Design of Authentication Model Preserving Intimacy and Trust in Intelligent Environments

Benchaà Djellali, Abdellah Chouarfia, Kheira Belarbi, Pascal Lorenz

2015, Volume 7, Issue 1

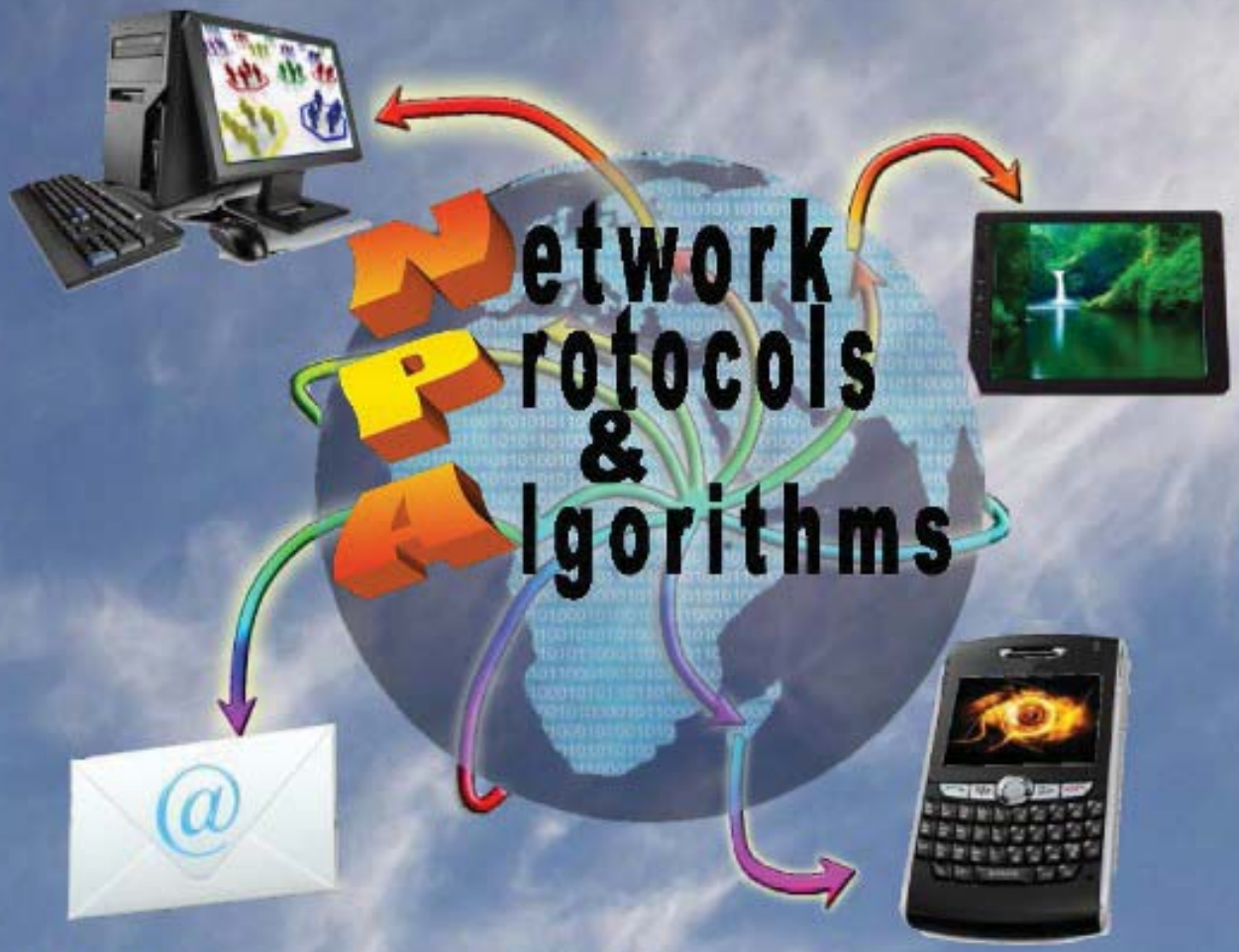

Table of contents:

http://www.macrothink.org/journal/index.php/npa

Macrothink Institute™

Macrothink Institute™

Network Protocols and Algorithms
ISSN 1943-3581
2015, Vol. 7, No. 1

Design of Authentication Model Preserving Intimacy and Trust in Intelligent Environments

Benchaa Djellali[1,2], Abdallah Chouarfia[1], Kheira Belarbi[1]
[1]Department of Computer Sciences, University of Sciences and Technology of Oran
31000, Oran, Algeria
E-mail: djellali_ben@yahoo.fr, chouarfia@mail.com, belarbi_khe@yahoo.fr

Pascal Lorenz[2]
[2]GRTC, University of Haute Alsace
68000, Colmar, France
E-mail: Benchaa.djellali@uha.fr, lorenz@ieee.org

Received: March 8, 2015 Accepted: April 6, 2015 Published: April 30, 2015

DOI: 10.5296/npa.v7i1.7208 URL: http://dx.doi.org/10.5296/npa.v7i1.7208

Abstract

With the recent advances in communication technologies for low-power devices, pervasive computing environments (PCE) spread as new domains beyond legacy enterprise and personal computing. The intelligent home network environment is thing which invisible device that is not shown linked mutually through network so that user may use device always is been pervasive. Smart devices are interconnected and collaborate as a global distributed system to infuse intelligence into systems and processes. This kind of environment provides various smart services and makes consequently an offer of convenient, pleasant, and blessed lives to people. However, the risk is high as long as the offer is pleasant and convenient. In such context, security is still very fragile and there is often a violation of user privacy and service interference. For this, a special interest in ubiquitous network security is going up. Safety lies primarily in the authentication of users accessing the network. It guarantees that only legitimate users can login and access to services indoor the network. In this paper, we propose an anonymous authentication and access control scheme to secure the interaction between mobile users handling smart devices and smart services in PCEs. In an environment based on public key infrastructure (PKI) and Authentication, Authorization, and Accounting (AAA), the proposed authentication protocol combines both network authentication technique based on symmetric keys and single sign-on mechanisms. The authentication protocol is simple and secure, protects the privacy of user and aims to satisfy the security requirements.

Keywords: Anonymity, Authentication, Authorization, Intelligent environment, Pervasive Computing Environment.

