ProteqsIT AffirmID API and TruYouID Plugin

Table of Contents

AffirmID API

AffirmID is an authenticator App for use on Android and iPhones. As part of its capability an API is provided via which enterprise users can interact remotely with the App for the purpose as disclosed below. As a first API operation the App must be registered by the user. Although this is often an optional feature enterprise clients may wish to make it mandatory. Registration as explained below provides information about the App with inclusion of a private key. This private key is required for encrypting all other API exchanges.

Secure Account Registration

Often authenticator accounts are setup over unsecure networks wherein security if there is any is limited to messaging on the wire using TLS/SSL. While this is good as far as it goes, it does little to secure the messages from dedicated MITM attacks.

Securing the API messaging is an imperative and to that end is addressed by the AffirmID registration process. The process among other things involves AffirmID providing in the initial exchange a public key of a key pair the private key of which is known only to the AffirmID app. This initial message is responded to by the relying party with a response message encrypted using the AffirmID public key. The message includes among other objects a public key whose corresponding private key is known only to the relying party.

Once key registration is complete, all other messages between relying party and AffirmID are encrypted using the respective public keys. In this way even in the unlikely event TLS encryption becomes compromised the messaging encryption remains to secure the exchanges.

As a point of interest, advanced standard protocols such as FIDO2, U2F, and Push are subject to MITM attacks that can circumvent the TLS security these protocols rely on. Settings made available via this API enable addressing that problem by using the API secure messaging schemes.

Disable Account

An API used to disable AffirmID. It is used to disable a specific authenticator account. Once applied the selected account is no longer available to the user. Short of a complete reregistration process, they user no longer has access to the selected account. The registration is also removed from the device backup should there be one. The account to be removed is identified by its Account ID encrypted using the registration public key.

Add, Replace, Remove Certificate

An API used to add, replace, ore remove a certificate. Certificates are stored on a per tenant bases for FIDO, U2F, and Push protocols. Each protocol has a global cert used in all cases where a unique cert has not been provided. An enterprise has the option to add a unique cert used by either of the authenticator protocols supported but only when authenticating for that specific enterprise tenant.

The enterprise can add a new certificate to be used by its account(s), replace an existing certificate, or to remove a certificate. The added certificate takes affect on the next authentication session. A replaced certificate takes affect on the next authentication session. A removed certificate is replaced by the default certificate that is used on the next and all following authentication sessions.

Push Request for Identity Affirmation

There are 3 push authenticators available for identity affirmation purpose.

Generic:

Solicits user intent to authenticate as a 2-button choice, Accept or Decline. Default response is Decline triggered immediately if original user is not in possession or on expiration of the session timeout period. An Accept response indicates the original user is in possession of the device and accepts the authentication request. An OTP code must be received and validated.

FIDO:

A push authenticator using the FIDO CAT2 protocol. A FIDO2 compliant affirmation request for the specified account is sent to the device. The response is Cancel in the event of timeout or in the case of an unknown user in possession. Acceptance result on receipt of a properly formed FIDO2 affirmation response. Refer to FIDO Alliance https:/www. https://fidoalliance.org/.

U2F:

Another push authenticator like FIDO but using U2F. A U2F affirmation request is accepted in which case an expected response is received and validated or canceled in which case identity affirmation has failed. Cancelation can occur because of timeout. The use of U2F protocol over push is unique however in full compliance with U2F standards published by the W3C.

Push Request for Registration

There are 3 push protocol authenticators.

Generic:

Registers the app for used in generic Push authentication. Upon acknowledgement by the user, a blind HOTP account is established. Returned is success for failure. Failure occurs on timeout, on device possession status unknown, or on refusal by original user.

FIDO:

Registers a FIDO2 account using standard FIDO CAT2 protocol. A negative response suggests a timeout or unrecognized user or user actively rejecting the ask.

U2F:

Registers a U2F account using standard U2F protocol. A negative response suggests a timeout or unrecognized user or user actively rejecting the ask.

(TBD)

TruYouID Plugin API

A mobile device plugin TruYouID is being release on GitHub for Android and iOS developers providing functionality to detect and recognize the person in possession of the device. Its intended application is to server as a smart lock providing only the known original user access to app facilities. A simple example would be the authenticator that by use of TruYouID limits access to only the original user and preventing access by another person. It could likewise be used to secure access to any app or app facility.

Additional information will added as it becomes available and transitioned to GitHub when the product does so.

(TBD)