Password Reset Page Configuration Guide

Introduction

The Password Reset Page is for end-users to reset their passwords securely, without any help desk assistance.

Self-service Password Reset can be achieved using various 2-Factor Authentication workflows.

Prerequisites

1. Create a **New Realm** for the Password Reset Page

2. The SecureAuth IdP directory Service Account must have the write privileges to **modify** in order to change user passwords

3. If using **Active Directory**, the following **Outbound Ports** must be open for password reset:

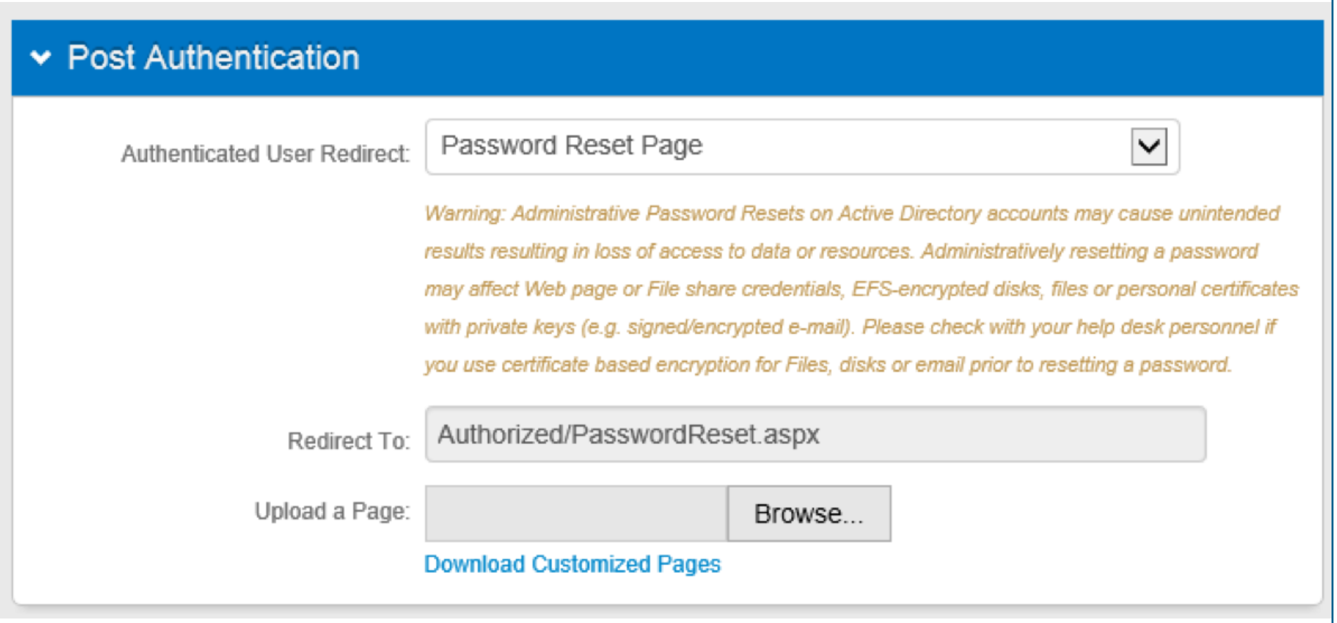
- **139** – DFSN, NetBIOS Session Service, NetLogon
- **445** – SMB/CIFS, DFSN, LSARPC, NbtSS, NetLogonR, SamR, SrvSvc
- **464** – Kerberos Change\Set Password

4. Configure the following tabs in the Web Admin before configuring the **Post Authentication** tab:

- **Overview** – the description of the realm and SMTP connections must be defined
- **Data** – an enterprise directory must be integrated with SecureAuth IdP
- **Workflow** – the way in which users will access this application must be defined
- **Registration Methods** – the 2-Factor Authentication methods that will be used to access this page (if any) must be defined

Configuration Steps

Post Authentication



Post Authentication

Authenticated User Redirect: Password Reset Page

Warning: Administrative Password Resets on Active Directory accounts may cause unintended results resulting in loss of access to data or resources. Administratively resetting a password may affect Web page or File share credentials, EFS-encrypted disks, files or personal certificates with private keys (e.g. signed/encrypted e-mail). Please check with your help desk personnel if you use certificate based encryption for Files, disks or email prior to resetting a password.