1. Introduction

The lower cost of components and their miniaturization make possible a world in which the electronic is likely to be incorporated into any object. Technically, the addition of a chip in an object doesn't represent any particular difficulty and economically, add a chip and an embedded small program in an object is not a commercially prohibitive cost. In term of use, the additional service rendered is simple, easily discernible by the user and is quite justified. Thus, instant communication implementation to our service of panels indicators, screens or communication devices as soon as we step across the threshold of a home, a hotel bedroom , a warehouse or a public space is the essence of ambient intelligence [1,2] and pervasive networks [3].

The ubiquitous network [5, 6] is the support of transparent collaboration between equipment which constitute it collectively and permanent cooperation of the network of personal objects of every individual who crosses its threshold. The ubiquitous network is a network of continuity which must, as the origin of its name indicates, be present everywhere, all the time and this without breaking. Nevertheless, in an environment where by default each object will be connected and accessible, arise necessarily issues of confidentiality, privacy and non-intrusion [4].

Pervasive computing environments with their interconnected devices and services promise seamless integration of digital infrastructure into our everyday lives. While the focus of current research is on how to connect new devices and build useful applications to improve functionality, the security and privacy issues in such environments have not been explored in any depth [7].

A pervasive network includes a variety of network protocols and is expected to support many service models such as a client-server model, a peer-to-peer communication model, and hybrid model. For that, it is difficult to definitely decide which mechanism is suitable for pervasive network. Despite all this, to allow only legitimate users in PCEs, securisation of interaction between mobile users and services can be performed by various methods namely ID-password-based authentication method, certificate-based authentication method, or biometric information-based authentication method [8].

While traditional distributed computing research attempts to abstract away physical location of users and resources, pervasive computing applications often exploit physical location and other context information about users and resources to enhance the user experience. The same features that make pervasive computing environments convenient and powerful make them vulnerable to new security and privacy threats. Control’s gaining of users’ devices by hacker, eavesdropping of communications channels, modification of sensitive m-commerce transactions, transaction of services or goods in other party’s identities are examples of threats taking advantage of ubiquitous communications dynamism and facing the ubiquitous environment. Traditional security mechanisms and policies may not provide adequate guarantees to deal with the new exposures and vulnerabilities.

In this context, being particularly interested by strengthening security and preserving

privacy into pervasive computing environments, we propose in this paper an authentication model for pervasive computing environment (PCE) based on public key infrastructure (PKI) and Authentication, Authorization, and Accounting (AAA). The authentication protocol which corresponds to such infrastructure combines both authentication technique based on symmetric keys and single sign-on mechanisms.

The remainder of this paper is organized as follows: In section 2, we discuss the Pervasive Computing Environment (PCE). In section 3, we attempt to understand the security in PCE by identifying issues and challenges. In section 4, we present an infrastructure model and an authentication protocol for PCE security. We provide also in this section an evaluation of the approach. We conclude in section 5 by highlighting the on-going potential areas of future research on security protocol model for ubiquitous networks.

2. Pervasive Computing Environment

Microprocessors are embedded in the everyday object we use but we are largely unaware of it. Marc Weiser [10] put forward the view that ubiquity will have been achieved only when computing has become invisible and there is intelligent communication between the objects that anticipate our next move. After that, technology has advanced along many dimensions, especially in hardware progress and wireless communication technologies. A number of leading technological organizations are exploring Pervasive Computing Environment. But it is far from Weiser's vision becomes reality. Pervasive Computing will be the future. Pervasive computing will be a fertile source of challenging research problems in computer systems for many years to come [4, 11].

Pervasive computing will surround users with a comfortable and convenient information environment that merges physical and computational infrastructures into an integrated habitat. This habitat will feature a proliferation of hundreds or thousands of computing devices and sensors that will provide new functionality, offer specialized services, and boost productivity and interaction. In an active space which is a dynamic information-rich space, individuals may interact with flexible applications that may follow users, define and control space function, or collaborate with remote users and applications [7].

2.1 Heterogeneous Composition

The ubiquitous network is a combination of technologies and services offered by the cable, wired and mobile telephony, wireless and satellite which could quickly lead to reliable and complete network coverage. The tools deployed indoor the pervasive network vary between the smallest device with reduced autonomy and capacity of processing and storage, and the sophisticated, powerful and very fast computer. The ubiquitous network infrastructure is conceived to offer ideal conditions of interconnection of variety of heterogeneous components, so that services and applications are accessible at anytime, anywhere and in any condition of the network environment [4, 9].

2.2 Dynamic and Self-Organizing features

A pervasive network is characterized by a self-organisation and dynamism of offer and demand: From a wide choice of suppliers, it offers a wide variety of services (see figure 1). These services could be utilized by a variety of different ubiquitous network users' devices. Ubiquitous network users move easily in the network and enjoy a dynamism which enables them to join or leave instantaneously the network. They can travel from one network to another without obstacle. But, each network has its management peculiarity and its security policy. For that, the passage from one network to another triggers automatically a dynamic reconfiguration in order to make it possible to transiting users to take advantage of the offerings of the networks of which they cross [9]. A service provider may at any time become a user of other services and vis versa a service consumer can become in turn a service provider. To note that certain services provided by the ubiquitous network as TV, multimedia and video on demand, require a quality of service which should be supplied by the ubiquitous network or the network on which is built the ubiquitous network [4].

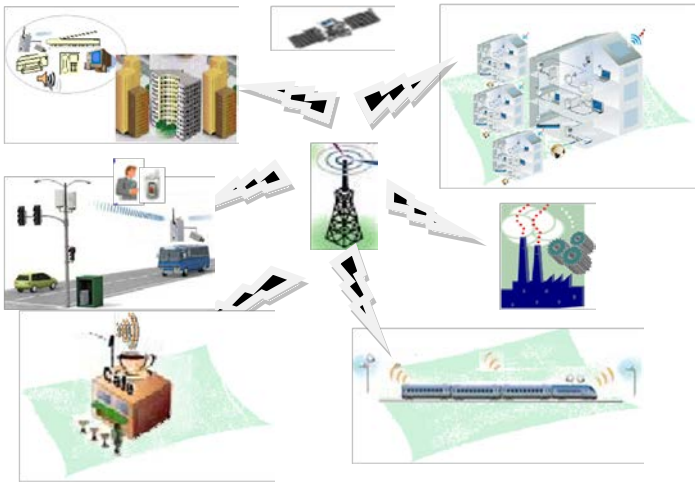

Figure 1. Pervasive Computing Environment

3. Security in Ubiquitous Networks

3.1 Problem

With the evolution of communication systems, mobile devices require access to an increasing number of services. Using a range of communications technologies, a wide range of services could be provided to a variety of ubiquitous computing devices. Ubiquitous telecommunications systems will allow heterogeneous wired and wireless access to a vast range of services. Increasingly growing, users' requests and services offers led to the collaboration of peripherals of various horizons. It drove to the contribution of several ubiquitous computing environments, namely the personal, the home, the office and the vehicle environments.

