails to authenticate the message.
LogLevel	Default value set to **0**, which turns off logging. Do not change this value unless instructed to do so by a SecureAuth Engineer.
WindowTitle	Indicates the title text that displays on the Epic EPCS VAM dialog box.
RequirePassword	When set to **1**, the user is required to enter their password for the configured data store in the Authentication API realm when using the Epic EPCS VAM.
LogoPath	Specifies the full file path on the appliance for the custom logo image displayed in the Epic EPCS VAM. For example, C:\SecureAuth\logo.png

Setting	**Description**
EnablePush	Enables Push-to-Accept as the second factor method provided by the Epic EPCS VAM. Valid values: ■ **0** – Default value, which disables push-to-accept. ■ **1** – Enables this setting. The following scenarios can occur: • When the Push-to-Accept method is used with accounts that have multiple registered mobile devices, the software displays a list of devices from which the user can select. • When no devices are found, the software notifies the user • When only one device is found, it automatically sends the push request without waiting for a selection.
PreferStaticPin	Enables or disables the static PIN feature. Valid values: ■ **0** – Indicates the use of TOTP (OATH) token ■ **1** – Enables the use of a static PIN. When this feature is enabled, it verifies against the static PIN. When no PIN is found for the user, it defaults to using a standard TOTP token.
PreventNonMobileDevices	Skip Oath/TOTP/PUSH authentication for non-mobile devices: ■ 0 – Enable authentications for all devices. ■ 1 – Disable authentication for Windows/Mac devices. That's means authentication will be skipped for Windows/Mac devices.
ShowMFA	Indicate whether to show the MFA workflow to users. Valid values: ■ **0** – Disable the MFA workflow ■ **1** – Enable the MFA workflow

MFAOrder	This is a comma separated MFA option value to decide the order of MFA options shown on the user interface. The default order is PUSH,TOTP,PIN.
BioLogoPath	Specify the file path for the custom logo image displayed for biometric options list. For example: C:\SecureAuth\ bioLogo.png
PinLogoPath	Specify the file path for the custom logo image displayed for PIN options list. For example: C:\SecureAuth\ pinLogo.png
NoMFAError	Error message to show to user when there is no MFA option available.
NoMFASelected	Error message to display to user when no MFA option selected.
AcceptDenyPushRequest	Text to display once user selects biometric/push request option.
APIConnectionError	Text to display if there is an error while validating the input.
APIDeniedError	Text to display if biometric/push request is denied.
APIFailedError	Text to display if biometric/push request has failed.
APINotFoundError	Text to display if biometric/push request is not found.
APIInvalidError	Text to display if biometric/push request is invalid.
APIUnknownError	Text to display if there is some unknown error.
EnterPasscode	Text prompting user to provide passcode.
EnterTOTP	Text prompting user to provide TOTP.
EnterPIN	Text prompting user to provide PIN.
Setting	**Description**
UnknownMFAError	Text prompting user to select another authentication method (in case of current selection option is not working for any reason).
PasscodeNotMatchError	Text to display when the passcode the user provides does not match.
SelectedDeviceError	Text to display when there is an issue with the selected device.
TOTPNotMatchError	Text to display when the TOTP the user provides does not match.
BlankPasscode	Text to display when the user provides a blank passcode.
BlankTOTP	Text to display when the user provides a blank TOTP.
BlankPIN	Text to display when the user provides a blank PIN.

6. Please refer below sample of SASettings.xml file.

```
<?xml version="1.0" encoding="utf-8" ?>
<SASettings>
    <sASetting EPICVersion="Hyperdrive" />
    <sASetting WindowTitle="SecureAuth Epic Login" />
    <sASetting SecureAuthAPIUrl="https://<HOST>/SecureAuth17" />
    <sASetting EPCSUrl="https://<HOST>/SecureAuth17/secureauth.aspx" />
    <sASetting AppID="XXXXXXXXXXXXXXXXXXXXXXXXXXXX" />
    <sASetting AppKey="XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX" />
    <sASetting Retry="3" />
    <sASetting LogLevel="2" />
    <sASetting EnablePush="0" />
    <sASetting LogoPath="logo.png" />
    <sASetting RequirePassword="0" />
    <sASetting PreferStaticPin="0" />
    <sASetting PreventNonMobileDevices="0" />
    <sASetting ShowMFA="1" />
    <sASetting MFAOrder="PUSH,TOTP,PIN" />
    <sASetting BioLogoPath="bioLogo.png" />
    <sASetting PinLogoPath="pinLogo.png" />
    <sASetting NoMFAError="Please select one of the available MFA method."
    <sASetting NoMFASelected="Please select a MFA." />
    <sASetting AcceptDenyPushRequest="Please Accept/Deny the push request
    <sASetting APIConnectionError="Unable to validate or send {reqType} pu
    <sASetting APIDeniedError="Push Request Denied." />
    <sASetting APIFailedError="Push Request Failed." />
    <sASetting APINotFoundError="Push Request Not Found." />
    <sASetting APIInvalidError="Push Request Invalid." />
    <sASetting APIUnknownError="Push Request Unknown Error." />
    <sASetting EnterPasscode="Enter Passcode sent to your device." />
    <sASetting EnterTOTP="Enter the code from your SecureAuth OTP Mobile A
    <sASetting EnterPIN="Enter your registration PIN." />
    <sASetting UnknownMFAError="Some error occurs! Please select another M
    <sASetting PasscodeNotMatchError="Provided passcode does not match." /
    <sASetting SelectedDeviceError="Selected device is not a mobile device
    <sASetting TOTPNotMatchError="Provided TOTP does not match." />
    <sASetting BlankPasscode="Please enter the passcode sent to your devic
    <sASetting BlankTOTP="Please enter the code from your SecureAuth OTP M
    <sASetting BlankPIN=" Please enter your registration PIN." />
</SASettings>
```

7. **Save** the SASettings.xml file.

