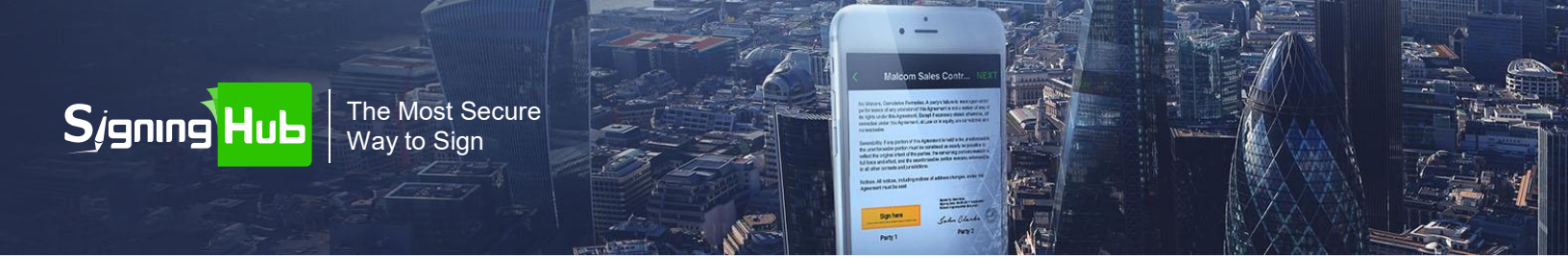


SigningHub – Using Qualified Remote Signatures

This document explains how SigningHub implements Qualified Remote Signatures that are compliant with eIDAS EN 419 241 Level 2 Sole Control requirements. In plain English this means ensuring that a user has been authenticated and authorised the use of their centrally held Qualified Signature key with high degree of confidence. This high-trust solution uses the user's registered mobile device to create a digitally signed authorisation file which SigningHub verifies before releasing the centrally held, HSM protected Qualified Signing key and certificate.

The user can see what they are being asked to sign. The signed authorisation file identifies the user, the document being signed, the mobile device fingerprint and the qualified certificate to be used server-side. The signed authorisation file is kept as long-term evidence.

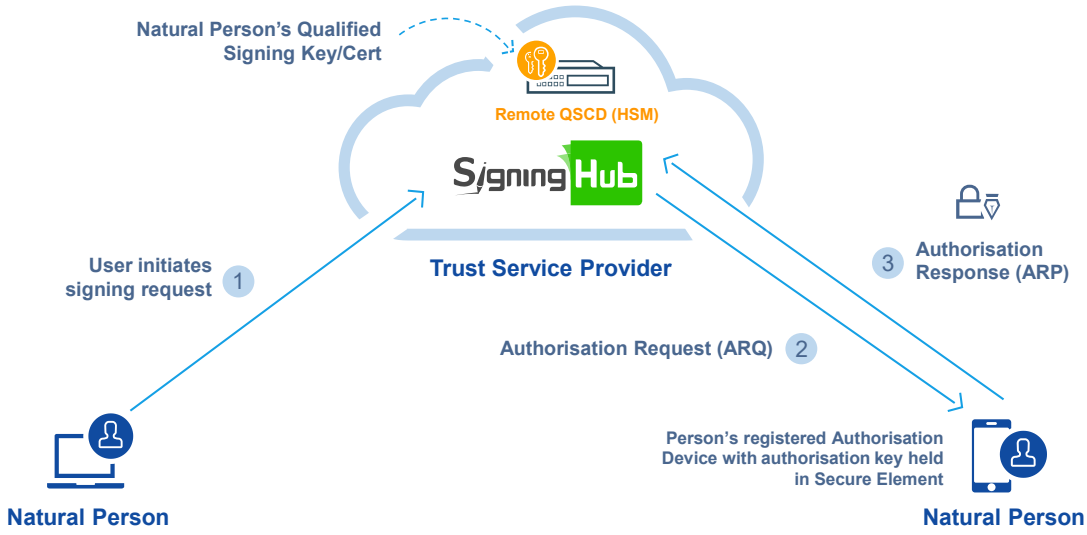
System Actors

The following actors are involved in the creation of a remote QES:

- **Natural Person**
  This is the owner of the Qualified signing key and Certificate held securely by the TSP. To ensure the properties of "Level 2 Sole Control" this person's key is only released for signing purposes once the person authorises this via a trusted channel.
- **SigningHub**
  This software / service allows the person to review documents and to request the creation of a qualified signature. SigningHub manages the process and requests the person to authorise the signature via a separate channel.
- **Remote Qualified Signature Creation Device (QSCD)**
  This is the trusted tamper-resistant Qualified Signature Creation Device HSM certified to Common Criteria level EAL4+, using the Protection Profile specified in EN 419211-5.