Current research in pervasive computing focuses on building infrastructures for managing active spaces, connecting new devices, or building useful applications to improve functionality.

However, security and privacy issues in such environments have not been explored in depth. Indeed, several researchers and practitioners have admitted that security and privacy in this new computing paradigm are real problems [7].

For this, there is a need to work towards the development of secure ubiquitous applications and the provisioning by secure environment to operate on. Ubiquitous network infrastructure will require the provision of certain degree of security between participating user devices. Different degrees of trust may be required for different users and their devices to access services in ubiquitous networks. These will be reflected in the ubiquitous network record and resources to determine whether the users and their devices are authorized to access. Security architecture for ubiquitous network environment should be designed to allow safe execution of trusted applications [9].

The security of pervasive computing environment refers to establish mutual trust between infrastructure and device in a manner that is minimally intrusive. In such environment, a close relationship binds any smart device to its owner who by a universal remote control that kept secured, is recognized by the smart device. When user deploys device, secure transient association is used and imprinting can be used to establish shared secret. However, control gain of users' devices by a hacker, eavesdropping of communication channels, modification of sensitive commerce transactions, DoS, transaction of services or goods in other identities, are among numerous threats that are difficult to track and secure in ubiquitous networks [4].

Thus, ubiquitous network infrastructure will require the provision of certain degree of security between participating user devices. And therefore, there are interesting and challenging problems in providing consistency in the management of security and in specifying authorization policies for pervasive computing environments [4].

Authentication is one of the most important characteristics of ubiquitous computing security. Authentication provides confirmation of user access rights and privileges to the information to be retrieved. During the authentication process, a user is identified and then verified not to be an imposer. The authentication process is the assurance process that a party to some computerized transaction is not an impostor [4, 11].

3.2 Challenges

The ubiquitous network is nowadays almost at hand. The combination of technologies and services offered by the cable, wired and mobile telephony, wireless and satellite could quickly lead to reliable and complete network coverage. For that fact, hopes on pervasive computing environments do not cease to increase. Nevertheless, challenges remain very important and the challenge string touches all stages of service life cycle. Traditional security requirements include authentication, authorization and confidentiality. The security must be defined in terms of services themselves, the way they are dynamically added and removed, the way they are discovered and delivered, and their availability. In the other side, a service consumer expects from the system its peculiarity protection and a maximum of available service with a free access. Between service and consumption, the problem is likely to be

complex and interest conflicts may be generated [4].

Since Cyber-criminals and computer villains are already considering new, ingenious attacks that are not possible in traditional computing environments [6], current pervasive computing research will not continue without considering privacy in the system [12, 13] and securing pervasive computing environments presents challenges at many levels [14, 15, 7].

3.2.1 Privacy

Unfortunately, the privacy of users could be severely threatened. The entire system now becomes a distributed surveillance system that can capture too much information about users. In some environments, like homes and clinics, there is usually an abundance of sensitive and personal information that must be secured. Moreover, there are certain situations when people do not want to be tracked.

3.2.2 Access control mechanism

The access control mechanisms should allow groups of users and devices to use the active space in a manner that facilitates collaboration, while enforcing the appropriate access control policies and preventing unauthorized use. While designing access control mechanisms, it has to be taken into account that users in the active space cannot easily be prevented from seeing and hearing things happening in it.

4. Security Infrastructure Model for Ubiquitous Networks

4.1 Security System Model

The adoption of legacy network services as part of the infrastructure supporting new ubiquitous applications is a natural evolutionary approach to technology that allows new systems to build on top of existing knowledge and technologies. However, the ubiquitous computing paradigm does not have the same requirements as those related to traditional enterprise and personal computing. It's characterized by a larger scale and higher dependability requirements. To ensure the dependability of ubiquitous applications, network services upon which these applications are built must be robust and embed dependability paradigms in their design. When adopting legacy network services, it is essential to take in consideration the new requirements of the ubiquitous computing environment [16].

An open architecture platform must be required to communicate with heterogeneous networks. It's designed for supporting interconnection and compatibility with heterogeneous networks. This platform should be independent of hardware and software vendors, should connect with existing entire networks, and accept new networks in the future [17].

To perform authentication, granting privilege, a centralized authority with interfaces for several devices is integrated. In this context, we distinguish two representative protocols for reusing tickets within the ticket's valid time; the Neuman-Stubblebine authentication protocol [18] and the Kerberos authentication protocol [19] developed by MIT. [18] takes advantage of its ability to prevent replay attacks, within a ticket's valid time, by not requiring

synchronization of the time stamp with each party [17]. Kerberos [20] is a widely deployed authentication system. It has several characteristics that make it an ideal candidate for implementing authentication systems for ubiquitous applications. Two of the main characteristics consist of re-use of cached security credentials and the use of lightweight symmetric key cryptography [16]. Key management can be based on trust relationships [14]. Tickets and session keys are the fundamental atoms of Kerberos and the Key Distribution Center (KDC) is the centralized authority which issues to applicant a ticket [21] that proves its identity to others. This service is implemented in each network domain controller [9]. The Kerberos authentication protocol [19] consists of Authentication Server (AS) and Ticket Granting Server (TGS) [17].

AS it is shown in Figure 2, to communicate securely with Service Provider, a user is authenticated firstly by an AS and receives after a service issue ticket from TGS. Using this ticket, the user achieves service privileges from the Service Provider. Within the ticket's valid time and without communicating with the Kerberos system, the user keeps the privileges for the service [17].

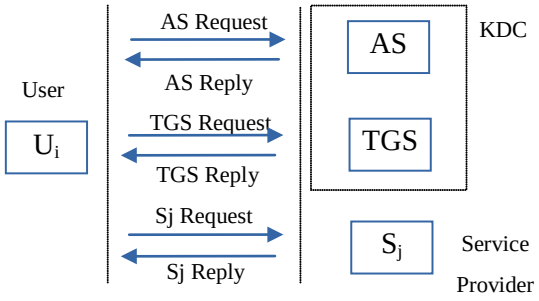

Figure 2. Kerberos protocol

The pervasive network is comprised of pervasive network users, smart devices, smart network services, and centralized authority which is supposed to be responsible for the security of every pervasive and do therefore a central role in the pervasive network [4].

