On Identity-Aware Devices: Putting Users in Control across Federated Services

Marco Casassa Mont, Boris Balacheff, Jason Rouault, Daniel Drozdzewski

HP Laboratories, Bristol

HPL-2008-26
March 26, 2008*

identity management, device, privacy, user control, trust, federated services

This paper describes R&D work on "Identity-aware Devices", in the context of federated services. The aim is to put users in control of their credentials and identities and enable simple, secure, trustworthy and transparent access to federated services. Current users' experience in networked and federated services is difficult and painful, especially when using mobile devices (e.g. mobile phones, laptops, PDAs, etc.): users need to contact online service providers and authenticate against them; additional credentials might be issued and required to access services; credentials need to be stored in a safe and secure place. Users have little control over the release of their identity information and related processes. A solution to address these issues is presented, based on the concept of "Identity-aware Devices" and federated "Provisioning Services". "Identity-aware Devices" leverage trusted modules and are driven by policies and users' preferences. Part of this work has been carried out in the context of a Liberty Alliance initiative, in collaboration with BT and Intel teams, aiming at driving the next generation of interoperable identity solutions. A full working prototype has been developed and successfully demonstrated in a joint project. This is work in progress. Next steps and plans are presented and discussed.

Internal Accession Date Only Approved for External Publication

Submitted to TRUST BUS 2008, September 4-5, 2008, Turin, Italy

On Identity-Aware Devices: Putting Users in Control across Federated Services

[1]Marco Casassa Mont, [1]Boris Balacheff, [2]Jason Rouault, [1]Daniel Drozdzewski

[1]Hewlett-Packard Labs, Bristol, UK [2]Hewlett-Packard, US
{marco.casassa-mont, boris.balacheff, jason.rouault}@hp.com

Abstract. This paper describes R&D work on "Identity-aware Devices", in the context of federated services. The aim is to put users in control of their credentials and identities and enable simple, secure, trustworthy and transparent access to federated services. Current users' experience in networked and federated services is difficult and painful, especially when using mobile devices (e.g. mobile phones, laptops, PDAs, etc.): users need to contact online service providers and authenticate against them; additional credentials might be issued and required to access services; credentials need to be stored in a safe and secure place. Users have little control over the release of their identity information and related processes. A solution to address these issues is presented, based on the concept of "Identity-aware Devices" and federated "Provisioning Services". "Identity-aware Devices" leverage trusted modules and are driven by policies and users' preferences. Part of this work has been carried out in the context of a Liberty Alliance initiative, in collaboration with BT and Intel teams, aiming at driving the next generation of interoperable identity solutions. A full working prototype has been developed and successfully demonstrated in a joint project. This is work in progress. Next steps and plans are presented and discussed.

1 Introduction

Mobile devices (e.g. PDAs, smart phones, laptops, etc.) are becoming more and more pervasive. They are used by people to carry out personal and work-related tasks: this includes accessing information and services on the Internet via network and telecom providers. Trends in this space, affecting users and service providers, include: converged, all-IP (Next Generation) networks; new IP-Multimedia Subsystem (IMS) services; federated services. There are great expectations and business opportunities but also issues that need to be properly addressed.

Current users' experience with mobile devices, in networked and federated services, is difficult and painful: users need to create (one or more) user accounts, disclose profile information, authenticate against service providers, get additional credentials to access services and ensure that these credentials are stored in a safe and secure place. For example, to access the Internet via a wireless network, a user, using their mobile device, might need to register to a local wireless provider (or login to their telecom provider with an existing account) and disclose some personal and financial data (e.g. credit card details); an "access token" (i.e. a credential) is likely to be "released" to the user to access the wireless network. This "access token" might need to be locally stored on the device, in case the user is accessing the wireless services at different points in time, during token's validity period.

Different types of credentials (also referred in this paper as "identity tokens" or, simply, as "tokens"), might need to be handled in different contexts and scenarios, including: 802.1X wireless authentication tokens, VPN tokens, InfoCard/CardSpace tokens, SAML assertions, OpenId tokens, X.509 credentials, etc. Some of these credentials might contain personal information. They need to be protected and processed according to user's preferences, privacy and security policies. Credentials, policies and preferences have a lifecycle (creation, provisioning, update/changes, expiration, revocation, etc.) that must be properly managed and controlled. Misuses (from the user side) of these credentials should also be prevented.

In general, users have little control about their identities, credentials and involved processes. In the context of federated services and federated identity management, such as [1,8,17,18], Identity Providers and associated Service Providers (Relying Parties), can help in providing authentication and single-sign-on capabilities, along with some management of user profiles and preferences. However, users still have to “manually” deal with their authentication to online services and potentially handle additional “credentials/identity tokens” (along with their lifecycle) to access federated services and/or other online services.

