Requirements Analysis for Identity Management in Ambient Environments: The HYDRA Approach

Hasan Akram, Mario Hoffmann

Fraunhofer Institute for Secure Information Technology,
Darmstadt, Germany
{hasan.akram;mario.hoffmann}@sit.fraunhofer.de

Abstract. The research field of Ambient Environments and Ubiquitous Computing aims toward the future vision of intelligent mobile and wireless network scenarios. In such environments where the wireless network consists of numerous nodes, like intelligent devices, sensors and mobile devices, a highly secured and well defined Identity (ID) Management System is required that deals with issues like virtual and temporary identities of users and devices as well as users' awareness in information disclosure and privacy. One major goal of the EU-project HYDRA[1] ("Networked Embedded System middleware for heterogeneous physical devices in a distributed architecture") is the support of developers of such ambient environments to manage context sensitive identity information and assure integration and interoperability of existing ID Management approaches. Based on this project in this paper we identify and analyze ten requirements for a middleware architecture to create a bridge between existing identity management technologies and also allow a framework to make them available for application developers of ambient environments.

Keywords: Identity Management, Ambient Environments, Security by Design, Privacy Protection, Identity Metasystem, HYDRA ID

1. Introduction

"7 trillion wireless devices serving 7 billion people in 2017" states the website[2] of the Wireless World Research Forum. This vision reflects the increasing trend of introducing micro- and nano-sized computers to everyday devices and tools (Ubiquitous Computing, Internet of Things). However, in such ambient environments not only computer systems become transparent and ubiquitous to users but also the users and their contexts become transparent and ubiquitous to the systems running in the background. And the more computers become transparent and ubiquitous the more the users' privacy and control is at stake.

---

[1] HYDRA: Networked embedded system middleware for heterogeneous physical devices in a distributed architecture. http://www.hydra.eu.com (2007) contract number: IST-2005-034891, duration: 07/2006-06/2010.

[2] WWRF: http://www.wireless-world-research.org

One necessary measure counteracting this rising challenge is a well defined combination of identity management for and virtualisation of both users and devices supported by future middlewares. This paper, therefore, focuses on a comprehensive requirements analysis for Identity Management in ambient environments and introduces recommendations for future system middleware architectures.

One of the main objectives of this paper is to draw a boundary of Identity Management support at middleware level in the context of HYDRA (Section 3). It is important to note that even a developer works keeping an end-user in mind. Therefore, as primary input for deriving middleware level requirements, the HYDRA home automation scenario is taken (Section 2), which basically illustrates application level use cases.

Taking advantage of the basic concepts of Kim Cameron's *Identity Metasystem* [1, 2], we show an extended application of the *Identity Metasystem* and derive ten particular requirements for Identity Management in ambient environments referring back to our HYDRA home automation scenario.

2. HYDRA Test Scenario: Intelligent Home

The HYDRA project uses the IDON method [6] for futuristic scenario definitions. By means of this systematic approach, fictitious scenarios have been derived in three domains - building automation, healthcare, and agriculture, which are likely to be practiced in reality in 2015 [3]. Many of these scenarios are derived from business cases from the perspective of an end-user, i.e. from application level. As a consequence Identity Management can have a large range of implications to information systems encompassing scenarios of access control, *Single Sign On (SSO)* in single and cross organizational domains, virtual identity, identity life cycle, session management and many other related issues. However, in case of designing a middleware for identity management the perspective of requirements analysis shifts from the end-user to a developer. This results in a different set of development time use cases from the very same application use cases.

With the intention to illustrate the necessity of an Identity Management System in HYDRA middleware we will take as a basis a detailed technical scenario of a heating system breakdown at "Krøyers Plads" housing complex located in Copenhagen that deploys the "Hydra Building Automation System" (HBAS) [5]. The resident living in a new flat in this building complex is equipped with automated lamps, computers and a wireless network, as well as a Hydra-enabled heating system and many other usual sets of automated devices. While the resident is at his office, the heating system of the flat breaks down and the water pressure rapidly decreases down to a level that is detected as an emergency situation by the HBAS which is shown as legend 1 in figure 1. As a result of that HBAS sends out an alert message to the resident (legend 2 in figure 1).

