SuisseID V2.0 SDK

Domaine spécialisé: IT-Security
Chargé: Prof. Gerhard Hassenstein
Expert: Dr. Igor Metz (Glue)

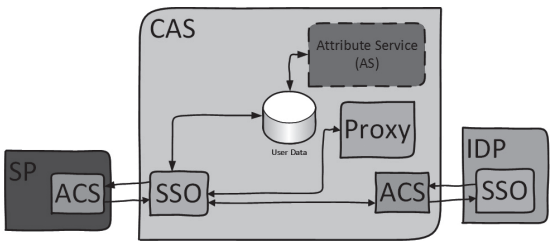
La SuisseID est le standard électronique Suisse d'identité sécurisée permettant à la fois une signature numérique valable juridiquement et une authentification sécurisée. Grâce à elle, des transactions peuvent être conclues en ligne entre des particuliers et des entreprises, entre entreprises et entre les citoyens et l'administration. Depuis mai 2010, la SuisseID est disponible sous forme de carte à puce ou de clé USB pour toute personne physique.

Contexte

Avec l'introduction de la SuisseID en mai 2010, un **Software Development Kit (SDK)** fut également mis à disposition. Le SuisseID SDK contient tous les éléments nécessaires afin de pouvoir faire une requête d'authentification et d'attributs auprès d'un **Identity Provider (IdP)** SuisseID.

La définition des attributs dans le SDK est entrée de manière fixe, rendant le SDK statique et inapproprié à des développements futurs. Une intégration de requête d'attributs provenant d'autres **Claim Assertion Service (CAS)** n'a pas été prévue, de même qu'aucune notion de proxying n'a encore été intégrée.

Le but de notre thèse de Bachelor a été de rendre le SDK plus flexible aux besoins futurs en y intégrant de nouvelles fonctionnalités.



Communication

Implémentation

Tout en se basant sur les spécifications d'**OASIS SAML** et ceux de la SuisseID, le SDK a été réécrit afin de contenir diverses extensions permettant d'implémenter le support des métadonnées, des notions de proxying, ainsi que de rendre le système plus dynamique en utilisant des schémas au lieu d'inscrire les informations de manière fixe.

Afin de rendre le système plus dynamique, la validation des attributs est faite au **runtime** à l'aide de schémas. Les attributs ne sont plus entrés de manière fixe au sein du SDK mais proviennent des metadatas.

Les notions de proxying ont également été intégrées au SDK et de ce fait, un Claim Assertion Service a été mis en place avec une adaptation modulaire aux requêtes ainsi que la possibilité d'ajouter de nouveaux attributs dans une réponse.

Afin de pouvoir conserver la **Chain of Trust** entre le **Service Provider** et l'**Identity Provider**, le **Claim Assertion Service** retourne les **assertions** signées par l'**IdP** au **Service Provider**. De ce fait, le **Service Provider** peut vérifier la provenance des attributs reçus grâce à la signature de l'**IdP**.

Une page web destinée à la gestion des métadonnées a également été mise en place. Elle configure la métadonnée locale. Grâce à elle, il est possible de sélectionner les attributs que l'on souhaite récupérer via les métadonnées des **IdPs** que nous connaissons. Elle permet également de définir l'**IDPList** pour le proxying.

Elie Jeannerat

Ciril Saner

Damien Schaeffer