Redirect To: Authorized/PasswordReset.aspx

Upload a Page:

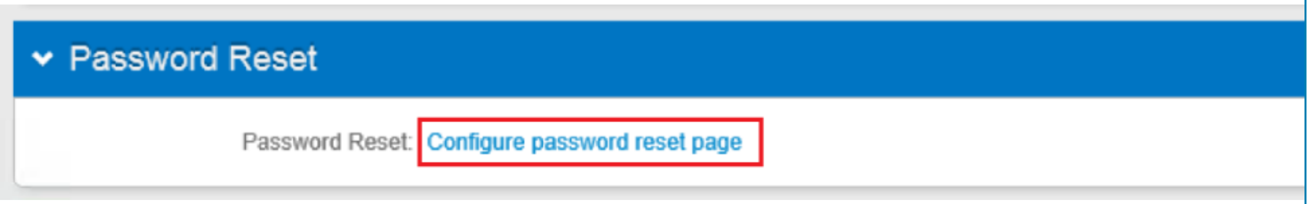
1. Select **Password Reset Page** from the **Authenticated User Redirect** dropdown in the **Post Authentication** tab in the Web Admin

2. An unalterable URL will be auto-populated in the **Redirect To** field, which will append to the domain name and realm number in the address bar (Authorized/PasswordReset.aspx)

3. A customized post authentication page can be uploaded, but it is not required

Click **Save** once the configurations have been completed and before leaving the **Post Authentication** page to avoid losing changes

Password Reset

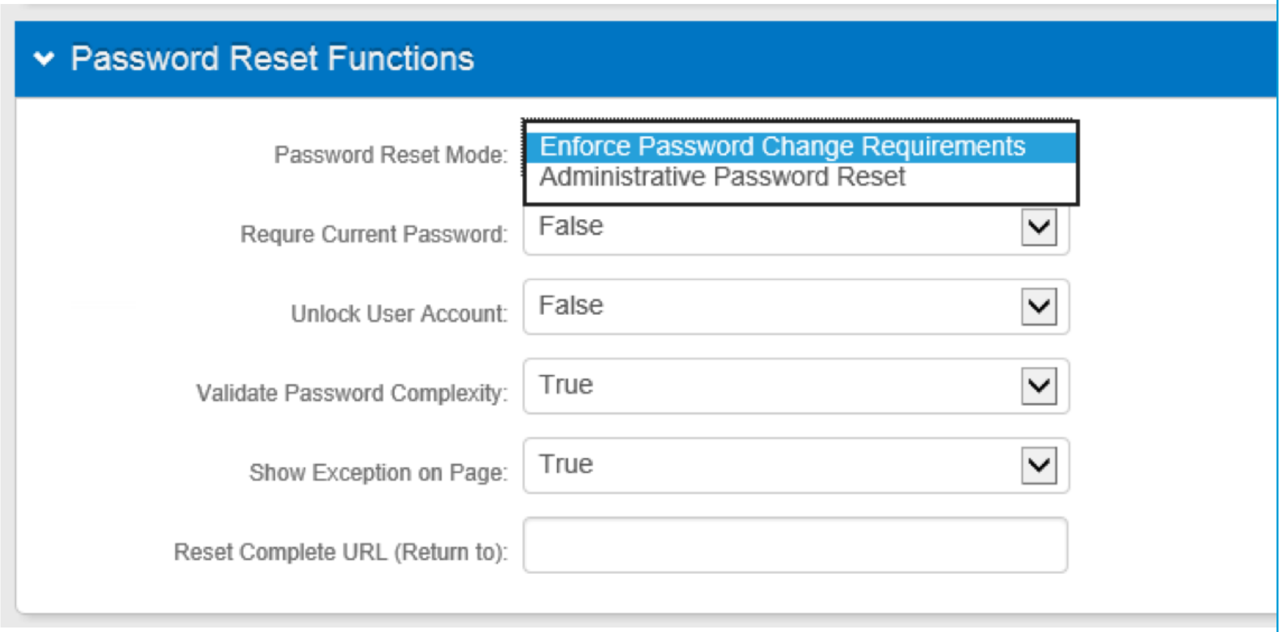


Password Reset

Password Reset: Configure password reset page

4. Click **Configure password reset page** to design the password reset settings

Password Reset Functions



Password Reset Functions

Password Reset Mode: Enforce Password Change Requirements / Administrative Password Reset