In order to get the heating system fixed as soon as possible the resident chooses a service provider from a list of providers matching the emergency requirements and his preferences best. The service provider then sends a service agent (e.g. a specialized technician) to the house. The challenge here is to allow a particularly

authorized service provider and his technician remotely to get into the house to fulfill a specific task. Therefore, included in the repair order a specifically restricted HBAS authorization ticket guarantees that in this case a service agent can enter the flat and get access to the heating system (legend 3 and 4 in figure 1). After entering the flat upon successful authentication procedure the service agent gets authorization to access additional context aware information required to perform his job (legend 4 in figure 1).

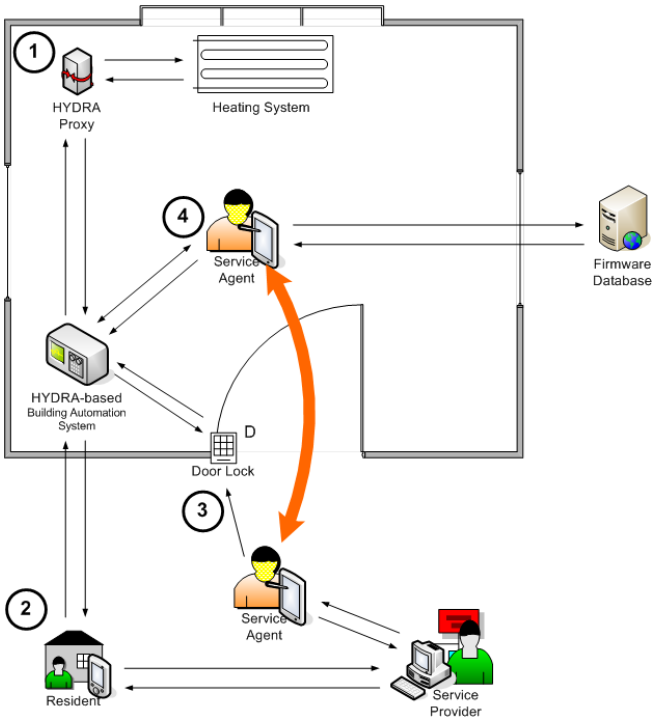

Fig. 1. Sequence of steps for the technical scenario [3].

This representative scenario can be basically adopted by many kinds of similar scenarios of remote authorization such as large housing areas with housekeeping service, office buildings with restricted access, airports, and hospitals. Thus, with the basic scenario of HYDRA being illustrated we can go one step forward in our process of HYDRA identity requirements analysis. In the next section we will provide a bird's eye overview of the *Identity Metasystem* and show how it relates to HYDRA use cases, which will be our basis on deriving HYDRA Identity Manager (HIM) requirements.

3. Application of Identity Metasystem in Extended Use Cases of the Hydra Scenario

This section is the basis for our requirement analysis process. The objective here is to establish the connection of *Identity Metasystem* and the given HYDRA scenario (Section 2). We start with the introduction of *Identity Metasystem* which is followed by an elaborate use case analysis focusing federated identity.

3.1 Identity Metasystem

Identity Metasystem [1, 2, 10, 11] is a claim[3] based architecture for an identity layer proposed by Kim Cameron in 2005 that uses federated identity as its underlying principle. The main goal of this architecture was to introduce an identity layer for the Internet that decouples Identity Management layer from the rest of the other layers in applications. Identity Metasystem is designed to be technology agnostic.