The security infrastructure of a pervasive network essentially boils in the centralized authority. It consists mainly of an authentication entity, an authorization entity and a security policy. Through their close cooperation, these entities secure access to pervasive network components. Such infrastructure is installed in each domain of the ubiquitous network and all pervasive network packets must pass through it. Whenever a new pervasive network access is detected, it should be able to authenticate, authorize and enforce security policy [4, 22, 23]. The centralized authority corresponds to Key distribution Center (KDC) and the authentication and authorization authorities correspond to authentication (AS) and ticket granting (TGS) servers in Kerberos.

4.2 Security Policies

It is important in pervasive computing to have a flexible and convenient method for defining and managing security policies in a dynamic and flexible fashion. Policy management tools provide administrators the ability to specify, implement, and enforce rules

to exercise greater control over the behavior of entities in their systems. Currently, most network policies are implemented by systems administrators using tools based on scripting applications [24, 25] that iterate through lists of low-level interfaces and change values of entity-specific system variables. The policy management software maintains an exhaustive database of corresponding device and resource interfaces. With the proliferation of heterogeneous device-specific and vendor-specific interfaces, these tools may need to be updated frequently to accommodate new hardware or software, and the system typically becomes difficult to manage. As a result, general purpose low-level management tools are limited in their functionality, and are forced to implement only generic or coarse-grained policies [26]. Since most policy management tools deal with these low-level interfaces, administrators may not have a clear picture of the ramifications of their policy management actions. Dependencies among objects can lead to unexpected side effects and undesirable behavior [27]. Further, the disclosure of security policies may be a breach of security. For example, knowing whether the system is on the lookout for an intruder could actually be a secret. Thus, unauthorized personnel should not be able to know what the security policy might become under a certain circumstance [7].

In order to strengthen the authentication and authorization mechanisms, security policy rules are managed by the security policy entity (see figure 3). The administration of this entity can vary between an intelligent and automatic generation of rules depending on the needs and the behaviour authentication and authorization entities, and the intervention of authorized agent. These rules strengthen particularly the decision-making of the security entities and generally the pervasive network security.

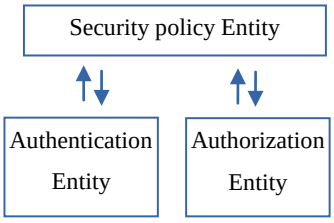

Figure 3. Security infrastructure for pervasive network

Based on these rules, the authentication and authorization entities authenticate and authorise users or devices accessing the pervasive network [23, 28, 29]. Thus, the pervasive computing environment will be able to provide services to only legitimate members and make each user of the pervasive network reliable and able to use safely the pervasive network services.

4.3 Proposed Protocol

The authentication process involves *principals* and a principal authentication and authorization authority. Principals represent users and services registered in the domain. A database of principals is maintained and secret keys are shared between authority and each one of principals. The protocol realizes user's access to offered services in three types of exchanges developed in three types of operations: the protocol exchanges include the *Authentication Server (AS) exchange*, the *Ticket Granting Service (TGS) exchange* and the

Server Provider (AP) exchange [16]. The process is shown in figure 4.

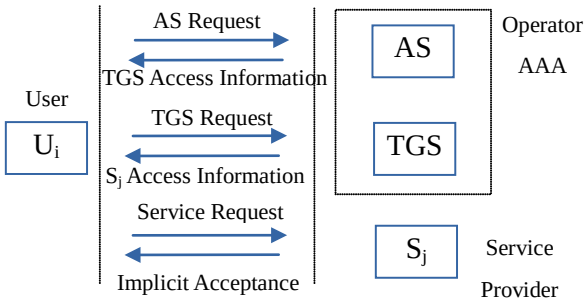

Figure 4. Chronology of authentication

As result, from these three exchanges, three types of operations [9] are formed. They include:

Authentication stage: By single sign-on techniques, user is authenticated by Authentication Server (AS). Then, once user gets a ticket with a limited lifetime from ticket granting server (TGS), he obtains a right to request access to authorized services.

Access control stage: By detainning a ticket with a limited lifetime, user can receive a service acces authorization from solicited services providers. The ticket contains an implicit authentication and recognition by providers of services that the user has access right.

Key negotiation stage: Two session keys are generated in the protocol; the first one is delivered by the Authentication Server (AS) to the user for communication between user and Ticket Granting Server (TGS), the second is delivered by the TGS to the user for communication between user and the service provider.

Table 1. Notations used in the proposed scheme.

Symbol	***Definition***
U_i	User i
S_j	Service j
ID_i	User's identity
TID_i	Transformed user's identity
PW_i	User's chosen password
AS	Authentication Server
TGS	Ticket Granting Server
h(.)	One-way hash function
T_A	Timestamp generated by A
L_A	Ticket lifetime
AL_{Ui}	Access level of U_i
$E_{PB(A)}\{M\}$	Encryption of message using public key of A
$E_K\{M\}$	Encrypted of message using symmetric key K
$E_{A-B}\{M\}$	Encryption of message using symetric key between A and B

The proposed authentication scenario is outlined in each step as follows (the notation

used is provided in table 1):

Step 1. $U_i \Rightarrow AS : <M_1 = E_{PB(AS)} (UID_i,PW_i)>$

User U_i logs into his mobile device and requests access to a particular service. For this purpose, User U_i transmits UID_i and PW_i to AS/AAA. The first message M_1 is encrypted using AS's public key.

Step 2. $AS \Rightarrow U_i : <Ticket_1,M_2>$

Receiving the message M_1, AS proceeds as follows:

1. decrypt the message M_1 using AS's private key,
2. Verify in its users' database that it knows U_i, and then authenticates client,
3. Based on U_i login, AS calculates a symmetric key $SK=f(UID_i,PW_i)$,
4. Generate a random R_1,
5. Compute a U_i′s transformed identity $TID_i=f(UID_i, R_1)$,
6. Generate a random R_2,
7. Compute a session key U_i-TGS: $S_{Key1}=h(SK, R_2)$,
8. Encrypt the message using SK as follows: $M_2=E_{SK} (AS_{ID},TID_i, TGS_{ID},R_2,h(S_{Key1},UID_i),T_{AS})$ where T_{AS} is AS timestamp,
9. Create a ticket and encrypts it using AS-TGS session key: $Ticket_1=E_{AS\text{-}TGS}(AS_{ID},TID_i,TGS_{ID},S_{Key1},T_{AS}, L_{TGS},AL_{Ui})$ where L_{TGS} is ticket lifetime and AL_{Ui} is U_i acces level,
10. Deliver $Ticket_1$ and M_2 to U_i.