Requre Current Password: False

Unlock User Account: False

Validate Password Complexity: True

Show Exception on Page: True

Reset Complete URL (Return to):

5. Select **Enforce Password Change Requirements** or **Administrative Password Reset** from the **Password Reset Mode** dropdown

6. Select **True** from the **Require Current Password** dropdown to require the current password to reset it to a new one

Selecting **False** will not require the current password and the user will only be required to go through the designated **Workflow**

7. Select **True** to **Unlock User Account** along with password reset

Selecting **False** will require administrative action to **Unlock User Account**

8. Select **True** from the **Validate Password Complexity** to require certain complexities for password reset, which can be configured in the **Password Complexity** section

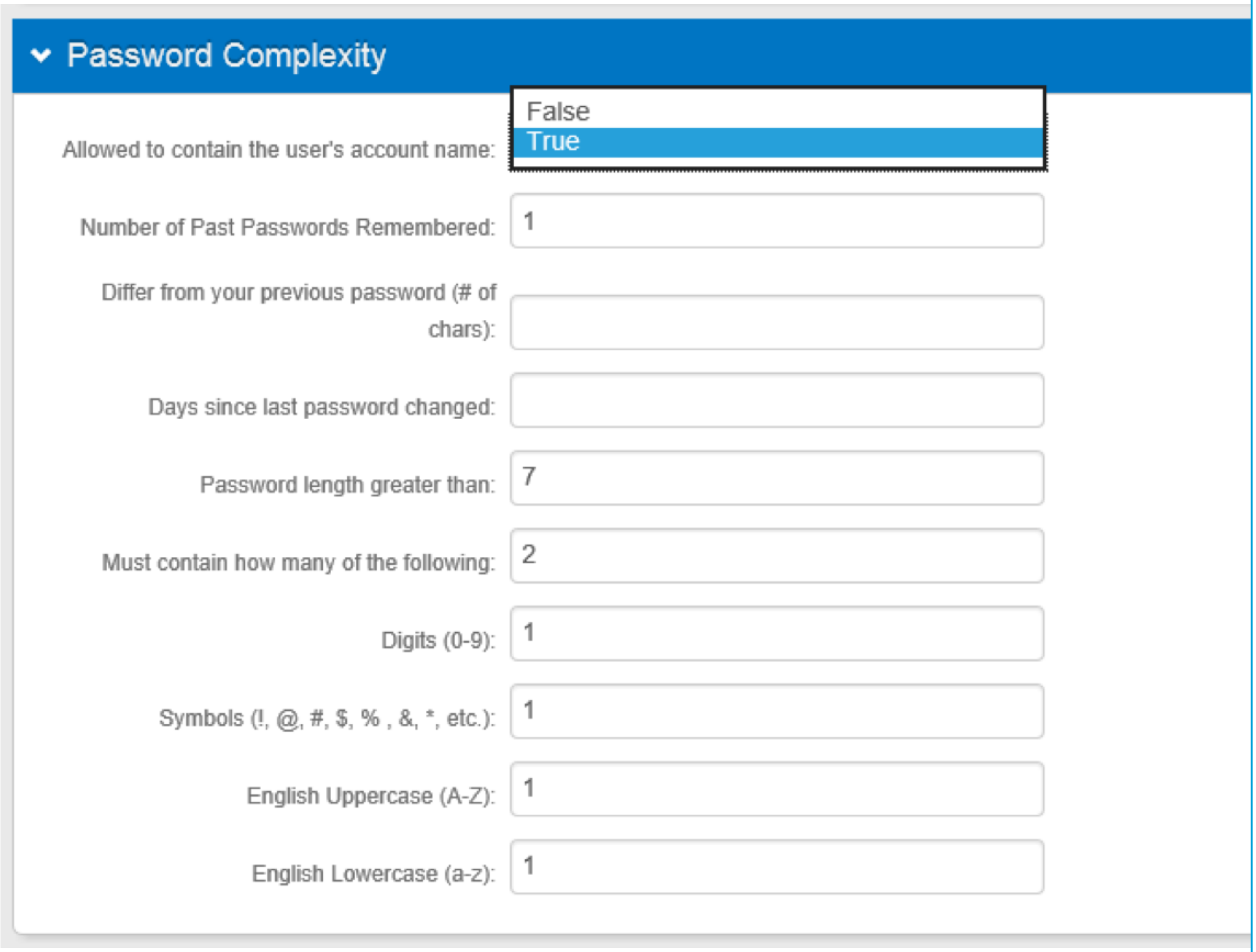
Selecting **False** will allow users to reset their passwords to any password format