2 Objectives and Problem Space

Our work aims at exploring the roles that (mobile) devices and related “support” services can have, in the context of federated services, to: (1) help users to handle their “credentials/identity tokens” according to preferences, security and privacy policies; (2) mediate and simplify users’ interactions with service providers. Our goal is to make progress towards addressing the following problems:

- How to enable users to improve their control of their credentials (e.g. identity tokens, identity credentials, identity attributes, access tokens/rights, etc.) by confidently and safely using (mobile) devices;
- How to handle the lifecycle of credentials by securely provisioning, storing, accessing and using them in “trustworthy” devices, driven by policies and user preferences;
- How to enable a simple, secure, trustworthy and transparent access to federated services (via predefined protocols) by using these devices;
- How to enable “service providers” to have degrees of confidence and assurance that these credentials will be subject to agreed policies rather than abused.

3 Analysis and Vision

Our analysis focused on a federated service context where online services can be accessed by users via devices. We have been pragmatic and took into account current and foreseeable constraints, i.e. the fact that, to access the services or engage in the business interactions, credentials and (degrees of) identity information need to be disclosed by users, either to Service Providers or to Identity Providers (that will mediate interactions with Service Providers). For example, in Liberty Alliance [1] an Identity Provider has some knowledge of users’ identity information and profile: it releases assertions and credentials to Service Providers in order to enable users to access their services. Privacy management is indeed important: progress will be made in this direction, for example as envisioned in PRIME [6].

In the short and medium term, we believe there is an opportunity to innovate at the “device” and service level, to provide users with additional degree of control and protection of their credentials, “simpler” interactions with federated services along with “assurance” credentials will not be misused. In this context our research effort has been focusing on how to handle credentials based on policies, how to provision them to devices and deal with their lifecycle, how to empower users.

Our vision, centered on the concept of “Identity-aware Devices” and “Provisioning Services”, aims at simplifying the way users interact with federated services by means of “trustworthy” devices, which ensure that personal data and credentials, once provisioned to an “Identity-aware Device”, can be stored, processed and disclosed in a safe and trustworthy way, according to predefined policies and preferences: users have degrees of control over their data and credentials, for example during their disclosure and usage. In addition, identity providers, releasing (some of) these “credentials” to the users, and service providers (relying on them), will have additional assurance that the credentials they might issue are not going to be misused. Our vision of “Identity-aware Devices” consists of the following key aspects:

- “Identity-aware Devices” are secure and trustworthy “Personal Identity Providers”, driven by security and privacy policies;
- These devices can used “safely to store “credentials/identity tokens” and related policies, created either directly by users or by means of trusted “Registration and Provisioning Services” that can also manage their lifecycle. Usage and processing of these credentials is subject to these associated policies;
- These devices can act on behalf of their users and/or other identity providers (for example, identity providers in a federated services scenario) via delegated “credentials/identity tokens”;
- Users have a simplified, safe and secure interactions with federated services (including single-sign-on, identity management tasks, access requests) as communications with service and con-

verged network providers are mediated by the “Identity-aware Devices”, driven by policies and specified preferences.

This vision has been shaped in the context of collaborative Liberty Alliance projects with BT and Intel. More details are provided in Section 4. It factors in the evolution of the role that (mobile) devices can have in the context of federated services, as shown in Figure 1.