The 4th and 5th operations are optional to preserve the identity of the user. Otherwise, $TID_i=UID_i$.

Step 3. $U_i \Rightarrow TGS : <Ticket_1,M_3>$

Receiving the message M_2, U_i proceeds as follows:

1. Calculate the symmetric key $SK=f(UID_i,PW_i)$,
2. Decrypt the message M_2 using SK,
3. Calculate $S'_{Key1}=h(SK, R_2)$,
4. Verify if $h(S'_{Key1},UID_i)$?= $h(S_{Key1},UID_i)$,
5. Generate a random R_3,
6. Compute authenticator as follows: $M_3=E_{S'Key1}(TID_i,TGS_{ID},S_j,R_3,T_{Ui})$ where S_j is the requested service and T_{Ui} is U_i timestamp,
7. Send $Ticket_1$ and M_3 to TGS.

Step 4. $TGS \Rightarrow U_i : <Ticket_2,M_4>$

Receiving the message M_3, TGS proceeds as follows:

1. Decrypt $Ticket_1$ using AS-TGS session key and obtains session key S_{Key1},
2. Verify T_{AS} and L_{TGS},
3. Decrypt authenticator M_3 using S_{Key1},
4. Verify T_{Ui} and $TID_i(M_3)$=? $TID_i(Ticket_1)$,
5. Check if U_i's access level (AL_i) permits acces to S_j,
6. Generate a random R_4,
7. Compute a session key U_i-S_j: $S_{Key2}=h(R_3, R_4)$,
8. Encrypt the message using R_3 as follows: $M_4=E_{R3}(TGS_{ID},TID_i, SID_j,R_4,h(S_{Key2},TID_i),T_{TGS})$ where T_{TGS} is TGS timestamp,
9. Create a ticket and encrypts it using TGS-S_j session key: $Ticket_2=E_{TGS\text{-}Sj}(TGS_{ID}, TID_i, SID_j, S_{Key2}, T_{TGS}, L_{SIDj})$ where T_{TGS} is TGS timestamp and L_{SIDj} is ticket lifetime,
10. Send $Ticket_2$ and M_4 to U_i.

Step 5. $U_i \Rightarrow S_j$: <$Ticket_2$,M_5>

Receiving the message M_4, U_i proceeds as follows:

1. Decrypt the message M_4 using R_3,
2. Compute $S'_{Key2}=h(R_3, R_4)$,
3. Verify if $h(S'_{Key2},TID_i)$?= $h(S_{Key2},TID_i)$,
4. Generate a random R_5,
5. Compute authenticator as follows: $M_5=E_{S'Key2}(TID_i,SID_j,R_5,T'_{Ui})$ where T'_{Ui} is U_i timestamp,
6. Send $Ticket_2$ and M_5 to S_j.

Step 6. $S_j \Rightarrow U_i$: <M_6>

Receiving the message M_5, S_j proceeds as follows:

1. Decrypt $Ticket_2$ using TGS-S_j session key, and obtains session key S_{Key2},
2. Verify T_{TGS} and L_{SIDj},
3. Decrypt authenticator M_5 using S_{Key2},
4. Verify T'_{Ui} and $TID_i(M_5)$=? $TID_i(Ticket_2)$,
5. Encrypt the message using S_{Key2} as follows: $M_6=E_{SKey2}(R_5)$,
6. Send M_6 to U_i.

Step 7.

Receiving the message M_6, U_i proceeds as follows:

1. Decrypt the message M_6 using S_{Key2},
2. Verify content?

To note that U_i can request a variety of services. He must specify in M_3 the requested services. TGS sends to him SID_j and corresponding ticket for every S_j requested.

Assuming that a user Ui, having access in services in local domain, wishes to access a service deployed in a remote domain. In order to permit this user's access to services in other network domains, an inter-domains authentication is established. The local KDC and the remote KDC share an inter-domain key which is used to secure inter-domain exchange. For this purpose, only operations of step 4 in the next proposed protocol are modified as follows:

Step 4. TGS $\Rightarrow$ U_i : <$Ticket_2$,M_4>

Receiving the message M_3, TGS proceeds as follows:

Step 4.1 TGS $\Rightarrow$ TGSR : < M'_3>

1. Decrypt $Ticket_1$ using AS-TGS session key and obtains session key S_{Key1},
2. Verify T_{AS} and L_{TGS},
3. Decrypt authenticator M_3 using S_{Key1},
4. Verify T_{Ui} and $TID_i(M_3)$=? $TID_i(Ticket_1)$,
5. Check if U_i's access level (AL_i) permits acces to S_j,
6. Encrypt the message using TGS-TGSR session key as follows: $M'_3=E_{TGS\text{-}TGSR}$ (TGS_{ID},TID_i,S_j,R_3,T_{TGS}) where T_{TGS} is TGS timestamp,
7. Send M'_3 to TGSR.

Step 4.2 TGSR $\Rightarrow$ TGS : < M'_4>

Receiving the message M'_3, TGSR proceeds as follows:

1. Decrypt M'_3 using TGS-TGSR session key
2. Verify T_{TGS},

3. Generate a random R_4,
4. Compute a session key U_i-S_j: $S_{Key2}=h(R_3, R_4)$,
5. Encrypt the message M_4 using R_3 as follows: $M_4=E_{R3}(TGSR_{ID},TID_i,SID_j,R_4,h(S_{Key2},TID_i),T_{TGSR})$ where T_{TGSR} is TGSR timestamp,
6. Create a ticket and encrypts it using TGSR-S_j session key: $Ticket_2=E_{TGSR\text{-}Sj}(TGSR_{ID}, TID_i, SID_j, S_{Key2}, T_{TGSR}, L_{SIDj})$ where T_{TGSR} is TGSR timestamp and L_{SIDj} is ticket lifetime,
7. Encrypt the message M'_4 using TGSR-TGS session key as follows: $M'_4=E_{TGSR\text{-}TGS}(Ticket_2, M_4)$ and send it to TGS.

Step 4.3 TGS $\Rightarrow$ U_i : <$Ticket_2$,M_4>

Receiving the message M'_4, TGS proceeds as follows:

1. Decrypt M'_4 using TGSR-TGS session key
2. Send $Ticket_2$ and M_4 to U_i.

The previous protocol rests on the following message flow shown in figure 5. User sends an access request to remote service to TGS in his domain. TGS takes care for representing him to TGSR of remote service domain and takes over by sending the user's request to TGSR and returning TGSR response to user. This latter can in final phase get closer to remote server. By no means in this approach, has user to entreat TGSR to access service in its domain.