9. Select **True** from the **Show Exception on Page** dropdown to notify users as to why their new password is not acceptable (if **True** is selected above)

Selecting **False** will deny the password if it is not acceptable, but will not provide the reason

10. Set a **Reset Complete URL** to where users will be directed once their passwords have been successfully reset (not required)

Password Complexity



Password Complexity

Allowed to contain the user's account name:	False True
Number of Past Passwords Remembered:	1
Differ from your previous password (# of chars):	
Days since last password changed:	
Password length greater than:	7
Must contain how many of the following:	2
Digits (0-9):	1
Symbols (!, @, #, $, % , &, *, etc.):	1
English Uppercase (A-Z):	1
English Lowercase (a-z):	1

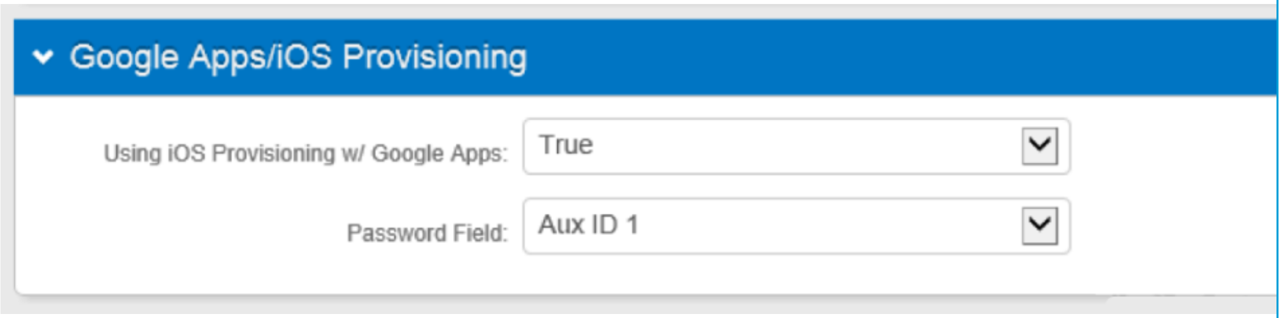
11. Select **True** from the **Allowed to contain the user's account name** dropdown if the new password can contain the username

12. Configure the **Password Complexities** as preferred

If **False** is selected in the **Validate Password Complexity** field in the **Password Functions** section, then no configurations are required

If **True** is selected in the **Validate Password Complexity** field in the **Password Functions** section, then the configurations set here will be enforced

Google Apps / iOS Provisioning



Google Apps/iOS Provisioning

Using iOS Provisioning w/ Google Apps: True

Password Field: Aux ID 1

13. Select **True** from the **Using iOS Provisioning w/ Google Apps** dropdown if the password synchronization functions are enabled