Create the SecureAuth login device

1. In Chronicles, access the **Authentication Devices (E0G)** master file, then go to <Data Management> **1. Enter Data | Create/Edit Device <Y>**.

2. Enter a **name** for the device. For example, SecureAuthLoginDevice.
3. Enter a new **ID**. For example, 10001+.
4. On the **General Settings** page, set the following:
   - Set the Description to **SecureAuthLoginDevice**.
   - Set the Platform to **1-Desktop**.
5. On the **Desktop Settings** page in the **ProgID** field, enter **SecureAuthLoginDevice.Receiver**.

Configure SecureAuthLoginDevice as the secondary authentication device

Once the SecureAuthLoginDevice is created, you will need to specify it as the secondary authentication device.

1. Open Hyperspace and go to **Epic | Admin | Access Management | Authentication Administration**.
2. Select the **System** level.
3. Select the desired **Context**.
4. Set the first authentication method as the **Primary Device**.

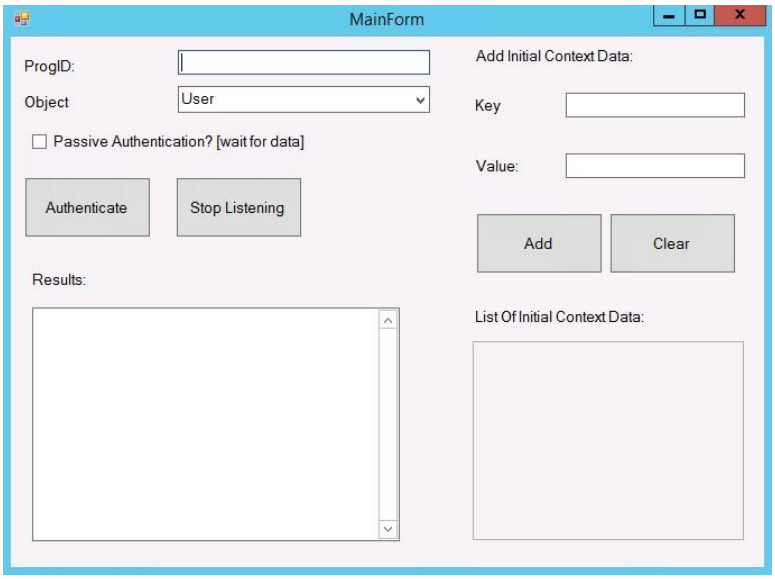
   Typically, this is the username and password.
5. Set the SecureAuthLoginDevice as the **Secondary Device**.
6. Click **Accept**.

Test the Epic EPCS VAM

Epic provides a standalone .NET Testing tool that can be used to verify that the Epic EPCS VAM is working before adding it to the Hyperspace configuration. The steps below outline how to use the tool.

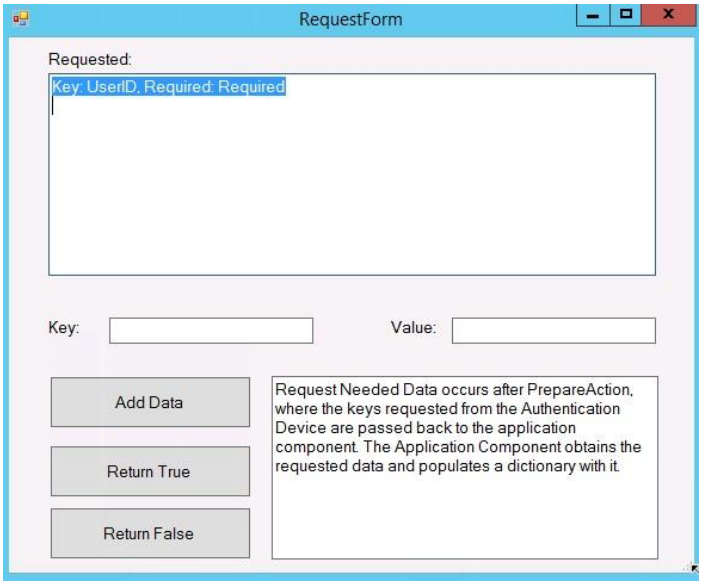
1. From the SecureAuth folder, open the **Test** folder.
2. Run the **StandAloneNETTester.exe** file.

   The **MainForm** window appears.



3. In the **ProgID** field, enter **SecureAuthLoginDevice.Receiver.**
4. Click **Authenticate**.

   The RequestForm window appears.



5. Do the following:
   a. In the **Key** field, enter the **UserID**.
   b. In the **Value** field, enter the **username** to be tested.
6. Click **Add Data**.
7. Click **Return True**.

The **Epic EPCS VAM** appears. If the device completes successfully, the **Results** field on the **MainForm** window updates with a success; otherwise the field displays **Perform Action Failed**.

Upgrade information

Before upgrading SecureAuth software, open a Support ticket. The process of upgrading to a newer SecureAuth software version might cause the SecureAuth VAM to become invalid and stop working. When your site is ready to upgrade SecureAuth software, get started by creating a support ticket and selecting **I have a question or issue regarding SecureAuth Value-Added Modules (VAMs)** from the "Submit a request" list. A SecureAuth Tailoring engineer will contact you to evaluate and ensure that the VAM will work with updated SecureAuth software.