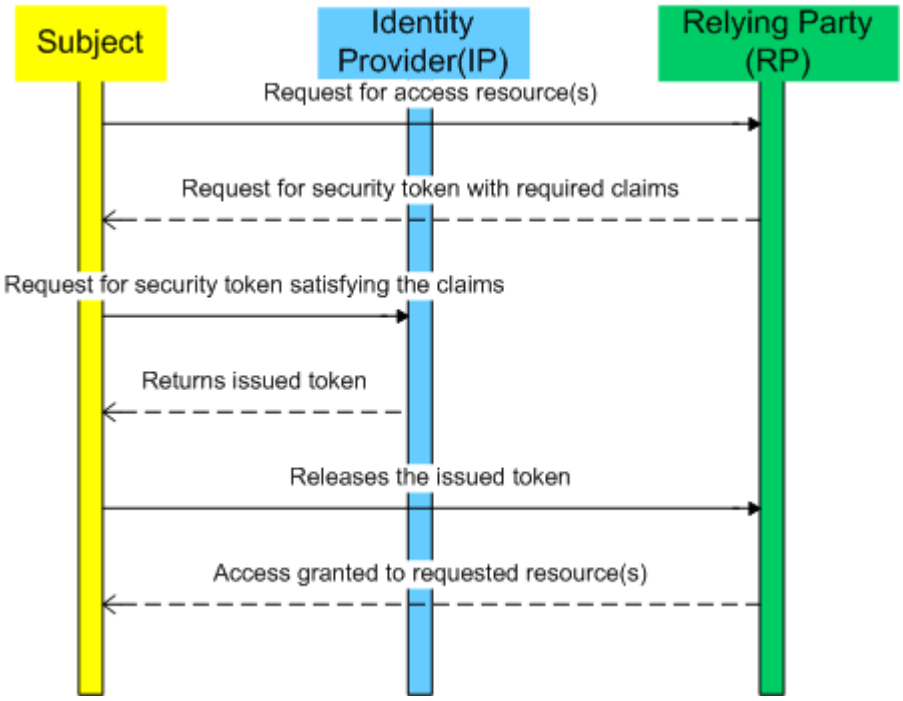

Fig. 2. Simplified sequence diagram of Identity Metasystem

Figure 2 illustrates a simplified version of the sequence diagram of Identity Metasystem. There are three roles in Identity Metasystem – the *Subject* (the user

[3] A claim is a declaration made by an entity. Examples include name, identity, key, group, privilege, and capability. - OASIS standard *Web Services Security: SOAP Message Security 1.0* [13].

whose identity is concerned), a *Relying Party*[4] (RP) and an *Identity Provider*[5] (IP). We can see in the sequence diagram that the Subject requests to have access to (a) particular resource(s) of a RP. The RP sends a set of claims or its identity requirements needed to access the requested resource(s) back to the *Subject*. The subject checks which IP(s) is suitable for this particular set of claims and chooses an IP. The *Subject* sends a security token request to the chosen IP. The IP issues and returns a security token satisfying the claims to the *Subject*. The *Subject* releases this token to the RP and gains access to his desired resource(s).

Based on this fundamental principle of Identity Metasystem, we will now analyze extended use cases of the Hydra in the next sub-section.

3.2 Extended Use Case Analysis of the Hydra Scenario

In this sub-section we will see use cases in HYDRA home automation scenario where the propagation of authentication information from entity to entity is based on contractual relationship. We will also observe how the three roles of Identity Metasystem explained in section 3.1 – *Subject*, *IP* and *RP* – shifts from endpoint to endpoint.

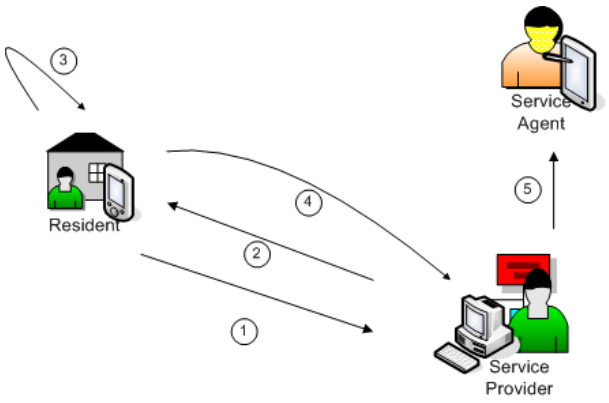

Fig. 3 Sequences in the process of the resident authenticating himself to the service provider and the service provider issues a cosigned token to the service agent.