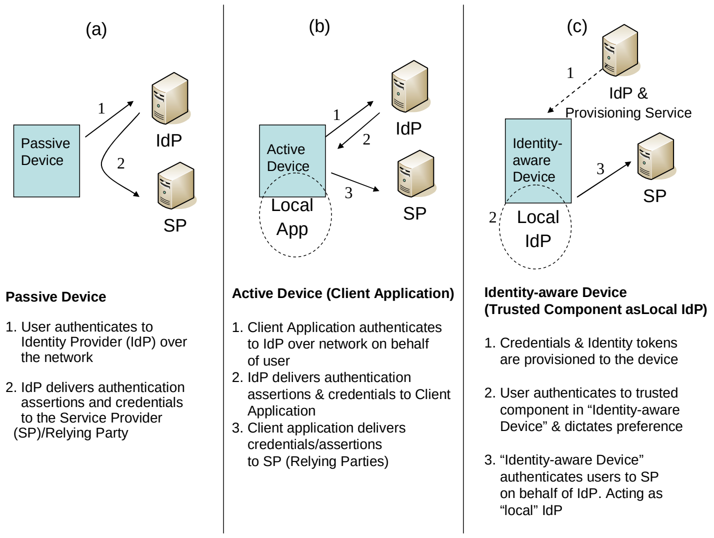

(a)
Passive Device
1
2
IdP
SP
Passive Device
1. User authenticates to Identity Provider (IdP) over the network
2. IdP delivers authentication assertions and credentials to the Service Provider (SP)/Relying Party
(b)
Active Device
Local App
1
2
3
IdP
SP
Active Device (Client Application)
1. Client Application authenticates to IdP over network on behalf of user
2. IdP delivers authentication assertions & credentials to Client Application
3. Client application delivers credentials/assertions to SP (Relying Parties)
(c)
1
IdP & Provisioning Service
Identity-aware Device
Local IdP
2
3
SP
Identity-aware Device (Trusted Component asLocal IdP)
1. Credentials & Identity tokens are provisioned to the device
2. User authenticates to trusted component in “Identity-aware Device” & dictates preference
3. “Identity-aware Device” authenticates users to SP on behalf of IdP. Acting as “local” IdP

Fig.1. Evolution of the role of Devices in Federated Services

Figure 1(a) illustrates the most common use of current devices, i.e. as “passive” clients. Users need to register with “Identity Providers” and potentially have to disclose personal data. Afterwards, users use these devices to authenticate to an Identity Provider (for example via web browsers) and be redirected to Service Providers. If required, the Identity Provider (IdP) will disclose, via a back-end channel, users’ profiles and identity information to the Service Provider.

Figure 1(b) describes the evolution of these devices towards “Active Devices”. Users still authenticate to an IdP to get access to Service Providers’ services. However this step is mediated by an “application” that is locally installed in the device and that can collect assertions and credentials. This application can interact with Service Providers to disclose these credentials, based on needs. This application could be a plug-in in a web browser or a standalone application running on the device.

Figure 1(c) illustrates the further step towards “Identity-aware Devices”. An “Identity-aware Device” has a “trusted module/component” that, among other things, acts as a “Local Identity Provider”. It can still receive credentials from a remote IdP, after user’s authentication, and store them locally. The remote IdP, using the “*Provisioning Service*”, is in charge of handling the lifecycle of these credentials (based on associated policies. Alternatively, part of these tasks can be delegated to the “Identity-aware device” that will locally handle the lifecycle of credentials, according to associated policies and users’ preferences. Some of these credentials might actually be “delegation credentials”, i.e. credentials that enable the local IdP to issue additional credentials (subject to policies and constraints).

A key peculiarity of the “Identity-aware Device” is that it enables its users to autonomously engage in interactions with federated services, without necessarily having to contact the “Remote IdP” (and/or requiring SPs having to interact with this remote IdP). This involves: (1) separating the “provisioning phase” of credentials from the phase where these credentials are actually used (i.e. a credential can be used at a different time and in a different place under supervision of a user); (2) enabling local (at the device level) processing and lifecycle management of credentials. This helps to simplify users’ interactions and gives them degrees of control of their credentials.

The remaining part of this section discusses, in more details, a possible model of “Identity-aware Device” and “Provisioning Service” along with a scenario illustrating their potential usage.

3.1 Model of Identity-aware Device

Figure 2 shows a high-level model of an “Identity-aware Device” and related “support services”, namely the *Provisioning and Registration Services*.

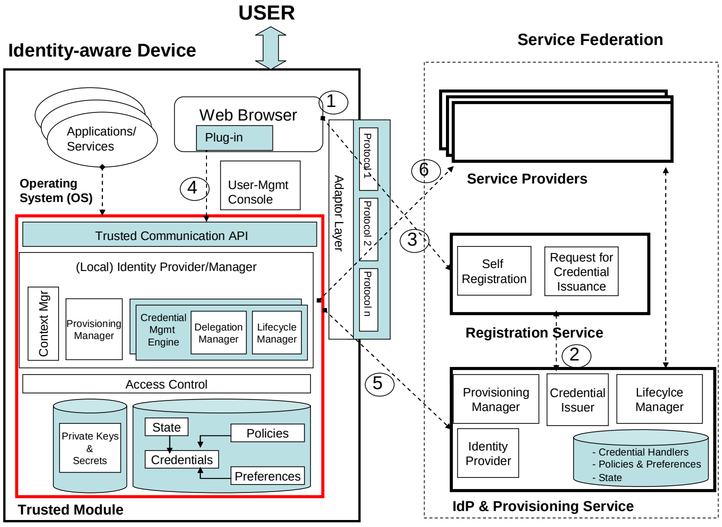


Fig.2. High-level Model of Identity-aware Device and Support Services

This model aims at being independent from specific federated service protocols and frameworks. Section 4 describes in more details an adaptation and deployment of this model in the context of Liberty Alliance [1], in a joint collaboration project with BT and Intel. This high-level model identifies the following key roles and entities:

- **User**: an individual interested in accessing (federated) services. Users use their “Identity-aware Device” to achieve this;
- **Identity-aware Device:** it is a device enhanced with “trusted components”. It can be provisioned with credentials/identity tokens and it can handle their usage and lifecycle autonomously from the IdP (according to policies and users’ preferences) whilst mediating users’ interactions with service providers (via standard federation protocols). In this context this device acts as “local Identity Provider (LIdP)”. As anticipated, “Identity Providers & Provisioning Services” could delegate to these devices the capability to issue new credentials, subject to well defined policies and constraints;
- **Service Provider**: it provides services to the authenticated users, based on their access rights and credentials. In this context it relies both on Identity Providers and “Identity-aware Devices” to get these credentials;
- **Registration Service**: users access this service in order to: (1) self-register their profiles and “Identity-aware Devices”; (2) request credentials to be provisioned. This service triggers the process leading to the issuance of credentials to the users, by means of interactions with the Provisioning Service and the “Identity-aware Device”. The actual issuance of credentials can be done by a “Trusted-Third Party” involved in the service federation or by the Identity Provider itself. Depending on the context, the role of “Registration Service” could also be played by the “Identity Provider”;
- **Provisioning Service**: this service primarily engages with “Identity-aware Devices” to provision them with credentials (and related policies). It also handles the service-side lifecycle of credentials. This role can be played by the Identity Providers;
- **Identity Provider**: it provides users’ authentication and enables single-sign-on across federated services. The Identity Provider can also play the role of Registration Service and Provisioning Service. It can delegate some of these tasks to the “Identity-aware Device”, subject to policies.

As illustrated in Figure 2, an “Identity-aware Device” contains a ***Trusted Module (TM),*** embedded within this device. The Trusted Module is tamper resistant and its integrity can be checked/verified (by a remote entity). It can be implemented as a (combination of) software and hardware components. Trusted

Computing technology, such as TPMs [2], can be leveraged to provide: tamper resistant storage of tokens and sensitive data; Direct Anonymous Attestation (DAA) capabilities [3] to anonymously check for its integrity. This increases its level of "trustworthiness" and assurance on how it is going to operate. Alternative approaches to implement Trusted Modules are currently under research: some options are illustrated in Section 6.

The Trusted Module contains the following key components:

- **Local Identity Provider**: this component is fundamentally an Identity Provider (IdP), ensuring that the device can interact with the federated services based on existing protocols. This component is an "orchestrator" of lower-level components: via an adaptor, it can use a series of federated protocols (e.g. Liberty Alliance, OpenId, WS-Federation etc.) to communicate with the service providers;
- **Provisioning Manager**: this component is in charge of interacting with a remote Provisioning Service to start the provisioning of credentials/identity tokens and ensuring that they are going to be locally stored and handled based on agreed policies and preferences. This is achieved by instantiating one or more Credential Management Engines;
- **Credential Management Engine**: this is the component that understands specific types of credentials/identity token and is in charge of dealing with their access and lifecycle, based both on local policies and further messages exchanged with the remote Provisioning Service. It is in charge of interacting with Service Providers (potentially mediated by the Local Identity Provider) to provide authentication tokens and credentials. This component can enforce different kind of policies, such as:
    a. **Access control policies:** policies dictating who can access credentials/identity tokens, whom these credentials can be disclosed to, etc.**;**
    b. **Retention/expiration policies:** these policies might include obligation constraints, that require to be handled independently from access control [19]**;**
    c. **context-based policies:** policies dictating constraints based on context, such as location;
    d. **Lifecycle management policies:** policies dictating criteria under which credentials can be updated, modified and deleted.

  The Credential Management Engine also takes into account users' preferences, associated with the credentials (for example in terms of black-lists of service providers not to be interacted with, retention and deletion time of credentials, etc.).
- **Tamper-resistant storage of credentials, policies, preferences and secrets (private keys, etc.):** this is a secure storage of confidential and private material. It can be implemented, for example, by using Trusted Computing technologies (e.g. TPMs);
- **Trusted Communication API**: a set of API to enable interactions with the external world.

The "Trusted Communication API" enables applications and/or users, locally authenticated with the device, to check for their credentials, manage these credentials (under specified policy constraints) and set preferences. Users can use this API to self-issue their own credentials, along with related policies and preferences. This provides users with additional control over their credentials (stored in a device) and their lifecycle.

A user, willing to engage with federated services, will use their "Identity-aware Device" to interact with the Registration Service (1) (see Figure 2), for example by means of a web browser (and secure, SSL-based web connections), to self-register his/her profile and the device. This service will also enable the user to require the issuance of credentials. The Registration Service requires (2) the "IdP & Provisioning Service" to generate these credentials, along with associated policies.

The Registration Service will return a set of "credential handles/references" to the user, via the web browser (3). The web browser communicates this information (e.g. via a plug-in) to the "Local Identity Manager" (4) on the "Identity-aware Device". The "Local Identity Manager" (via the "Provisioning Manager") will ensure that one or more "Credential Management Engines" are instantiated, potentially one for each "credential handle/reference" that has been received.