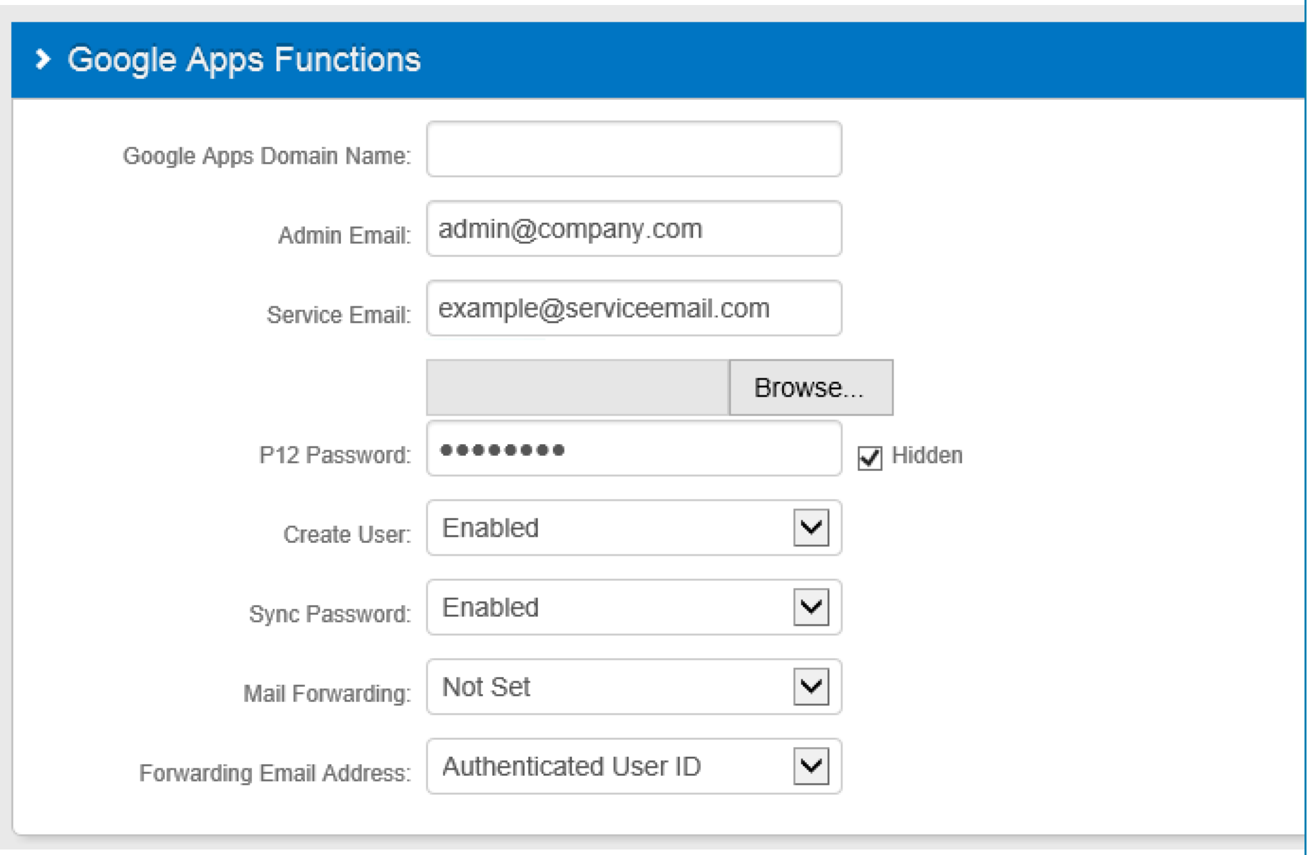
    More configuration steps are required in the **Google Apps Functions** in the **Post Authentication** tab

14. Select the **Profile Field** from the **Password Field** dropdown that corresponds to the password synchronization function

    Refer to **iOS G Suite Provision Configuration Guide** for more information

Click **Save** once the configurations have been completed and before leaving the **Password Reset Settings** page to avoid losing changes

Google Apps Functions



Google Apps Functions

Google Apps Domain Name:

Admin Email: admin@company.com

Service Email: example@serviceemail.com

P12 Password: ●●●●●●●● ☑ Hidden

Create User: Enabled

Sync Password: Enabled

Mail Forwarding: Not Set

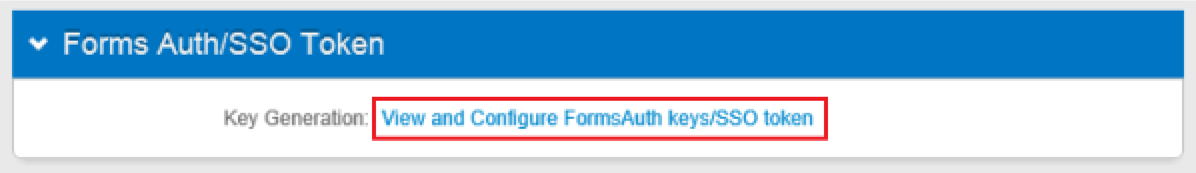
Forwarding Email Address: Authenticated User ID

(OPTIONAL) 15. Configure the realm for Google Apps provisioning, including password synchronization

Refer to **Google Apps Provisioning** for more information

Click **Save** once the configurations have been completed and before leaving the **Post Authentication** page to avoid losing changes