Let us start with the use case shown in figure 3. This is the first identity federation in our scenario. In step 1 the resident is sending a request to the service provider for a service agent to be sent to his flat to fix his heating system. In step 2 the service provider is asking for his credential as a set of claims. Here the resident has an option to choose an IP that can satisfy the claims from the RP that happens to be the service provider in this case. For simplicity we assume that the resident himself is able to issue himself an identity token that satisfies the claims and would also be accepted by the RP. So, in step 3 the resident issues himself a token and in step 4 releases it to the service provider. After receiving this token the service provider issues a cosigned

[4] In federated Identity Management Relying Party is an entity that requests a digital identity of the user in form of a set of claims, issued from an Identity Provider.

[5] An Identity Provider is a trusted party that provides digital identities. It can be a third party or the user himself or even the Relying Party to whom the identity is to be disclosed.

token to a service agent (step 5) who is to be sent to the resident's flat for repairing the heating system.

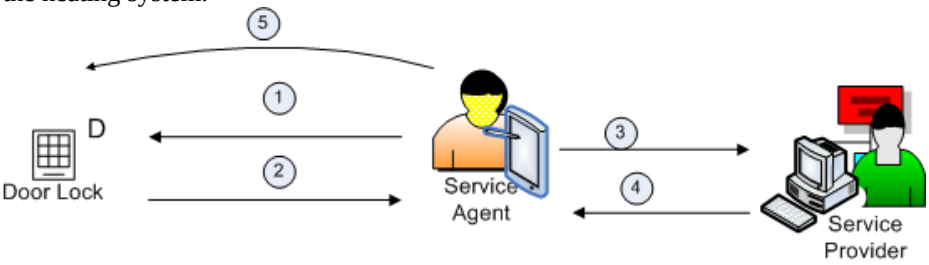

Fig. 4 Sequences in the process of the service agent getting authenticated by the door lock of the resident.

Now, let us look at another use case scenario shown in figure 4, where the service agent has to authenticate himself at the door lock of the resident's apartment. The roles - subject, RP and IP are shifted to the service agent, the door lock and the service provider correspondingly. Here, in step 1 the service agent sends a request to the door lock for accessing the flat. In step 2 the door lock sends a request for a security token as a set of claims. The service agent requests his IP (the service provider in this case) for a security token satisfying the claims. The service provider issues a token in step 4 and in step 5 the service agent releases this token to the door lock. Due to transitivity of authentication information flow as a part of the contract between the resident and the service provider, the door lock accepts his request; i.e. the door lock accepts authentication assertion (in form of a security token) from the resident, the resident sends the token to the service provider and the service provider issues a cosigned token to the service agent, consequently the door lock accepts the authentication information of the service agent. This process is repeated in each identity discovery taking place in the scenario.

Having these extended use cases being shown we can now derive specific *Identity Requirements* for the HYDRA middleware. Using this technical scenario and security requirements engineering of Hydra [9] we conceptualize the requirements of an *Identity Manager* being part of the Hydra middleware in the next section. Moreover, we identify and define what – from a *Security by Design* perspective – a developer may expect from Hydra middleware while developing an Identity Management System for ambient environments.

4. Identity Requirements in HYDRA

The illustration of the Hydra home automation scenario and the extended use case analysis with relation to Identity Metasystem make it obvious that developers of such ambient environment applications will have expectations up to a certain degree getting support for identity management processes by the middleware. Based on the given scenario, Hydra security, trust and privacy requirements as well as the trends toward identity requirements (e.g. Kim Cameron's Laws of Identity [1]) we define the following requirements for HIM (Hydra Identity Manager).

Definition: In Hydra an identity is assigned to any kind of entity, such as users, devices, and applications, being part of a Hydra-enabled infrastructure. An identity in Hydra typically comprises

(1) virtual temporary identifiers, e.g. specific Hydra IDs (HIDs) for (virtual) devices,

(2) an open list of specifying attributes, such as user preferences or device capabilities (e.g. stored in a device ontology), as well as

(3) a – if so timely restricted – history list of access events to and from this entity.