At a due time (for example driven by the user and/or related policies) a "Credential Management Engine" will interact with the "IdP & Provisioning Service" (5), to retrieve the actual credential and related policies (associated to a "credential handle/reference"). This interaction can take place via hostile medium (i.e. Internet), as the whole transmission between the "IdP & Provisioning Service" and the "Credential Management" is protected by the cryptographic protocols. IIn case, the user might express further preferences on how to handle the credential (e.g. on retention time or access control).

A provisioned "Identity-aware Device" can then be used to engage with federated Service Providers (6). If the right credentials have been provisioned, the "Local Identity Provider" (via the underlying "Credential Management Engine") will engage in the user authentication process and disclosure of credentials, according to specified policies and preferences.

In this model, the steps involving the registration, provisioning, local management and usage of credentials/identity tokens can be explicitly managed and are subject to the enforcement of policies.

In particular, we believe it is important to differentiate the "registration phase" from the "provisioning phase". The registration phase gives users the "right" to get some credentials/tokens. The actual credentials might not yet be active or ready to be used e.g. based on agreed policies or context (such as location). Only during the "provisioning phase" the user (by means of the "Identity-aware Device) actually gets the credentials, along with relevant policies and constraints.

3.2 Reference Scenario

This section briefly describes a reference scenario, used in the remaining part of this paper. It involves a telecom provider having both a web service and a "converged network" (all-IP) presence. This telecom provider supplies network-based federated services: other services are provided by its business partners (e.g. telecom providers operating in other countries).

In this scenario, the telecom provider also plays the role of the "Identity Provider" as it drives the service federation. "Identity-aware Devices" are supported by the involved parties.

Users are required to create an account with this telecom provider and might share some identity profiles (e.g. financial details). The telecom provider allows users (via its federated "Registration and Provisioning" Services) to provision their "Identity-aware Devices" with "authentication" and "network access" tokens (for subsequent access to their wired/wireless networks and related servcies), along with related usage policies and users' preferences.

This information is safely stored in the "Trusted Module" within the "Identity-aware Device". The "Identity-aware Device" is then used to directly access federated services (subject to policies and preferences): this device acts as a "Personal Identity Provider". Further "credentials" (e.g. about financial details) can be directly self-created by the user and injected into the device, along with related policies.

Specifically, in this scenario, a registered user wants to provision his/her "Identity-aware Device" with a "network access" credential/token. This credential enables the user to access wireless networks supplied by the telecom provider and/or its partners. This is achieves by interacting with the Registration Service and Provisioning Service. Associated policies (defined by the Identity Provider) dictate constraints on what can be accessed, by whom and for what time. The user specifies additional preferences dictating additional constraints on which Service Providers the device can interact with these credentials. This happens whilst the user is at work (by connecting to these services via a traditional LAN connection).

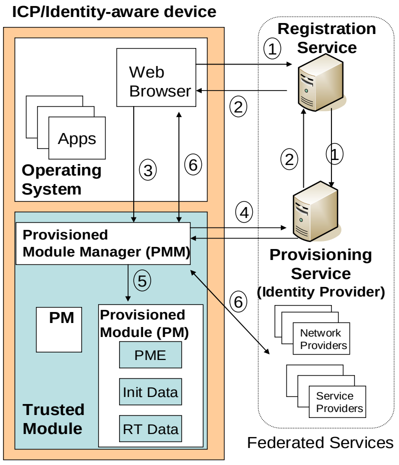
The user then needs to travel. The user will use their "Identity-aware Device" at an airport lounge to mediate the access to the telecom wireless network and other federated services. In this context, the process of checking for the integrity and trustworthiness of the involved systems is (transparently) carried out by both the "Identity-aware Device" and the service providers by means of DAA attestation. To access federated services, no further authentication or connection with the remote "Identity Provider" is required – as long as the provisioned "credentials/identity tokens" are valid. The user has degrees of control on their credentials/identity tokens as they are "actively" involved during the interactions with federated services (e.g. by acknowledging the explicit disclosure of credentials).

4 Current Approach

This section describes a possible approach and solution to implement "Identity-aware Devices". It is the result of a collaborative project, and a related Proof-of-Concept (PoC), with BT and Intel, in the context of the Liberty Alliance [1]'s "Advanced Client Technologies" (ACT) [4] initiative.

This initiative, based on Liberty Alliance standards [1], aims at defining and specifying technologies in the areas of identity-aware devices, single-sign-on, identity federation, service hosting, reporting and provisioning. In this context we refer to “Identity-aware Devices” as “**Identity Capable Platforms” (ICPs)**. An ICP device can be seen as a specific mapping/view of the “Identity-aware Device” concepts in the context of Liberty Alliance (LA). Please notice that the “Identity-aware Device” concepts can be mapped and instantiated in other federated identity and service management contexts.

