Portable Personal Identity Provider in Mobile Phones

Md. Sadek Ferdous & Ron Poet

Software Engineering and Information Security (SE & IS) Research Group,
School of Computing Science, University of Glasgow.
Email:m.ferdous.1@research.gla.ac.uk, ron@dcs.gla.ac.uk

1 Introduction & Motivation

So many service providers (SPs) which cause user attributes to be scattered among different providers.

This results in serious consequences.

The management of attributes becomes increasingly difficult.

+

Users have limited control over those attributes.

+

Crucial attributes are often stolen by hackers from these large providers.

❑ To tackle these problems, we propose the concept of Portable Personal Identity Provider (PP-IdP) for mobile phones.

❑The current generation of smartphones are equipped with powerful h/w and intuitive s/w and can be used to access online services everywhere.

❑The iniquitousness of these smartphones can be leveraged to manage online identities of the users as well as manage user attributes.

❑PP-IdP is a type of identity provider (IdP) that is hosted in a mobile device owned and/or used by the user and that is under the full control of the user.

2 Architecture

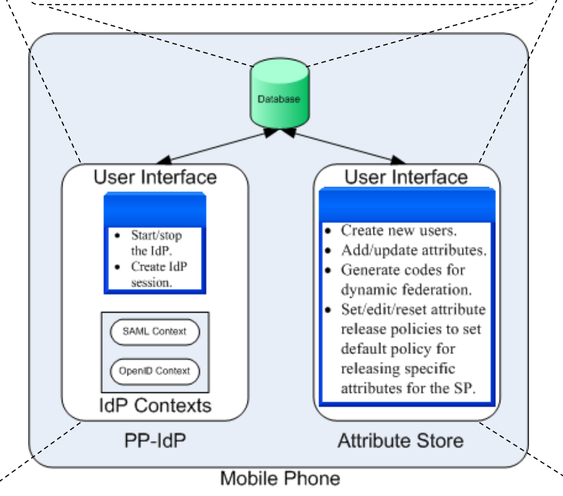
❑ The PP-IdP consists of a user interface (UI) and different IdP contexts.

❑The UI is used to start the IdP running in background and stop the IdP .

❑The IdP contexts are made of several individual context each of which is for handling a respective request.

❑For example, when an SAML request will come, the SAML context will handle that request.

The database is used to store user attributes, federation status and the attribute release policy.



❑ The attribute store is the central repository where user attributes will be stored.

❑It consists of a back-end database and the user-interface (UI).

❑The database actually holds the attributes. and also stores attribute release policy (ARP) for SPs.

❑The UI will allow the user to create and update user attributes, create new users for the IdP and add/edit ARP.

3 Challenges

❑Current Identity Management systems are based on Web Technologies. Therefore, the PP-IdP should be web-enabled meaning a web server has to be installed inside the mobile phone.

❑The current implementation of the existing Identity Management protocols such as SAML, OpenID requires that both the IdP and SP are visible to each other. Since our PP-IdP will be essentially hosted inside, we will need to slightly modify the API to ensure that can talk with each other.

❑For SAML, we need to find a way how federations can be created in a dynamic fashion.

❑ Implicit in every Identity Management is the issue of trust. Therefore, we need to closely examine the trust assumptions and requirements.

❑The PP-IdP should have a backend database with an appropriate user interface to store user attributes safely and securely.

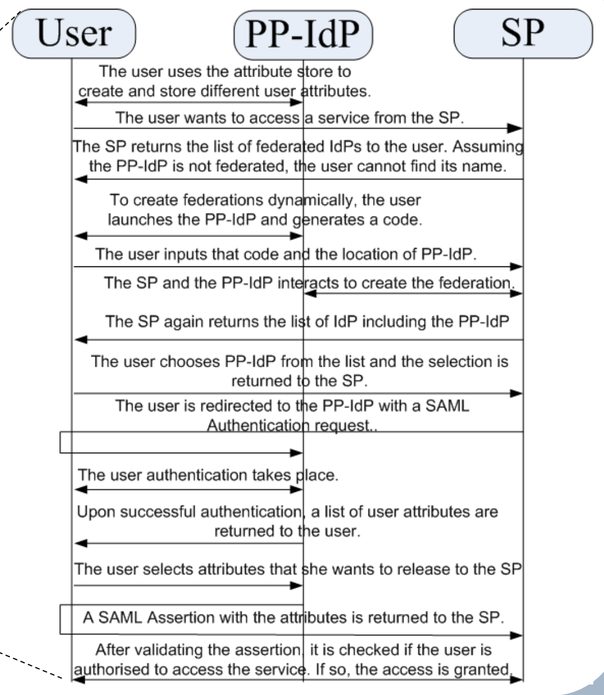
4 Use-case

❑ A simple use-case scenario based on SAML is presented here.

❑The assumption is that the PP-IdP and the SP are not federated previously.

❑Therefore, the federation is created at first in a dynamic fashion.

❑Then the usual SAML protocol flow continues.



5 Advantages

❑This approach enables users to have full control over the IdP and the attributes stored.

❑Users control which attributes they want to release and to which SP.

❑The Attribute Store acts as the central repository to store crucial user attributes. Since the attributes are not scattered over many IdPs, their management becomes easier.

❑Usually attributes stored in traditional IdPs are static in nature. PP-IdP can be used to create dynamic attributes on the fly (e.g. location) and could be used to deploy context aware Identity Management.

❑Advanced authorisation frameworks such as XACML could be integrated with the IdP to provide fine-grained access control mechanisms over the user attributes which could tackle the problem of releasing attributes to unknown parties common in current smartphones.

6 Current Status & Future Work

❑A proof of concept for the PP-IdP has been developed for the android platform.

❑SQLCipher has been used as the back-end database to store attributes securely.

❑Embedded Jetty has been used as the Servlet container.

❑Each IdP Context has been developed as a servlet.

❑The current implementation is based on SAML and OpenID.

❑The current implementation only supports SAML Web Browser SSO Profile. We plan to add more profiles soon.

❑We plan to add the functionalities of OpenID PAPE and Attribute Exchange into the OpenID Context.

❑Other IdM protocols especially the very popular oAuth could be added to our PP-IdP.

7 Conclusions

❑As the online services proliferate, it will be increasingly difficult for people to manage their attributes which are currently scattered over many places.

❑Using a portable personal IdP in mobile phones can be a great tool in aiding users to manage their attributes and will allow users to get back the control over their own attributes which is missing in the current setting.

❑We strongly believe that our approach has great potential and can have a far reaching impact in a world of mobile devices.

❑However, there are still a few improvements need to be done before its full potential can be harnessed.