Sub-identities contain particular subsets of a Hydra identity depending on specific context information and collaboration partners. An entity may have different sub identities for and even in a specific context as long as a specific action still can be performed.

In a service oriented architecture Hydra's Identity Management System provides support to the developer to implement integrity, confidentiality and authenticity of such context specific actions, e.g. in work flows, transactions and processes performed by orchestrated services.

1. User Empowerment: Awareness and Control

The first identity requirement of HYDRA concerns the user in an ambient environment and emphasizes on two key words – "awareness" and "control". In a transaction taking place between two entities in HYDRA each entity must have full knowledge regarding the information he or she is about to disclose and to whom he or she is about to disclose. Besides having full knowledge about the information disclosure the entities must also have full range of control power to decide whether to disclose a particular set of information or not [1, 11].

An Identity Management System that does not confirm this need will suffer from serious security flaws. Lack of knowledge about the party to whom information is sent raises probability of phishing attacks. Information disclosed without knowledge of the user and lack of control of the user to decide what to disclose, violates his or her privacy.

2. Minimal Information Disclosure for a Constrained Use

Let us focus on our Hydra building automation scenario (Section 2) in order to clarify the second Hydra identity requirement. We have already stated that there is a contractual relationship between the resident and the service provider. Therefore, authentication information propagates in a transitive fashion to the service agent; i.e. since the agent is authenticated by the service provider, he is also authenticated by the resident and all the Hydra enabled devices in his or her apartment. In the process of fixing the heating system, the service agent will need to have access to certain information, e.g. the usage pattern of the heating system. The service agent will request the information needed in form of a set of claims. Here the service agent must be provided with a minimal information set that is only relevant for fixing the heating system. The usage pattern of the heating system supplied by the Hydra enabled

devices to the service agent must somehow guarantee that no other information is retrievable from it that goes beyond the necessity of fixing the heating system, e.g. the service agent should not be able to figure out from the usage pattern that during which period of the year the resident makes holiday or remains out of the flat.

3. Non-repudiation:

The term "Non-repudiation" has a traditional legal meaning and at the same times has a different meaning in terms of digital security [14]. We will focus on the latter meaning of "Non-repudiation" and then relate its necessity to our Hydra scenario (Section 2). In a crypto-technical sense transfer of data from one entity to another must guarantee authenticity, integrity and a time stamp, so that neither of the parties involved can deny that the transfer of the data took place.

Let us examine the issue of authenticity within the scope of the Hydra Scenario. The endpoint of the service provider receiving a message from the endpoint of the resident must know if the message is really transmitted from the resident or if it is under a spoofing or masquerade attack [5]. Therefore, there is a need of mechanism(s) that guarantees identity preservation.

To illustrate integrity, we continue with our running scenario example: the service provider receives a message from the resident over HTTP, he must guarantee the integrity of the message content. From a middleware viewpoint, there must be supports that allow the developer ensuring that the messages sent from one node to another is not being changed in an intermediary node or not under falsification attack [5]. To guarantee integrity it is also important that any kind of message manipulation has to be detected [15].

Another vital point is to make sure a time stamp is attached to the message. This is required to combat replay attacks. A time stamp attached to the message will make the message valid only for a certain period of time and as a result of that lower the probability of replay attacks.

Thus, we can sum up by saying that unforgeable identity, non-falsifiable message exchange, and provision of a time stamp is required in Hydra so that the identity of the sender and the integrity of the message cannot subsequently be refuted.

4. Support for directional identity topologies:

In the domain of ubiquitous computing communication takes place in various topologies and so does identity exchange. For example, when the resident communicates with the service provider (figure 3), it is a simple endpoint to endpoint (point to point or peer to peer) identity exchange. In the same scenario when the service provider comes into the flat of the resident, he or she needs to transmit his or her identity in a broadcast topology (figure 2) so that the Hydra enabled devices can detect his or her presence and take necessary actions. In the first example identity is unidirectional and in the latter case it is omni-directional. Based on this need of directional identity [1] the Hydra Identity Management middleware has to have supports for the following directional identity exchange topologies: 1. Broadcast (omni-directional) 2. Point to point (unidirectional) 3. Multicast (omni-directional and/or unidirectional).