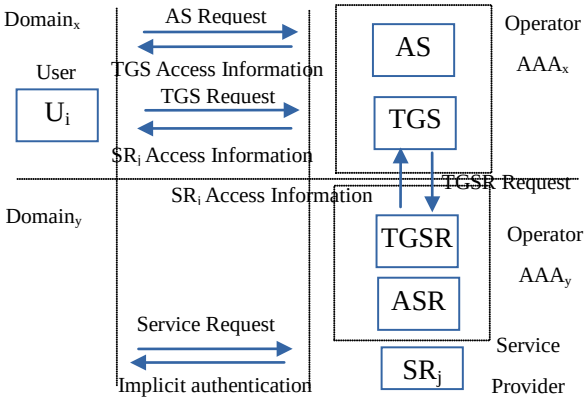

Figure 5. Remote authentication with direct care

In the second approach, TGS of user's domain transmits to him a ticket granting to TGSR of remote service domain. Only by getting closer to TGSR that the user will get an access right ticket to the remote service requested.

The operations of the first 4 steps are unchanged. Only the 9th operation of the 4th step will be amended as follow:

9. Create a ticket and encrypt it using TGS-TGSR session key: $Ticket_2=E_{TGS\text{-}TGSR}(TGS_{ID},TID_i,S_j,S_{Key2},T_{TGS},L_{TGSR},AL_{Ui})$ where T_{TGS} is TGS timestamp and L_{TGSR} is ticket lifetime,

The remainder of the protocol is summarized as followed:

Step 5. $U_i \Rightarrow$ TGSR : <$Ticket_2$,M''_4,TGS_{ID}>

Receiving the message M_4, U_i proceeds as follows:

1. Decrypt the message M_4 using R_3,
2. Compute $S'_{Key2}=h(R_3, R_4)$,

3. Check if $h(S'_{Key2},TID_i)$?= $h(S_{Key2},TID_i)$,
4. Generate a random R_5,
5. Compute authenticator as follows: $M''_4=E_{S'Key2}(TID_i,S_j,R_5,T''_{Ui})$ where T''_{Ui} is U_i timestamp,
6. Send $Ticket_2$, M''_4 and TGS_{ID} to TGSR.

Step 6. TGSR $\Rightarrow$ U_i : <$Ticket_3$,M''_5 >

Receiving the message M''_4, TGSR proceeds as follows:

1. Decrypt $Ticket_2$ using TGSR-TGS session key and obtains session key S_{Key2},
2. Verify T_{TGS} and L_{TGSR},
3. Decrypt authenticator M''_4 using S_{Key2},
4. Verify T''_{Ui} and $TID_i(M_5)$=? $TID_i(Ticket_2)$,
5. Check if U_i's access level (AL_i) permits acces to S_j,
6. Generate a random R_6,
7. Compute a session key U_i-S_j: $S_{Key3}=h(R_5, R_6)$,
8. Encrypt the message using R_5 as follows: $M''_5=E_{R5}(TGSR_{ID},TID_i,SID_j,R_6,h(S_{Key3},TID_i),T_{TGSR})$ where T_{TGSR} is TGSR timestamp,
9. Create a ticket and encrypts it using TGSR-S_j session key: $Ticket_3=E_{TGSR\text{-}Sj}(TGSR_{ID},TID_i,SID_j,S_{Key3},T_{TGSR},L_{SIDj})$ where T_{TGSR} is TGSR timestamp and L_{SIDj} is ticket lifetime,

Send $Ticket_3$ and M''_5 to U_i.

Step 7. $U_i \Rightarrow S_j$: <$Ticket_3$,M_5>

Receiving the message M_5, U_i proceeds as follows:

1. Decrypt the message M''_5 using R_5,
2. Compute $S'_{Key3}=h(R_5, R_6)$,
3. Verify if $h(S'_{Key3},TID_i)$?= $h(S_{Key3},TID_i)$,
4. Generate a random R_7,
5. Compute authenticator as follows: $M_5=E_{S'Key3}(TID_i,SID_j,R_7,T'''_{Ui})$ where T'''_{Ui} is U_i timestamp,
6. Send $Ticket_3$ and M_5 to S_j.

Step 8. $S_j \Rightarrow U_i$: <M_6>

Receiving the message M_5, S_j proceeds as follows:

1. Decrypt $Ticket_3$ using TGSR-S_j session key, and obtains session key S_{Key3},
2. Verify T_{TGSR} and L_{SIDj},
3. Decrypt authenticator M_5 using S_{Key3},
4. Verify T'''_{Ui} and $TID_i(M_7)$=? $TID_i(Ticket_3)$,
5. Encrypt the message using S_{Key3} as follows: $M_6=E_{SKey3}(R_5)$,
6. Send M_8 to U_i.

Step 9.

Receiving the message M_6, U_i proceeds as follows:

1. Decrypt the message M_6 using S_{Key3},
2. Verify content?

The previous algorithm is described using the message flow shown in figure 6.

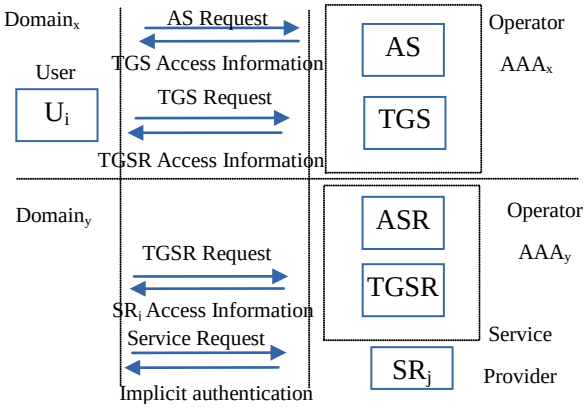

Figure 6. Remote authentication with indirect care

If the user loses connexion to his basic authority of authentication and authorization by migration to the surrounding domains, he will have to register and authenticate to the new authority of which he wants to acquire services. He can dispense with the registration operation by sending a service request to AS of the new domain. This request initiates a dialogue between old and new authorities as follows (it is described using the message flow shown in figure 7):

Step 1. $U_i \Rightarrow AS_{new} : <M_1 = E_{PB(ASnew)} (TID_i, PW_i, AS_{ID}>$

User U_i logs into his mobile device and requests access to a particular service in the new domain. For this purpose, User U_i transmits his identity (TID_i, PW_i) and the identity of AS that recognizes him to the new AS/AAA. The first message M_1 is encrypted using AS_{new}'s public key.