Qualified Signature Creation Process

The following diagram explains how the SigningHub remote qualified signature process works.

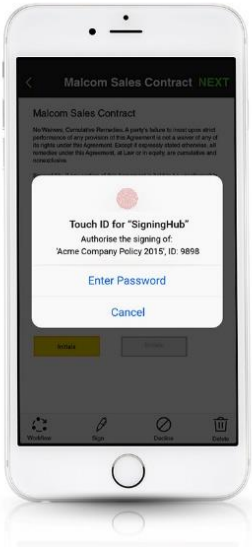


The diagram depicts a person interacting directly with SigningHub using its web interface, it is also possible for SigningHub functionality to be embedded inside a front-end business application's webpages using the SigningHub REST/JSON API. The steps are:

1) The person logs into SigningHub using a secure HTTPS session. The login process may require a simple username/password and optionally using a second factor. SigningHub can also use the identification and authentication service of an external Identity Provider (IDP) using the SAMLv2 protocol. Once logged in the user can review a document and request it to be signed.

2) SigningHub will create an Authorisation Request (ARQ) message. The ARQ contains:
   (a) Document Name and the Document ID
   (b) Hash of the document to be signed
   (c) SigningHub Instance ID and the user's ID
   (d) The User's Qualified Cert Alias

3) SigningHub then displays a message to the person asking them to launch the SigningHub mobile app on their registered device to authorise the remote signing request. Once the person launches the SigningHub mobile app the ARQ is automatically downloaded into the user's mobile app and shows a pop-up message as shown (iOS version is shown):



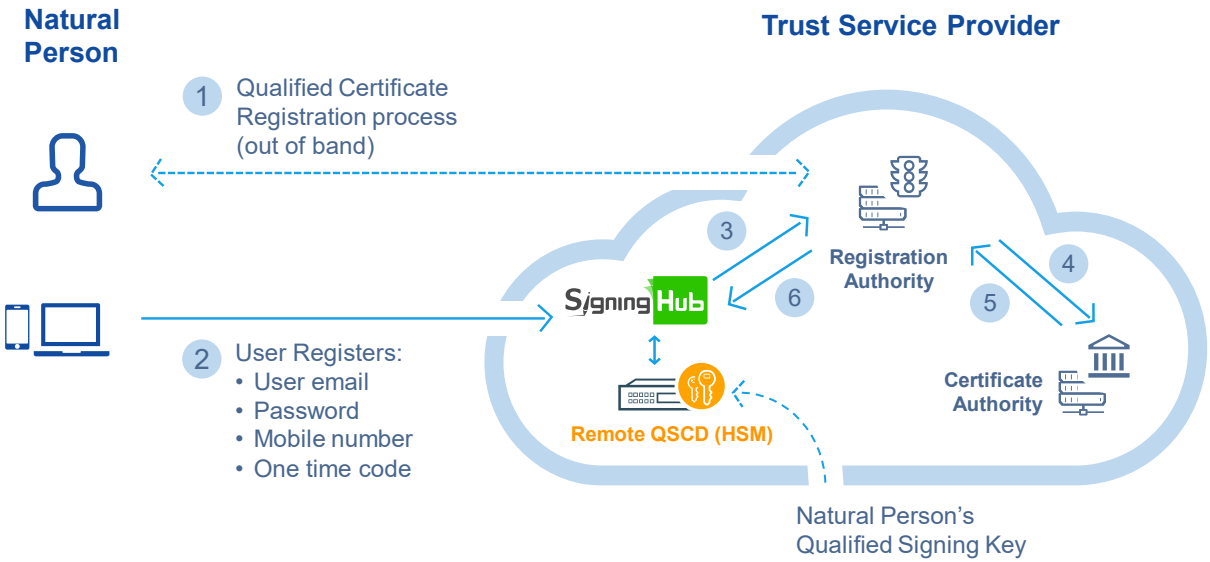
   a. The SigningHub app displays the Document Name and ID which is being requested for authorisation for remote signing. The person may authorise using their fingerprint biometric check assuming this is supported by the mobile device or on iOS by entering the SigningHub account password.
   b. If the user is authenticated successfully, the mobile app creates a signed Authorisation Response (ARP). It signs the ARP using the authorisation private key stored in the mobile device secure element /enclave and returns the signed ARP back to the SigningHub Server.
   c. SigningHub verifies the ARP was signed using the person's registered device and uses its registered certificate to confirm and verify the ARP signature. If it is valid then the user's wrapped private key is passed to the HSM, unwrapped and the used to create the Qualified Signature on the document.
   d. SigningHub writes a secure log entry containing the signed ARP as a digitally-signed proof that the remote signature was authorised by the person using one of their pre-registered authorisation devices. The signed ARP message can be downloaded from the server and verified independently.

