Workflow Realm Settings Endpoint

Introduction

Use the /workflow PATCH endpoint to dictate the end-user login process, configure Device Recognition, enable redirects, customize token settings.

Prerequisites

1. Complete the Enablement and Header Steps in the **Admin API Guide**
2. Have access to the application code that calls to the API endpoint(s)
3. Integrate a membership and profile directory(s) with SecureAuth IdP (**Data Realm Settings Endpoint**)

/workflow Endpoint

The following endpoints are prepended with the URL, **https://<SecureAuth IdP Domain>/api/v1/realms/<realm ID>**, if running **SecureAuth IdP v9.1** – in which **realm ID** is the ID number of the realm to configure –

or **https://<SecureAuth IdP Domain>/api/v2/realms/<realm ID>**, if running **SecureAuth IdP v9.2 or later**

Workflow Settings /workflow PATCH Endpoint

Use this endpoint to configure the realm's workflow settings, including client-side login process, device recognition, token preferences, and user redirects.

HTTP Method	Endpoint	Example	SecureAuth IdP version
PATCH	`/workflow`	`https://secureauth.company.com/api/v1/realms/26/workflow`	v9.1
PATCH	`/workflow`	`https://secureauth.company.com/api/v2/realms/26/workflow`	v9.2 or later

Field Definitions and Accepted Values for Configuration

Defaulted values in **bold**

Field	Description	Accepted Values	Note
deviceRecognitionMethod	Settings for persistent token	N / A	
integrationMethod	Device limitation and functionality of client	**CertificationEnrollmentAndValidation**	Only one option supported
clientSideControl	Credential (persistent token) used in the workflow	**DeviceBrowserFingerprinting**	Only one option supported
browserProfileSetting	Settings for Device Recognition browser profiles	N / A	
fpMode	Deliver cookie to browser to compare with browser profile	• **NoCookie** • Cookie	For browser profile
	Deliver cookie to mobile device or use Device Recognition mobile app to compare with mobile profile	• **Cookie** • MobileApp	For mobile profile
cookieNamePrefix	Name prepended to cookie name	any	Full cookie name: cookieNamePrefix + company name + hashed value of user ID For browser and mobile profiles

cookieExpireLength	Number of hours during which cookie is valid	any, numerical	For browser and mobile profiles
matchFpIdInCookie	Require match between profile ID in directory and profile ID of current login	• true • **false**	For browser and mobile profiles
authenticationThreshold	Percentage of current profile score matched against stored profile score required to bypass additional authentication	any, defaulted to **90**	For browser and mobile profiles
updateThreshold	Percentage of current profile score matched against stored profile score required to update stored profile after successful additional authentication	any, defaulted to **89**	For browser and mobile profiles
mobileProfileSetting	Settings for Device Recognition mobile profiles	N / A	
skipIpMatch	Skip IP address matching between device and stored profile	• **true** • false	
profileSetting	Settings for Device Recognition profiles	N / A	
fpExpirationLength	Number of days during which profile is valid	any, defaulted to **0**	0 or negative: no expiration
fpExpirationSinceLastAccess	Number of days profile is valid since last access	any, defaulted to **0**	0 or negative: no expiration
allowOnlyOneFpCookiePerBrowser	One cookie allowed per browser	• true • **false**	
totalFpMaxCount	Number of Device Recognition profiles allowed per user account at single time	number, defaulted to **-1**	**-1** : no maximum amount
whenExceedingMaxCount	Action to take when exceeding max profile amount	• **Allow** • NotAllow	If **totalFpMaxCount** sets limit
replaceInOrderBy	Method to replace existing profiles with new ones when exceeding max amount	• **CreateTime** • LastAccessTime	If **totalFpMax Count** sets limit and **" whenExceedingMaxCount": "Allow"**
fpAccessRecordsMaxCount	Number of access history records stored per profile	number, defaulted to **5**	
loginScreen	Settings for client-side login pages	N / A	
defaultWorkflow	Workflow for end-user login	• UsernameOnly • Username_SecondFactor • ValidPersistentTokenOnly • UsernameAndPassword • UsernameAndPassword_SecondFactor • Username_Password • **Username_SecondFactor_Password** • ValidPersistentToken_Password • ValidPersistentToken_SecondFactor • ValidPersistentToken_SecondFactor_Password	
publicPrivateMode	Designated mode for end-user login	• **PublicPrivate** • PublicOnly • PrivateOnly	
publicPrivateModeDefault	Default selection on client-side login page	• **Public** • Private • NoDefault	If **" publicPrivateMode": "PublicPrivate"**
rememberPublicPrivateUserSelection	Automatically select end-user's last selected publicPrivateMode option	• **true** • false	