Step 2. $AS_{new} \Rightarrow AS : <M_2 = E_{ASnew-AS} (TID_i, PW_i)>$

Receiving the message M_1, AS_{new} proceeds as follows:

1. decrypt the message M_1 using AS_{new}'s private key,
2. Encrypt using ASnew-AS session key the identity of user to be identified and transmits it to AS,

Step 3. $AS \Rightarrow AS_{new}$

Receiving the message M_2, AS proceeds as follows:

1. decrypt the message M_2 using ASnew-AS session key,
2. Verify in its users' database that it knows U_i, and then authenticates the message of AS_{new}. AS sends the information requested by AS_{new} if necessary.

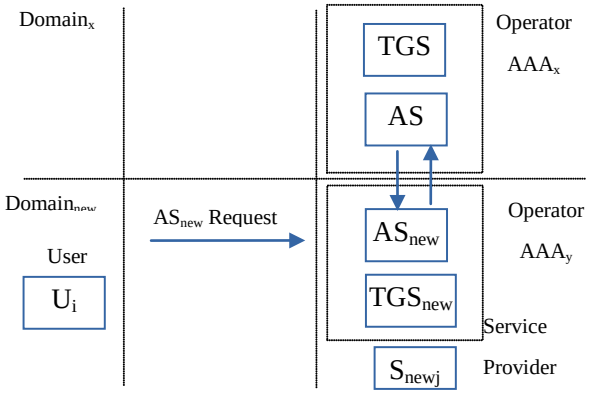

Figure 7. Migration to neighbouring domain

Recognizing U_i, AS_{new} will start the authentication process for U_i's acces to services of the new domain.

4.4 Proposed Protocol Evaluation

Under the assumption that public key or symmetric key infrastructure is used according to TGS's storage and computation capabilities, the symmetric key is shared between AS and TGS. If AS is an additionnel component to bring to the system in order to form a security infrastructure, TGS may be an already existing component; this later may be a Home Gateway in ubiquitous home network [30, 31] or a base station in Wireless Sensor Network [32]. In these cases, as authentication mechanism of AS in the ubiquitous environment, TGS (HGW or BS) knows that AS is legitimate using the PKI-based public key algorithm or symmetric algorithm. Generally, Communications in ubiquitous networks can be secured at all levels by use of symmetric algorithm which is computationally fast: Both provider and consumer of services trust AS and TGS. A symmetric key is also shared between TGS and service provider SP. The KDC composed of AS and TGS manages the environment, authenticates users, grants privileges, and controls accounting.

To note that encryption by public keys prevent user's personal information (identity and password) from guessing techniques. During authentication and authorization process, the user operates in an anonymous manner: His identity is hidden and Ticket Granting Server (TGS) knows only transformed identity while service provider may know nothing.

The transmission of user's access lever by AS permits to TGS to control the access request to the service. The single sign-on mechanism and freshness of messages by introduction of time stamp prevent a replay attack. The use of tickets and session keys permits to attribute an identity to only its owner and consequently prevents passive and active attackers. It fully satisfies the security requirements of pervasive computing environment.

In order to achieve the authentication operations, only 6 messages have been designed to permit a user to access to services indoor the same domain. In other hand, 8 messages have been designed in each scenario of user access to services belonging to neighbouring domains. Whatever the case, the messages of authentication between user and authority of authentication or between user and service provider are almost the same (M_1, M_2, M_3, M_4, M_5, M_6). The messages (M'_3, M'_4) are added for dialogue between TGS and TGSR in the case of TGS support for access request to remote service. The messages (M''_4, M''_5) are added in the case that user fellows a personal approach to access to remote service after receiving from his authority a downstream which helps him near remote authority. We recall that whatever the access type, it's always a single sign-on mechanism.

In the messages that we have designed for the protocol described above, we have used 8 Bytes to identify the user, authentication authority, authorization authority and service provider. It follows at the same for the other parameters. We recall that the messages M_i for $i \in \{1,3,5\}$ are respectively requests of authentication, authorization and access to services sent by the user U_i to respectively the autorities of autentication, authorization and service provider. Therefore, the messages M_i for $i \in \{2,4,6\}$ are the returned answers. The messages

M_i and M_{i+1} for $i \in \{2,4\}$ carry from the previous requested entity to the user a privilege title which allows his recognition by the next requested entity. These messages are a very satisfactory size. Figure 8 show the size of the designed messages. Messages M_1 and M_6 are with so low number of bytes because they simply match to the first identification request and the implicit final authentication response.

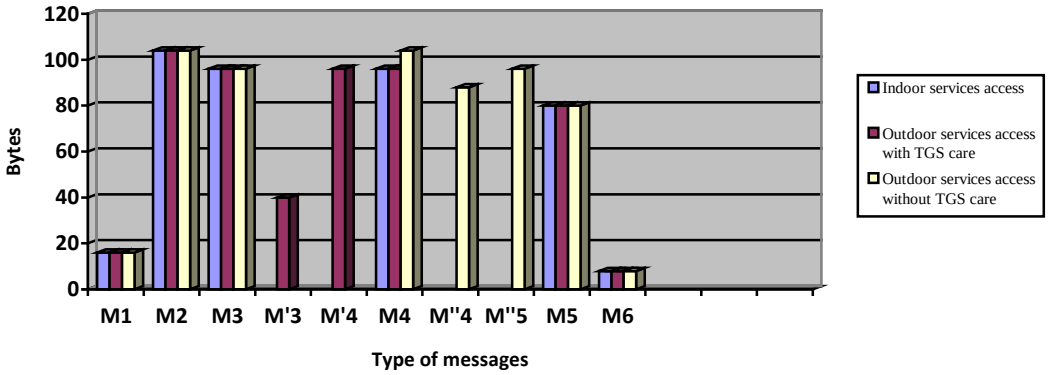

Figure 8. Bandwidth cost of messages

The messages are intended authentication between user and authentication authority of user's domain (AS, TGS) or surrounding domain (ASR, TGSR) and between user and provider service in user's domain (SP) or surrounding domain (SPR). Therefore, messages are composed of authentication and messaging information and of privilege title embedded in the messages in order to recognize the user by the future protagonist parties in the authentication chain by logical and chronological order. The amount of bytes of each message taking into account its information is shown in figure 9.

