# SSO Logon with X.509 certificate

**Author :** Tobias Hofmann

**Date :** July 24, 2020

SSO logon with an X.509 certificate offers some benefits. In this blog, I'll cover the main benefits, problems and attention areas when using X.509 for SSO. As a practical example the X.509 logon with NetWeaver ABAP is shown.

To access an ICM service on a NetWeaver ABAP system (NW ABAP), you can choose from a wide range of logon procedures. A common requirement is to use a user certificate: X.509. In my example I am using in the following blogs, I'll enable X.509 based logon to the SAP WebGui for HTML ICF service.

Logging in to NW ABAP web app (WDA, UI5, OData) with a X.509 certificate is a web scenario. An HTTP client is accessing a protected resource. In NW ABAP, the web server is ICM, and ICM is handling the client certificate requests. The login itself is handled by standard user authentication in NW ABAP.

Login with X.509 is not overly complex. It helps to know the main components and flows involved in the process.

- Client certificate: X.509. Certificate that identifies a user.
- Intermediate server: Server that receives a client certificate and forwards it to NW ABAP.
- ICM: HTTPS server in NW ABAP that receives the client certificate
- NW ABAP: Server that hosts the protected web resources a user wants to access. Handles user authentication.
- PSE: Keystore in NW ABAP containing certificates of peers and CAs.

## The good

Using X.509 is bullet proofed. As I like to say: it's old technology. It's not cool, neither is it the perfect solution for many cloud native scenarios. But it works. It's that kind of solution that almost everyone supports. If you encounter a problem, chances are very good the problem was solved by someone else years ago. And it is secure. If not to say: it is very secure. And user friendly. Admins can automatically distribute a certificate to a user, and a browser can use it for logon without even showing a logon screen or dialog to the user.

Having X.509 logon and more modern protocols like SAML 2.0 or OpenID Connect working together is not problem at all. To log on at an IdP, you can use X.509. It’s a very good replacement for basic auth. You log on to your IdP with X.509 and then to other systems with the received tokens. For mobile solutions, this is even a best practice: distribute X.509 via MDM, log on to IdP for OpenID Connect, and then to the app services via the OIDC token. The user is not even seeing a logon screen once. The app? Uses only modern authentication, and can easily switch from OIDC to SAML, all transparent to the user.

To configure NW ABAP to support X.509 logons, follow these steps:

1. Configure ICM to accept client certificates
2. Add CA certificates to PSE
3. Create a X.509 user certificate
4. Enable users for X.509 based logon
    1. Mapping table
    2. Rule based to user
    3. Rule based to alias
    4. Rule based to specific user
5. Test

## The bad

Use of X.509 for logons comes with some drawbacks too. For instance, you cannot use it to create automatically a user in the SAP NetWeaver ABAP system. To achieve this, you should use SAML 2.0. If you need to provide additional user information like profile or groups, you cannot get this kind of information with a X.509 certificate. If you need this kind of information, I recommend using SAML 2.0.

Good UX implies to make life easier for the user. For X.509 this means that the certificate must be distributed to the user, or: the device the user is using. For browser-based logons, the user’s certificate must be available in the browser/computer key store. If the user has 2 laptops: 2 certificates must be distributed. In case the user has a mobile device, same rule applies: distribute a valid X.509 certificate to all the users’ devices. The certificates must be maintained, that is: updated / refreshed before they expire.

Certificates expire. They can be configured to be valid for years or decades, but they should be valid for at max 1 year in a production environment. The shorter the validity, the better. This means that you have to update the certificate from time to time. With an automatic process this is an easy task. In reality, its more complicated. But worth the effort.

## The ugly

Validating a certificate is an important task too. Browser vendors like Google are not checking the validity of a certificate any longer. You cannot simply do the same for user logons in a corporate environment. In case a certificate was revoked, the server needs to check this and deny a logon. A certificate can be checked by OCSP or CRL. This is important as besides the expiration day, no other information is normally checked. In case the user shared the private key, or is not any longer part of the organization, the certificate does not contain this kind of information. The receiving server must check at the issuing CA if the certificate is still valid.

Using CRL or OCSP for this check is easy in an on premise environment. In the cloud this is more complicated. Most cloud providers do not check the certificate. As long as the certificate chain validates, the user is logged on. Solving this is technically possible, but not easy. For a corporate PKI, the cloud service must be able to access the on premise CA. Depending on the service, this is rarely part of the underlying design. A user with an invalid certificate is able to log on and get a valid session. You can block the user in the backend, but what if a technical user is used? Or SAML 2.0 is used to automatically create a user in the backend? This is a complex topic that needs to be validated in the security architecture: IDM to ensure the user cannot log on with X.509 as well as a working MDM solution the delete certificates from devices.

## Wrap-up

With the above listed attention areas for X.509, using X.509 based logons are one of the better solutions for SSO. Don’t use basic auth. Try to use X.509 as the standard replacement for basic auth. Certificates can be used in combination with other logon procedures (OIDC, SAML, OAuth, etc.) or used in combination with an additional credential type in 2FA/MFA. Most important learning is: don’t be afraid of X.509 certificates. They are less complicated than everyone thinks and can offer a very high level of security.