The key goal of the joint project with BT and Intel was to explore and build Proof-of-Concepts (referred, in this paper, as PoC, version 1 and 2) to enable users, by means of mobile “ICP/Identity-aware Devices”, to engage in federated, multi-party interactions and transactions (on the Internet or other networks) in a simplified and transparent way: as anticipated in section 3 these devices store, process and potentially disclose “identity and credential tokens” in a secure, private and policy-controlled way. A first, cut-down prototype, implemented in PoC version 1 (PoCv1), consists of: (1) “ICP/Identity-aware Devices”; (2) Federated (identity management) Services; (3) Registration and Provisioning Services, accessible as federated services. Figure 3 provides a high-level, architectural view of PoCv1:



1. The users accesses the Registration Service. This service registers the device to be provisioned, along with required credentials/identity tokens at the Provisioning Service.
2. The Provisioning Service sends a reference to each token to the browser with instructions to send the reference to the PMM.
3. The browser submits the identity reference to the PMM
4. The PMM dereferences the token at the Provisioning Service and gets back the actual token
5. The PMM instantiates the PM within the trusted module
6. The user, via the “identity-aware device” access federated/network services, by using Liberty Alliance ID-FF and ID-WSF standards

Fig.3. Architecture of an Identity Capable Platform (ICP) and Interactions

In this implementation, the “ICP/Identity-aware Device” contains a “Trusted Module/Partition” (TM) hosting the following components (see Figure 3):

- **“Provisioned Module Manager” (PMM)**: the Provisioned Module Manager is a component running on the client platform which provides a “contact point” for provisioning operations, which can involve one or more Provisioned Modules. With reference to the “Identity-aware Device” model, this PMM component covers functionalities provided by the “Provisioning Manager” and the “Local Identity Provider/Manager” components;
- **One or more “Provisioned Modules” (PMs)**: With reference to the “Identity-aware Device” model, a PM module covers functionalities provided by the “Credential Management Engine” component.

  Specifically, in this implementation, the “**Provisioned Module”** handles provisioning, management and disclosure of credentials/identity tokens, based on policies and preferences. It supports:
  - ***Lifecycle of credentials/identity tokens*****:** This includes support for provisioning, update, deletion of credentials; activation, deactivation of credentials; serialization/de-serialization of credentials; credential portability; etc; The current PoCv1 prototype only supports the provisioning and update of credentials/identity tokens. Additional lifecycle management capabilities are going to be supported in PoCv2.
  - ***Policy controlled access and operations***: this defines which user can access which credential/identity tokens; what can be done with each token; lifetime of a token; enforcement of user preferences (e.g. on blacklisted service providers). PoCv1 specifically focused on “802.1X wireless authentication tokens”, based on the reference scenario illustrated in Section 3. Currently PoCv1 supports only very simple policies, focused on access control. More advance policies (including expiration/retention criteria and users’ preferences) are going to be implemented in future versions.

Consistently to the “Identity-aware Device” model, all these modules are part of a **Trusted Module (TM)**. In the current PoCv1 prototype, this Trust Module is implemented as a combination of a software agent and a hardware component. The hardware part leverages the Trusted Computing [16] TPM Module [2], to provide tamper resistant storage of tokens and sensitive data. Direct Anonymous Attestation (DAA) capabilities are planned to be used in a coming version, to enable anonymous integrity checking and attestations of the remote platforms. A software implementation of the Identity Capable Platform has been made available by Intel.

This “ICP/Identity-aware Device” currently supports both Liberty Alliance (LA) Identity Federated Framework (LA ID-FF) and Identity Web Service Framework (LA ID-WSF) standards as well as the Liberty Alliance (LA) Active Client Technology (ACT) [2] standard.

The Federated Services deployed in our PoC are also compliant with LA ID-FF and ID-WSF standards [1]. In this context the **Registration Service** is considered as a special type of “Service Provider”, that relies on the Identity Provider for user authentication and single-sing-on. Specifically, the **Registration Service** provides basic registration capabilities and interactions with the Provisioning service, to request for the issuance of credentials. As described in section 3, the actual issuance and provisioning of these tokens is done by the **Provisioning Service**. In PoCv1, the **Provisioning Service** has been co-located with the Identity Provider and is compliant with LA standards [1]. This service ensures that the provisioning process can be carried out in different phases: a first phase consists of just issuing the “ICP/identity-aware device” with a “reference/handle” to the credential/identity token – as this identity token might not yet be active or available to be used. In the next phase, the “ICP/identity-aware device” engages with the Provisioning Service to de-reference this token. This enables flexibility and a convenient way to differentiate the credential/identity token request phase from the actual provisioning and usage phases.