showInlinePasswordChange	Allow end-users to update expired passwords during login	• true • **false**	Requires Web Admin UI configuration
passwordThrottle	Settings for password throttling	N / A	Refer to **Password Throttling Configuration Guide** for more information
enabled	Enable password throttling in realm	• true • **false**	
maxFailedAttempts	Number of failed attempts allowed before action takes place	number, defaulted to **5**	
interval	Number of timeUnit during which failed attempts are counted	number, defaulted to **5**	
timeUnit	Unit of time for interval	• **Minutes** • Hours • Days	
action	Action to take when maxFailedAttempts is reached during interval: timeUnit	• **BlockUserUntilTimeLimitExpires** • LockUserAfterExceedingAttempts	
storageLocation	Property that contains the timestamps and count of failed password attempts	• **AuxID1** • AuxID2 • AuxID3 • AuxID4 • AuxID5 • AuxID6 • AuxID7 • AuxID8 • AuxID9 • AuxID10 • Email1 • Email2 • Email3 • Email4 • Phone1 • Phone2 • Phone3 • Phone4	
sessionTimeout	Settings for browser session during workflow	N / A	
sessionStateName	Name of session state	any, defaulted to **ASP.NET_SessionId<realm ID>**	
idleTimeoutLength	Number of minutes during which end-user must interact with browser before session expires and re-authentication is required	number, defaulted to **10**	
displayTimeoutMessage	Display message when session times out	• **Disabled** • DisplayTimeout • AutoRestart	
tokenPersistence	Settings for persistent token (Device Recognition profiles)	N / A	
validatePersistentToken	Check validity of token	• **true** • false	
renewPersistentToken	Generate new token once previous one is validated	• true • **false**	
redirect	Settings for workflow redirects	N / A	
invalidPersistentTokenRedirect	URL to which end-users are redirected if persistent token is invalid	URL path, /<SecureAuth IdP Realm Name>	/<realm name> supported if realms on same appliance
tokenMissingRedirect	URL to which end-users are redirected if persistent token is missing	URL path, /<SecureAuth IdP Realm Name>	
profileMissingRedirect	URL to which end-users are redirected if profile is missing	URL path, /<SecureAuth IdP Realm Name>, defaulted to **profilemissing.aspx**	

mobileRedirect	SecureAuth IdP realm to which end-users are redirected if on mobile device	realmName, e.g. SecureAuth14	
mobileIdentifiers	Identifiers of mobile devices to enable mobileRedirect	any, defaulted to **ios,iphone,ipad, android,wp7**	
terminationPoint	Settings for load balancer integration	N / A	
clientFqdn	Fully Qualified Domain Name (FQDN) set as client point of termination for SecureAuth IdP validation	FQDN	
sslTerminationCertificate	Trusted SSL certificate for bi-lateral authentication with SecureAuth IdP not acting as termination point	certificate BLOB	Not required if providing sslCertificateAddress
sslCertificateAddress	Load balancer FQDN where SSL connection is terminated	FQDN	Not required if providing sslTerminationCertificate
sslTerminationPoint	FQDN of where sslTerminationCert is terminated to allow SecureAuth IdP to validate information	FQDN	
customIdentityConsumer	Settings for pre-authentication workflow	N / A	
receiveToken	Type of token received by SecureAuth IdP from other site	• **SendTokenOnly** • None • Token • ClearTextQueryString • XORBase64QueryString • SendXORBase64Only • ReceiveTokenOnly	
requireBeginSite	Enable pre-authentication page for workflow	• true • **false**	
beginSite	Type of pre-authentication begin site	• **Custom** • BasicAuthentication • CertificateFinderV1 • CertificateFinderV2 • ClientSideSsl • FingerprintFinder • FormPost • MultiWorkflow • NativeCertificateFinder • WindowsSso • WindowsSsoSkipWorkflow • CiscoIse • YubiKey	Begin sites may require Web Admin UI configuration
windowsSsoUserImpersonation	Run SecureAuth IdP as user or service name when using IWA (Kerberos)	• true • **false**	
windowsSsoWindowsAuthentication	Enable Windows Desktop SSO (Kerberos)	• true • **false**	
yubiKeyProvisionPage	URL of end-user YubiKey provisioning page	URL path	
customBeginSiteUrl	URL of pre-authentication begin site	URL path	If **"beginSite": "Custom"**, otherwise *null*
receiveTokenDataType	Location of user ID in token received by SecureAuth IdP	• **Name** • UserData	