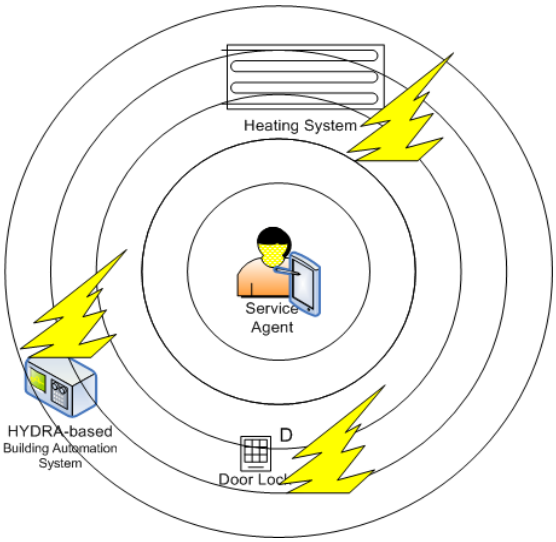

Fig. 5 The service agent transmits his or her identity in an omni-directional manner. The Hydra enabled devices senses his or her presence.

Fig. 6 Identity exchange taking place between the resident and the service provider. They both transmit their identity to each other in a unidirectional way.

Figure 5 and figure 6 illustrate omni-directional and unidirectional identities in the Hydra building automation scenario. In figure 5 the presence of the service agent is being sensed by the Hydra enabled intelligent devices at the resident's apartment, while the service agent transmits his identity in broadcast topology. In figure 6 the topology of unidirectional identity is shown in the use case where the resident and the service provider are exchanging their identities (point to point).

Following the notion of device discovery and service discovery in Hydra, we propose an identity discovery similar to figure 5 and 6. At Hydra middleware level device and service discovery will be transparent to a developer, he would rather be facilitated with identity discovery supports which is relevant for his identity layer of the application.

5. Universal Identity Bus:

Interoperability is one of the high level requirements of the Hydra project [4, 8]. Consequently, the Hydra Identity Management System inherently requires supporting interoperability between the garden varieties of Identity Management technologies available from different vendors. We propose a Universal Identity Bus (UIB) that will provide vendor to vendor interoperability functionalities. In order to achieve this

requirement the Hydra Identity Manager must support UIB that works as a bridge between different Identity Management technologies.

6. Provision of defining strength of identity:

In order to illustrate why Hydra necessitates provision of weak identities and strong identities, it is important to get back to the definition of identity in the context of Hydra shown in Section 2. We have seen that in Hydra identity relates to a person, a device or an application. Let us look at a case where a device is owned by a person. Here, the identity of the device is somewhat depended on the identity of the person, i.e. the identity of the device is incomplete without relating it to an identity of another entity. In a similar way many use cases may arrive where an identity does not suffice itself without being depending on an identity of another entity. Based on this criteria identity can be categorized to be strong (independent), weak (dependent) or somewhere in the middle. Thus, we can justify the requirement of a provision of having strength of an identity in the Hydra middleware. It is important to note that weak identities and strong identities are not the same as sub-identities (Section 2), which are basically subsets of identities. Identities or sub-identities both can be rated by their strength depending on their degree of being autonomous.

7. Decoupling identity management layer from application layer:

This requirement builds up another block on top of the "*Universal Identity Bus*" and separates the application layer from Hydra Identity Management layer. This is obligatory for the Hydra Identity Manager for two main reasons: 1) organizations are being able to change their identity policies without having an impact on the business layer and 2) the developers have an environment where they can work on the identity layer being transparent of the business layer or vice versa.

8. Usability issue concerning identity selection and disclosure:

We have already emphasized on the issue of empowerment of the user in case of revealing information in our first Hydra identity requirement. Lack of usability will make requirement 1 almost impossible to take place. In a user-centric design the user is the ultimate procurer and a methodic requirement specification of usability keeping the procurer in mind is unavoidable [12]. Therefore, HIM must facilitate the developer with adequate support for implementing usability.