The User Registration Process

Before a person can start using the SigningHub remote signing service they must first be securely vetted and their qualified signing key has to be created inside the HSM and their Signing Certificate issued by the TSP's CA.

The Trust Service Provider (TSP) will comply with all the relevant TSP policy and security requirements for TSPs supporting qualified signature creation and validation services as well as the issuing of qualified certificates. Of course, if this solution is being used for Advanced Electronic Signatures (AES) then adherence to these policy requirements is not essential.

The process for registering the person, creating the signing key inside the HSM and requesting a Qualified Certificate for the person is as illustrated and described below:



1) The person needs to complete a formal registration process for a qualified certificate with the Registration Authority of a Trust Service Provider (TSP). If granted, a One Time Code is usually provided to identify this user's request. It is possible for the TSP to make an API call to create the user's SigningHub account once the vetting process completes successfully - if so skip to (3).
2) Once the person's identity is verified, the person can register with SigningHub using a secure HTTPS session. The person provides their email address, mobile number and choose a password based on a strong password policy. The person must provide the OTC code issued in step 1 above. SigningHub checks that the person has access to their email address by sending an email account activation link to this address.
3) Once the person's account is active, SigningHub generates a qualified key pair inside the HSM for this person. The private key is exported from the HSM using an HSM Key Encrypting Key (KEK) and stored. SigningHub also creates a qualified certificate signing request (CSR in PKCS#10 format) and passes it to the TSP, together with the user's OTC value.
4) The TSP CA System retrieves its previous registration record using the OTC reference and verify that this certificate request is for the same person. If so it will make a certificate request and respond back to SigningHub with the certificate which it stores it in the database.

Authorised Device Registration Process

The SigningHub user is notified that they must download the SigningHub mobile app (iOS or Android) on to their mobile device(s) which they wish to set-up for remote signature authorisation purposes. The SigningHub app is available free of charge on Android and Apple stores.

Once the person has installed the SigningHub mobile app and logged in with their SigningHub credentials, they are asked if they wish to register this mobile device as an authorisation device for remote signing. If the person agrees then SigningHub will send an OTC via SMS to the user's registered mobile phone (this registered phone number may be same device as currently being registered for remote signing authorisation purposes or a different mobile device). This proves the user is in control of their mobile phone.

The SigningHub mobile app prompts the user to enter this OTC code.

The SigningHub mobile app will verify the OTC with the SigningHub Server and if found to be valid, the mobile app then:

a) Creates a secure authorisation key pair (ECDSA or RSA) within the mobile's Secure Element (referred to Secure Enclave on iOS devices).

b) Creates a certificate request message (CSR in PKCS#10 form) and sends it to the SigningHub Server.

c) SigningHub certifies this authorisation key using its internal CA and stores the authorisation certificate together with the device details in the user's account information.

d) SigningHub links the person's authorisation certificate with their Qualified Certificate and stores this information in the user's account information. This is protected using cryptographic checksums. When the user wishes to sign, SigningHub ensures that Authorisation Response (ARP) was signed by a correct authorisation certificate which is linked with the qualified certificate before allowing the Qualified Certificate private key to be used for signing inside the HSM.

e) SigningHub notifies the SigningHub mobile app that it is now registered for signing authorisation purposes. The mobile app displays a message to the user explaining that the authorised device set-up has completed.

Secure Audit Trail

All steps of the process are logged within the SigningHub secure logs. The integrity of these is cryptographically-protected using sequenced HMACs (with keys stored inside the HSM). The logs include both the:

a) The request message from the application, including the person's mobile number

b) The response message from SigningHub including transaction status information

The HMAC security prevent log records being added, deleted or changed without an integrity issue being reported to all identified ADSS Server operators.

Summary

SigningHub provides a high trust solution for Qualified Remote Signing that uses a trusted path approach and strong PKI security to record and confirm all the user's wilful requests to sign a document. The eIDAS EN 419241-2 requirements are met and Ascertia is currently progressing a Common Criteria Evaluation for this Qualified Remote Signing option.

This option will also be of value to those clients that wish to use it for Advanced Digital Signatures using existing high trust PKI environments where smartcards and USB tokens are seen as expensive and awkward compared with centrally held and managed keys and certificates.

We expect to see various national legislations outside the EU being updated to allow suitably secure solution such as this to be used for creating high-trust Advanced Electronic Signatures.

For further information review www.SigningHub.com and check the short videos and blogs.