Forms Auth / SSO Token



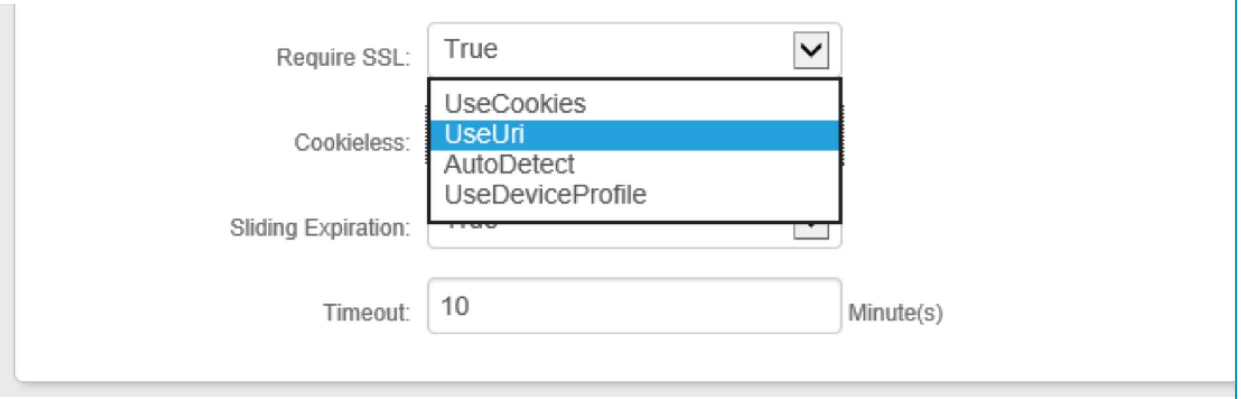
Forms Auth/SSO Token

Key Generation: View and Configure FormsAuth keys/SSO token

16. Click **View and Configure FormsAuth keys / SSO token** to configure the token/cookie settings and to configure this realm for Single Sign-on (SSO)

These are *optional* configurations

Forms Authentication



Require SSL: True

Cookieless: UseCookies / UseUri / AutoDetect / UseDeviceProfile

Sliding Expiration:

Timeout: 10 Minute(s)

1. If SSL is required to view the token, select **True** from the **Require SSL** dropdown

2. Choose whether SecureAuth IdP will deliver the token in a cookie to the user's browser or device:

- **UseCookies** enables SecureAuth IdP to always deliver a cookie
- **UseUri** disables SecureAuth IdP to deliver a cookie, and instead deliver the token in a query string
- **AutoDetect** enables SecureAuth IdP to deliver a cookie if the user's settings allow it
- **UseDeviceProfile** enables SecureAuth IdP to deliver a cookie if the browser's settings allow it, no matter the user's settings

3. Set the **Sliding Expiration** to **True** if the cookie remains valid as long as the user is interacting with the page

4. Set the **Timeout** length to determine for how many minutes a cookie is valid

No configuration is required for the **Name**, **Login URL**, or **Domain** fields