9. Consistent experience across contexts:

Context is one of the major concerns in Hydra test scenario (Section 2) and identity and context are closely related. Therefore, while analyzing HIM requirements the issue of context is considered. In Hydra an entity and its identity will have an n to m relationship, i.e. one entity can have multiple identities and one identity can be possessed by several entities. For example, the resident has several identical sets of devices and he wants to use them with one single device identity. In this example one identity is shared by multiple entities. The example one entity having multiple identities would be, the resident has an identity at his work, a different one for his shopping web sites and another different one for heating system repairing service providers. In this n:m relationship of identities and entities it is very important to have consistence experience for the user depending on contexts.

Along with the consistencies among context, the identities provided in different contexts should also be independent of each other, i.e. the identity the user provides at work should not be related to his identity for his shopping website and vice versa.

10. Scalability:

In an ambient environment the number of nodes joining in and out is dynamic and thus the necessity of scalability in managing the identities of these numerous nodes is inevitable.

What does this at this point mean to a developer?
Let us look at the matter from a middleware point of view. A developer who will be working on an Identity Management System for similar scenarios defined in section 2 will have expectation from the middleware for supports so that he can ensure the requirements stated above. Our objective is to present the developer such an environment where the underlying technologies and other layers of the Hydra middleware are transparent to the Identity Management layer and the developer is able to totally focus on his Identity Management related needs.

6 Comparison with related works

Identity Management in pervasive computing has been explored by researchers since almost the very beginning of pervasive computing. Requirements and principles of Identity Management has been analyzed and derived based on certain needs in certain scenarios. Obviously, these related works have some commonalities and disparities among themselves. In this section we briefly report a comparative study of our work with respect to a few selected related works in Identity Management in ubiquitous computing. In this comparative study we also highlight a justification of our proposal of requirements rather than choosing one of the existing works.

There are related works where they deal with application level requirements analysis for identity preservation in pervasive computing, e.g. the requirements proposed by Roy Champbell [17]. In their paper they have shown requirements of security in such ambient scenarios. Although there are partial overlaps with our requirements, there is a fundamental difference of looking at the problem from application perspective and a middleware perspective.

Privacy principles described by Langheinrich [18] also have some overlaps and as well as differences compared to our work. The differences and similarities are due to the fact that identity is a broader concept and it comprises many other elements including privacy.

Jendricke [16] proposed context driven Identity Management to comply with principles of Langheinrich [18] and illustrated their architecture and prototype. Again, it is designed from an application viewpoint. Moreover, the solution proposed in this paper is not federation driven. We have already seen in the extended use cases (Section 3.2) that Hydra scenario is federation driven. Therefore, this solution was also not totally pluggable to our need.

Kim Cameron's [1] laws of identity is also not fully meant for ambient environment and focused on the present situation of internet, whereas Hydra scenario

is focused on a projected scenario in 2015. Therefore, the seven laws stated in his white paper do not totally suffice our needs as well. However, one fundamental principle that is common with our scenario is the concept of federation. Identity metasystem [2, 9] architected by Kim Cameron is a federation based concept and is very much applicable to Hydra scenarios. This motivation led us to apply the concept of identity metasystem in our use case analysis (Section 3.2) and derive HIM requirements based on the use case analysis.

Finally, I would like to sum up by saying that if a Venn diagram is constructed for all the sets of laws of Identity Management proposed so far, there will always be an intersection region. At the same time there can be regions in each of these sets which are non-overlapping. This is simply because all these laws are based on some variable parameters; namely - perspective, time, computing environment etc.