The PoCv1 prototype leverages and extends the HP Software Select Federation [5] solution to implement the Registration and Provisioning Services. Additionally, Select Federation provided the underlying framework enabling single-sing-on and federated services.

Figure 3 provides more details about the overall interactions implemented in the current PoCs. Due to lack of space, it is out of the scope of this paper to describe the details of LA protocols and exchanged messages used in PoCv1. This information is publicly available in the Liberty Alliance portal [1,4].

A working demonstrator of the PoCv1 prototype (jointly developed in collaboration with BT and Intel) has been successfully presented at RSA 2007. This demonstrator was based on the reference scenario illustrated in section 3.

5 Related Work

Related work in this space has been carried out in the context of Liberty Alliance initiatives, specifically in the context of the “Advanced Client Technologies” [4] and “Identity Capable Devices”. We have been directly involved in this activity. This work has been inspirational: it helped us to shape our vision of “Identity-aware Devices” by further abstracting related notions and concepts, in a “protocol agnostic” way. In addition, our work aims at putting more emphasis on the concept of the management of policies and preferences associated to credentials/identity tokens, along with enabling users to have more direct control of the overall lifecycle process.

R&D work done in the EU PRIME Project [6] is also relevant, in particular the criteria and approaches to handle identity information at the client side, by enabling strong, privacy-aware access to identity information. We have also been directly involved in this project. This work influenced our approach to handle credentials based on policies and preferences. Our work on “Identity-aware Devices” focuses more on the capabilities and mechanisms that can be deployed on devices, in a federated service management context. It relies on additional services, i.e. the Registration and Provisioning Services and assumes that credentials will be disclosed and used to enable interactions, according to associated policies and preferences. Further research is going to be carried out to explore the provisioning and management of pseudo-anonymous credentials, such as IDEMIX credentials [7].

To the best of our knowledge, no commercial product, closely related to “Identity-aware Devices”, has been implemented so far. Although smartcards can be used to store and access identity tokens, they do not provide the required “Identity-aware Device” functionalities, in terms of user control, credential lifecycle management driven by policies and preferences and engagement in federated service contexts. Similarly, simple TPM enabled devices (or other types of Hardware Secured Appliances, e.g. [15]) can only be used to provide the basic security and attestation capabilities.

Microsoft InfoCard/Cardspace [8] is also important related work, as it enables users to have degrees of control of credentials with their Identity Selectors. However it primarily focuses on the direct provisioning of tokens to an Identity Selector and their usage in federated identity management context, without lifecycle management and policy-driven control of stored tokens. Cardspace currently does not support a multi-step provisioning process: credentials (or references to them) are stored in the Identity Selector at the issuance time. It might require the active involvement of an Identity Provider for accessing a service whilst in our solution this task has been delegated to the "Identity-aware Devices". Section 6 describes some additional R&D work that we are carrying out in this space.

6 Discussion, Current State and Next Steps

Our R&D work on "Identity-aware devices" is in progress. The current prototype, integrated with HP Software Select Federation solutions [5], can demonstrate only some of the "Identity-aware Devices" features. More work is required to fully implement its functionalities and explore the utilisation of this kind of device in multi-protocol, multi-framework federated services. We are advancing our vision and this technology by engaging in a second phase of the collaborative project with BT and Intel, aiming at refining specifications and having a technology trial.

Based on current experiments carried out in the Liberty Alliance framework, the usage of our prototype of an "Identity-aware Device" is reasonably intuitive and simple, as most of the underlying complexity is handled by the "Trusted Module", This module is "transparent" to the user (whilst engaging with federated services) and driven by policies and preferences. However, more experiments need to be done in heterogeneous federated frameworks.

More research is also required to explore the process of issuing and certifying "Identity-aware Devices". The current approach is that the "Trusted Module" will be produced and "certified" by trusted manufacturers. Other potential options have been explored in [20]. An open issue is about how to handle the loss of credentials, identity tokens and related policies/preferences in case of device faults on in case the device is lost or stolen. We believe that part of our previous work on "trusted migration of credentials" [9] can be leveraged to address the problem. Additional investigation is required.

We are also exploring additional extensions of the "Identity-aware Device" in the context of the EU PRIME project. We have investigated how to leverage "Identity-aware Devices" to deal with Microsoft CardSpace/InfoCard credentials [8] and ensure that they can be handled by the device via an "Identity Selector" that keeps into account complex privacy policies. A first prototype has demonstrated that it is relatively simple to handle the registration and provisioning phases of InfoCards: this has been implemented as a simple variant of the PoCv1 demonstrator. "InfoCard" credentials are yet another type of "identity tokens" that can be locally provisioned and managed by the "Identity-aware Device". At the moment these "InfoCard" credentials are provisioned, via an adaptor, to a Microsoft "Identity Selector" installed within the device. We are in the process of building our "Identity Selector", embedded in the "Identity-aware Device" (i.e. a special type of "Credential Management Engine") along with the enhanced policy management capability.