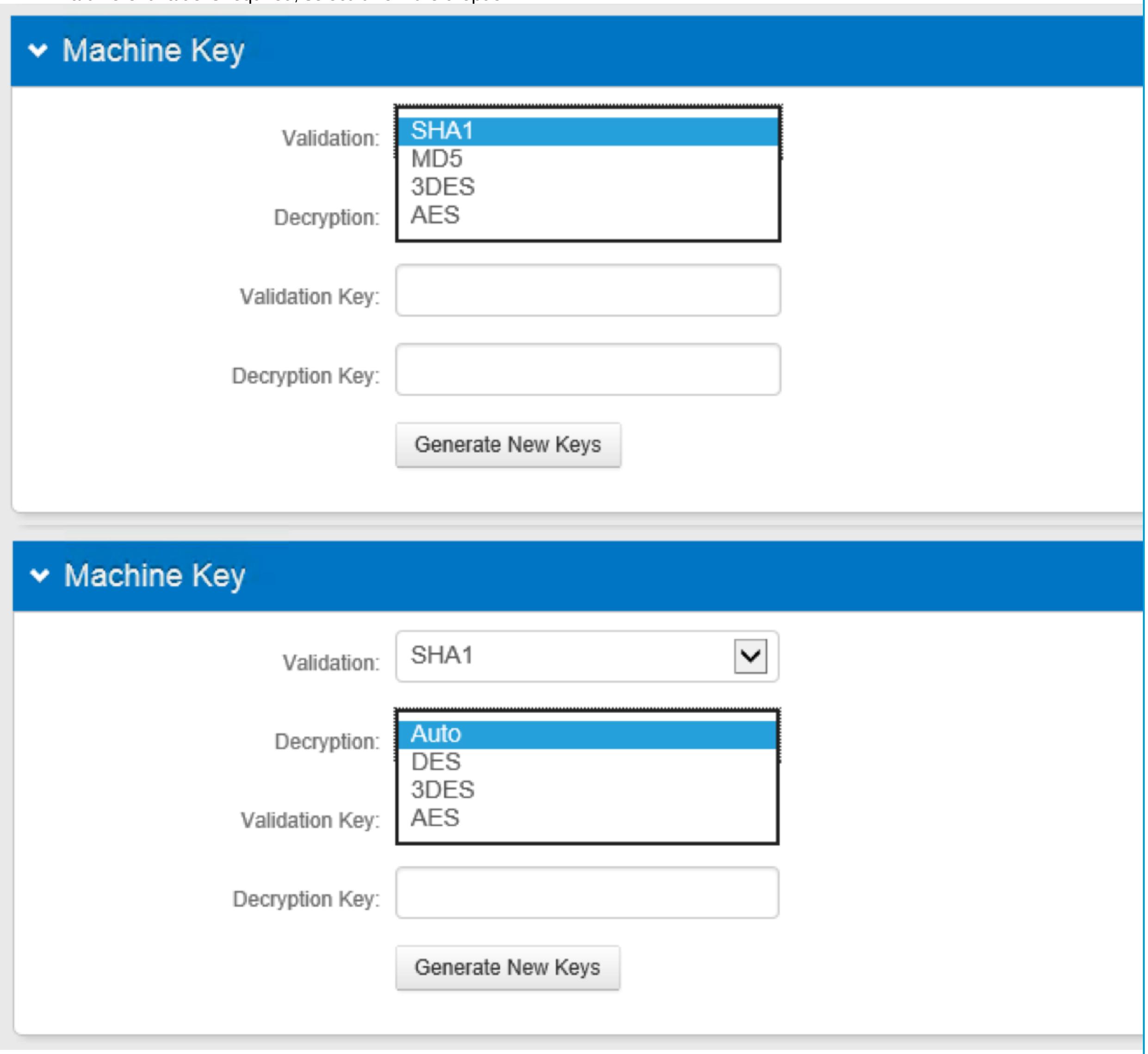
Machine Key

5. No changes are required in the **Validation** field, unless the default value does not match the company's requirement

    If a different value is required, select it from the dropdown

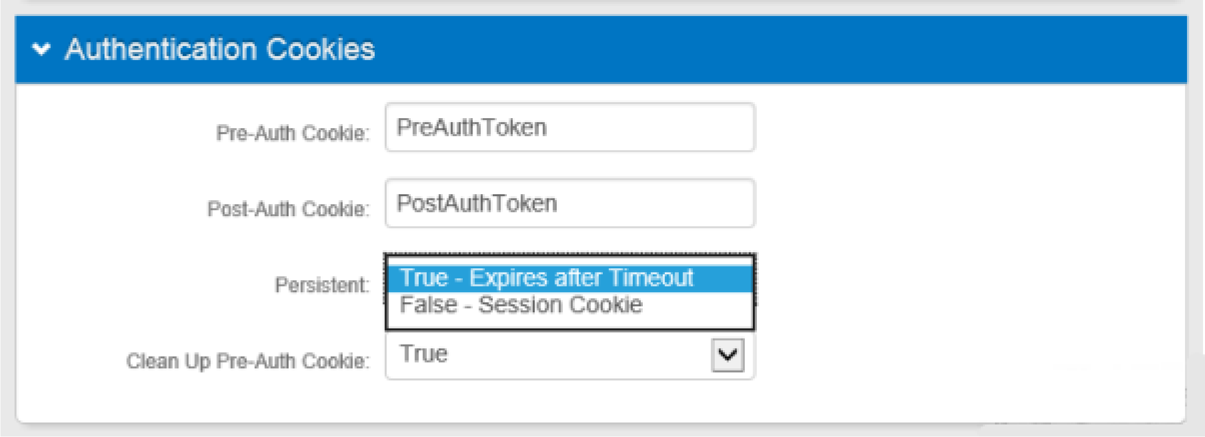
6. No changes are required in the **Decryption** field, unless the default value does not match the company's requirement

    If a different value is required, select it from the dropdown



ⓘ No configuration is required for the **Validation Key** or **Decryption Key** fields

Authentication Cookies



Authentication Cookies

Pre-Auth Cookie: PreAuthToken

Post-Auth Cookie: PostAuthToken

Persistent: True - Expires after Timeout

Clean Up Pre-Auth Cookie: True

7. Enable the cookie to be **Persistent** by selecting **True - Expires after Timeout** from the dropdown

Selecting **False - Session Cookie** enables the cookie to be valid as long as the session is open, and will expire once the browser is closed or the session expires