7 Conclusion & Outlook

In this paper we have illustrated the requirements analysis of Identity Management in futuristic scenarios of ambient environment or ubiquitous computing from a middleware viewpoint. We have shown the significance of Identity Metasystem [1, 2, 10, 11] in such futuristic scenarios and also seen the propagation of authentication in federated Identity Management. The following list summarises the requirements identified and analysed:

1. User Empowerment: Awareness and Control
2. Minimal Information Disclosure for a Constrained Use
3. Non-repudiation
4. Support for directional identity topologies
5. Universal Identity Bus
6. Provision of defining strength of identity
7. Decoupling identity management layer from application layer
8. Usability issue concerning identity selection and disclosure
9. Consistent experience across contexts
10. Scalability

The future goal of this research work is an evaluation of the state of the art technologies that best suites the requirements and eventually derive an architecture based on the requirements analysis of this paper. The results of the architecture specification for ambient environments will be published soon. Choosing the best suited technology for Identity Management, we plan to build the Hydra Identity Management SDK as a set of service library and integrate it into the HYDRA middleware. This part will be published by the end of this year.

Acknowledgements

We would sincerely like to thank Julian Schütte (Fraunhofer Institute for Secure Information Technology, Darmstadt, Germany) for his review and helpful remarks on this paper, which certainly added value to the entire work.

References

1. Cameron, K, Laws of Identity (2005), Microsoft Corporation.
2. McLaughlin, L., What Microsoft's identity metasystem means to developers, *Software, IEEE* , vol.23, no.1, pp. 108-111, Jan.-Feb. 2006.
3. HYDRA, Deliverable D3.3 Draft of architectural design specification, 11 May 2007, Version 1.3.
4. HYDRA, Deliverable D7.1 Security Requirements Specification, 8 March 2007, Version 1.0.
5. HYDRA, Deliverable D2.1a Scenarios for usage of Hydra in Building Automation, 25 January 2007 - version 1.41.
6. Galt, Chicoine-Piper, and Hodgson (1997). IDON Scenario Thinking: How to Navigate the Uncertainties of Unknown Futures. IDON Ltd.
7. The Hydra Project, http://www.hydra.eu.com
8. Hoffmann, M., Badii, A, Engberg, S., Nair, R., Thiemert, D., Mattheß, M., Schütte, J., "Towards Semantic Resolution of Security in Ambient Environments", *Ami.d – 2nd Conference for Ambient Intelligence Developments*, September 2007
9. Cameron, K., Jones M. B., Design Rationale behind the Identity Metasystem Architecture, http://www.identityblog.com/, http://research.microsoft.com/~mbj
10. J. Miller, Yadis 1.0, http://yadis.org/papers/yadisv1.0.pdf, March 2006
11. Bertocci, V., Serack, G., Baker, C., Understanding Windows CardSpace: An Introduction to the Concepts and Challenges of Digital Identities, December 27 2007, Addison-Wesley.
12. Artman, H. 2002. Procurer usability requirements: negotiations in contract development. In *Proceedings of the Second Nordic Conference on Human-Computer interaction* (Aarhus, Denmark, October 19 - 23, 2002). NordiCHI '02, vol. 31. ACM, New York, NY, 61-70. DOI= http://doi.acm.org/10.1145/572020.572029
13. *WS-Trust 1.3*, OASIS Standard 19 March 2007, http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.html#_Toc162064937
14. McCullagh, A., Caelli, W., Non-Repudiation in the Digital Environment, *First Monday, volume 5, number 8* (August 2000), URL: http://firstmonday.org/issues/issue5_8/mccullagh/index.html
15. Müller, G., Rannenberg, K., Multilateral Security in Communications, Vol.3, Technology, Infrastructure, Economy**,** Addison-Wesley, 15. July 1999.
16. Jendricke, U., Kreutzer, M., Zugenmaier, A., 2002. Pervasive privacy with identity management, *In Proceedings of the Workshop on Security in Ubiquitous Computing, UbiComp* 2002, Sweden.
17. Campbell, R, Al-Muhtadi, J., Naldurg, P., Sampemane, G., and Mickunas, M. D. Towards security and privacy for pervasive computing. *In Proceedings of International Symposium on Software Security*, Tokyo, Japan, 2002.
18. Langheinrich, M., Privacy by Design – Principle of Privacy-Aware Ubiquitous Systems, *Proceedings of the UBICOMP* 2001.