We are planning to explore alternative paradigms to implement the "Trusted Module" within an "Identity-aware Device". One of these paradigms would consist in using virtual machines (such as XEN [10]), "virtual TPMs" and compartmentalization. Applications can run in different compartments, within a virtual machine hosted by the device. The integrity of these virtual compartments is provided by associated virtual TPMs, linked by the chain of trust to a physical TPM, installed on the device. Related work in this space is carried out in the OpenTC project [11]. In this context, the "Identity-aware Device" components (hosted in the "Trusted Module") would run in a "special", very protected compartment, whilst other applications (e.g. web browser) will run in different, secluded compartments.

We are also investigating how to implement a version of "Identity-aware Devices" by leveraging HP TPM modules and enhanced HP ProtectTools solutions [12] to provide the functionalities of the "Trusted Module" , handle "authentication tokens" and support further credential management capabilities.

7 Conclusions

This paper introduced and discussed the concept of “Identity-aware Devices” in the context of online federated services. The aim is to put users in control of their credentials and identities and enable a simple, secure, trustworthy and transparent access to these services (as well as giving degrees of assurance to service providers).

A high-level model of “Identity-aware Devices” has been presented, based on the concept of having a “Local Identity Provider” and the capability of locally handling credentials and their lifecycle, base on associated policies and preferences. Trusted modules (for example based on Trusted Computing technology) have been used to provide secure storage of credentials, trust and platform integrity. Additional support services, namely the Registration and Provisioning Services have been used to enable the provisioning of credentials to “Identity-aware Devices” and part of their lifecycle management.

A full working demonstrator has been implemented in the context of a related Liberty Alliance (Advanced Clients Technologies) initiative, in collaboration with BT and Intel. This work is in progress. This paper discussed current results and open issues, along with additional R&D work to be carried out at HP Labs.

Acknowledgements

We would like to thank the BT, Intel (in particular Conor Cahill) and HP teams for their work and contributions in defining and refining the “Identity-aware Device” vision, along with their joint work in designing and building a working Proof-of-Concept. A special thank also to Daniel Drozdzewski that (during his internship at HP Labs) implemented most of the HP components of the PoCv1 prototype and demonstrator.

References

1. Liberty Alliance, http://www.projectliberty.org/, 2007
2. TCG, Trusted Computing Group TPM Specification, http://www.trustedcomputinggroup.org, Version 1.2, 2003
3. IBM, Direct Anonymous Attestation, http://www.zurich.ibm.com/security/daa/, 2004
4. Liberty Alliance, Advanced Client Technologies, http://www.projectliberty.org/resource_center/specifications/liberty_alliance_id_wsf_advanced_client_1_0_specifications, 2007
5. HP, HP OpenView Select Federation, http://h20229.www2.hp.com/products/slctfed/index.html, 2007
6. EU PRIME Project, Privacy for Identity Management in Europe, https://www.prime-project.eu/, 2004-2008
7. IBM, IDEMIX, http://www.zurich.ibm.com/security/idemix/, 2002
8. Microsoft, CardSpace, http://netfx3.com/content/WindowsCardspaceHome.aspx, 2007
9. Pearson, S., Casassa Mont, M., Novoa, M., Securing Information Transfer in Distributed Computing Environments, pp. 34-42, January/February 2008
10. XEN, XEN virtualization software, http://xen.xensource.com/, 2007
11. OpenTC, Open Trusted Computing initiative, http://www.opentc.net/, 2008
12. HP, “HP ProtectTools Security Solutions”, http://h20219.www2.hp.com/services/cache/45782-0-0-225-121.aspx, 2006
13. Liberty Alliance, ID-FF Specifications, http://www.projectliberty.org/liberty/specifications__1, 2008
14. Liberty Alliance, ID-WSF Specifications, http://www.projectliberty.org/liberty/specifications__1, 2007
15. Baldwin, A., Shiu, S.: Hardware Encapsulation of Security Services, ESORICS 2003, 2003
16. Pearson, S. (ed): Trusted Computing Platforms, Prentice Hall, 2002
17. OpenID, OpenId, http://openid.net/, 2008
18. Higgins, Higgins Project, http://www.eclipse.org/higgings/, 2008
19. Casassa Mont, M., Dealing with Privacy Obligations: Important Aspects and Technical Approaches - TrustBus 2004, 2004
20. Casassa Mont, M., Balacheff, B., On Device-based Identity Management in Enterprises, 4th International Conference on Trust, Privacy and Security in Digital Business 2007, TrustBus 2007, 2007