ⓘ No configuration is required for the **Pre-Auth Cookie**, **Post-Auth Cookie**, or the **Clean Up Pre-Auth Cookie** fields

ⓘ Click **Save** once the configurations have been completed and before leaving the **Forms Auth / SSO Token** page to avoid losing changes

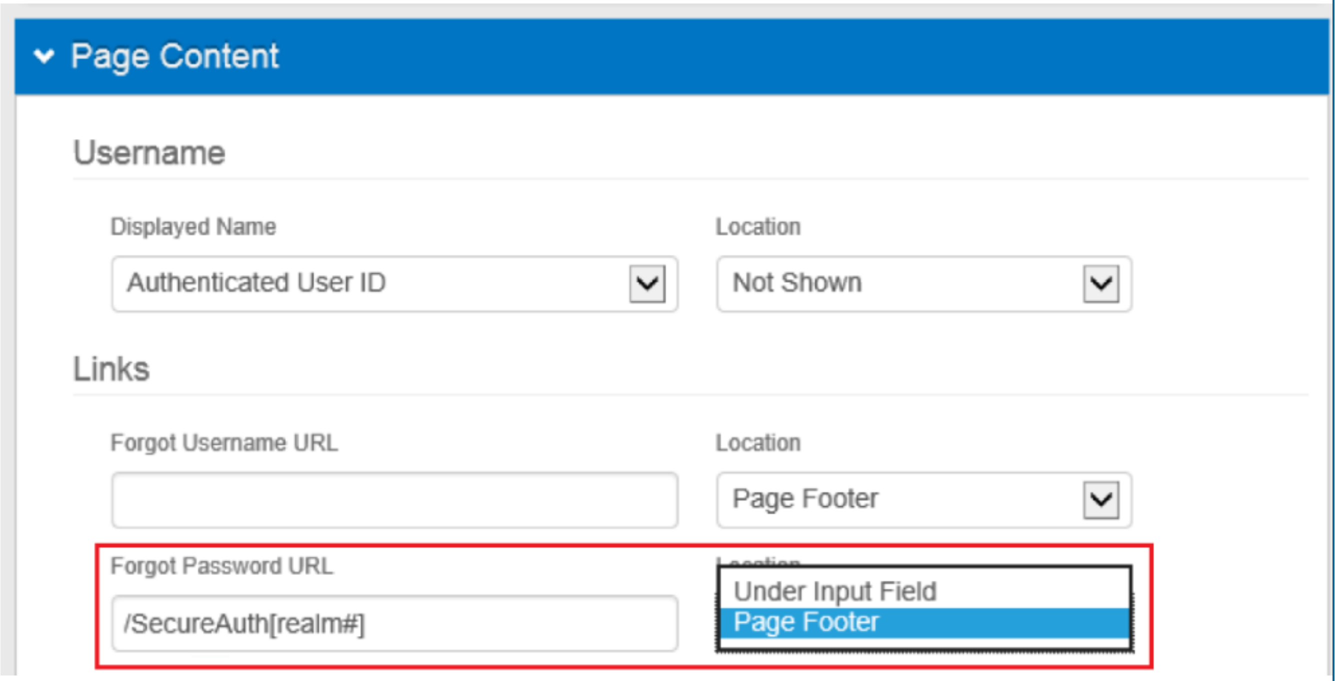
ⓘ To configure this realm for SSO, refer to **SecureAuth IdP Single Sign-on Configuration**

ⓘ To configure this realm for *Windows Desktop SSO*, refer to **Windows Desktop SSO Configuration Guide**

OPTIONAL Configuration Steps

The following configuration step is for other SecureAuth IdP realms, and **not** the **Password Reset Page** realm

Overview



Page Content

Username

Displayed Name: Authenticated User ID

Location: Not Shown

Links

Forgot Username URL:

Location: Page Footer

Forgot Password URL: /SecureAuth[realm#]

Location: Under Input Field / Page Footer

Update the **Password Reset URL** field and its **Location** on the login page within the *other* SecureAuth IdP realms in which the function is available

The URL would be: **/SecureAuth[PasswordResetRealm#]**

Related Documentation

- **Configure the Password Reset Email Notification**