sendTokenDataType	Location of user ID in token sent by SecureAuth IdP	• **UserId** • Password • Phone1 • Phone2 • Phone3 • Phone4 • Email1 • Email2 • Email3 • Email4 • AuxId1 • AuxId2 • AuxId3 • AuxId4 • AuxId5 • AuxId6 • AuxId7 • AuxId8 • AuxId9 • AuxId10 • FirstName • LastName • Custom	
userIdCheck	Check for "Cisco-specific" user ID	• true • **false**	For Cisco ASA integrations only
allowTransparentSso	Enable transparent SSO between associated realms / applications	• true • **false**	
delimiter	XOR delimiter used with shared secret to encrypt user ID	any	
getSharedSecret	Shared secret sent to SecureAuth IdP, provided by SP	number, 1 - 223	
setSharedSecret	Shared secret sent by SecureAuth IdP	number, 1 - 223	
fbaWebService	Settings for FBA Web Service	N / A	
enabled	Enable FBA Web Service	• true • **false**	
username	Username for FBA Web Service communication	any	
password	Password associated to username	any	

Parameters and Response Examples

Parameters

```
{
        "deviceRecognitionMethod": {
                "integrationMethod": "CertificationEnrollmentAndValidation",
                "clientSideControl": null
        },
        "browserProfileSetting": {
                "fpMode": "NoCookie",
                "cookieNamePrefix": "SecureAuthDFP_",
                "cookieExpireLength": 168,
                "matchFpIdInCookie": false,
                "authenticationThreshold": 90,
                "updateThreshold": 89
        },
        "mobileProfileSetting": {
                "fpMode": "Cookie",
                "cookieNamePrefix": "SecureAuthDFP_",
                "cookieExpireLength": 72,
                "matchFpIdInCookie": true,
                "skipIpMatch": true,
                "authenticationThreshold": 100,
                "updateThreshold": 90
        },
```

Success Response

```
{
 "status": "Success",
 "message": []
}
```

```
        "profileSetting": {
                "fpExpirationLength": 0,
                "fpExpirationSinceLastAccess": 0,
                "allowOnlyOneFpCookiePerBrowser": false,
                "totalFpMaxCount": -1,
                "whenExceedingMaxCount": "Allow",
                "replaceInOrderBy": "CreateTime",
                "fpAccessRecordsMaxCount": 5
        },
        "loginScreen": {
                "defaultWorkflow": "Username_SecondFactor_Password",
                "publicPrivateMode": "PublicPrivate",
                "publicPrivateDefault": "Private",
                "rememberPublicPrivateUserSelection": true,
                "showUserIdTextbox": false,
                "showInlinePasswordChange": false
                "passwordThrottle": {
                        "enabled": true,
                        "maxFailedAttempts": 5,
                        "interval": 14,
                        "timeUnit": "Minutes",
                        "action": "LockUserAfterExceedingAttempts",
                        "storageLocation": "AuxID3"
                }
        },
        "sessionTimeout": {
                "sessionStateName": "ASP.NET_SessionId220",
                "idleTimeoutLength": 10,
                "displayTimeoutMessage": "Disabled"
        },
        "tokenPersistence": {
                "validatePersistentToken": true,
                "renewPersistentToken": false
        },
        "redirect": {
                "invalidatePersistentTokenRedirect": "",
                "tokenMissingRedirect": "",
                "profileMissingRedirect": "profilemissing.aspx",
                "mobileRedirect": "",
                "mobileIdentifiers": "ios,iphone,ipad,android,wp7"
        },
        "terminationPoint": {
                "clientFqdn": "",
                "sslTerminationCertificate": "",
                "sslCertificateAddress": "",
                "sslTerminationPoint": ""
        },
        "customIdentityConsumer": {
                "receiveToken": "SendTokenOnly",
                "requireBeginSite": false,
                "beginSite": "Custom",
                "windowsSsoUserImpersonation": false,
                "windowsSsoWindowsAuthentication": false,
                "yubiKeyProvisionPage": "",
                "customBeginSiteUrl": "",
                "receiveTokenDataType": "Name",
                "sendTokenDataType": "UserId",
                "userIdCheck": true,
                "allowTransparentSso": false,
                "delimiter": "",
                "getSharedSecret": 111,
                "setSharedSecret": 111
        },
        "fbaWebService": {
                "enabled": false,
                "username": "",
                "password": ""
        }
}
```

Related Documentation

Workflow Tab Configuration