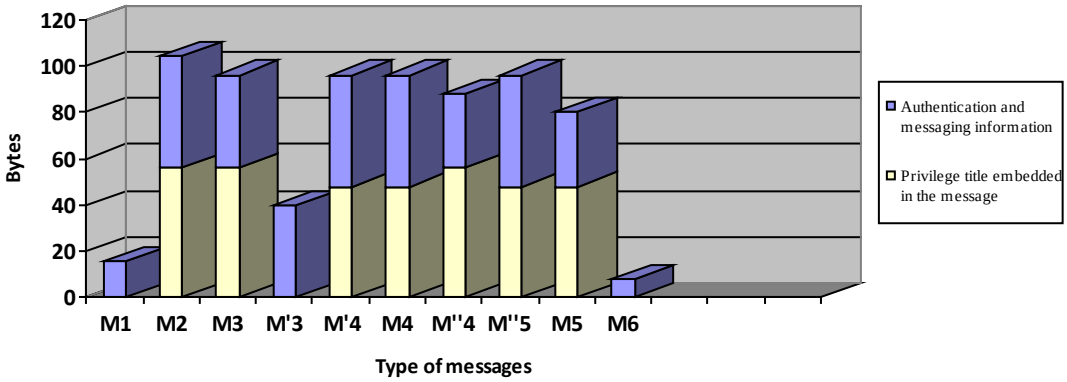

Figure 9. Messages constitution

4. Conclusion

In an environment where by default each object will be connected and accessible, it has become essential to implement infrastructures which secure the network and offer a pleasant ubiquitous setting. Over a wireless and/or wired network infrastructure, a pervasive computing

environment PCE consists of three type of entities: mobile users, services and back end authentication servers. To make the system architecture more scalable and flexible, a broker can be introduced between the user and service. Both users and services can interact with brokers to subscribe and distribute services. An access control scheme aims to secure the interactions among component entities of PCE. The security system is based on an infrastructure consisting of authority by domain. By modular composition, each authority is composed of entity of user authentication and entity of authorization for authenticated users to acces to the services provided by the pervasive network [4, 8].

A legitimate user must pass through 3 phases of recognition: authentication, authorization and service access. Authentication entity verifies the identity of the device and particularly the user like registration authority in PKI. An authenticated device receives a codified message that only authorization entity can decode in the access authorisation request to service submitted by the device. The authorization entity trusts the information in the request because the authentication entity already verified it. But, it restricts the access right to service. An authorized device received a second codified message that only smart server can decode in the final mutual authentication between smart service and smart device. This authentication and authorization process on 3-step will protect the network services, the users privacy and unmasks any adversary attempting fraudulent access by replication or alteration of messages addressed previously to authenticated users [4].

In the upcoming ubiquitous era, the user will want to easily and securely acquire the environmental information in his local area, but at the same time, the ubiquitous environment will want to provide its data to only legitimate users. For this purpose, mutual authentication between the user and the environment must be provided. The proposition in this paper aims to offer services to legitimate user in heterogeneous environment. It safeguards the privacy of the user and mutually authenticates user and service with the help of a third trusted party [32]. The scheme can be deployed in different domain of ubiquitous environment. It can be applied in Wireless Sensor Network [32], Home Network, smart Grid, Body Area Network or Ubiquitous Health Monitoring.

References

[1] "Ambient networks", https://www.fokus.fraunhofer.de/c483ab9db270ed57.

[2] "Wireless world initiative", http://www.wireless-world-initiative.org/.

[3] Haladjian R., "De l'inéluctabilité du réseau pervasif", FING, 2004.

[4] Djellali B., Lorenz P., Belarbi K., Chouarfia A., "Security model for pervasive multimedia environment", Journal of Multimedia Information Systems, Vol. 1, Issue 1, pp. 23-43, September 2014.

[5] Satyanarayanan M., "Pervasive computing: Vision and challenges", IEEE Personal Communications, Vol. 8, Issue 4, pp. 10-17, August 2001.

[6] Stajano F., "Ubiquitous Computing", in Security for Ubiquitous Computing, John Wiley

and Sons, Ltd, Chichester, UK, February 2002, http://dx.doi.org/10.1002/0470848693.ch2.

[7] Campbell R., Al-Muhtadi J., Naldurg P, Sampemane1 G, Mickunas M.D., "Towards Security and Privacy for Pervasive Computing", Proceeding of International Symposium on Software Security, Keio University, Tokyo, Japan, November 2002, appeared in Software Security –Theories and Systems, Springer LNCS, Volume 2609, pp. 1-15, November 2003.

[8] Djellali B., Belarbi K., Chouarfia A., Lorenz P., "User authentication scheme preserving anonymity for ubiquitous devices", Security and Communication Networks (2015) © 2015 John Wiley & Sons, Ltd., 1st published online 26 March 2015, http://dx.doi.org/10.1002/sec.1238.

[9] Yeun C.Y., Lua E.K., Crowcroft J., "Security of emerging ubiquitous networks", Proceeding of the IEEE 62nd Semiannual vehicular technology conference, Dallas, Texas, USA, Vol 2, pp. 1242-1248, September 2005.

[10] Weiser M., "The Computer for the Twenty-First Century", Scientific American, pp. 94-102, September 1991.

[11] Naqvi S.S.H., "Architecture de Sécurité pour les Grands Systèmes, Ouverts, Répartis et Hétérogènes", Thesis, Higher National School of Telecommunication, Paris, France, December 2005.

[12] Langheinrich M., "Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems", UbiComp 2001, Lecture Notes in Computer Sciences LNCS 2201, pp. 273-291, September 2001.

[13] Langheinrich M., "A Privacy Awareness System for Ubiquitous Computing Environments", 4th International Conference on Ubiquitous Computing, UbiComp 2002, Goteborg, Sweden, Lecture Notes in Computer Sciences LNCS 2498, pp. 237-245, September-October 2002.

[14] Kagal L., Finin T., Joshi A., "Trust-Based Security in Pervasive Computing Environments", IEEE Computer, Vol. 34, Issue 12, pp. 154–157, December 2001, http://dx.doi.org/10.1109/2.970591.

[15] Kagal L., Undercoffer J., Perich F., Joshi A., Finin T., "Vigil: Enforcing Security in Ubiquitous Environments", in *Grace Hopper Celebration of Women in Computing 2002*, 2002.

[16] Zrelli S., Okabe N., Shinoda Y., "XKDCP: An Inter-KDC Protocol for Dependable Kerberos Cross-Realm Operations", Journal of Networks, Vol. 8, Issue 2, pp. 290-296, February 2013, http://dx.doi.org/10.4304/jnw.8.2.290-296.

[17] Jeong J., Chung M.Y., Choo H., "Secure User Authentication Mechanism in Digital Home Network Environment", Proceeding of the 2006 international conference on Embedded and Ubiquitous Computing (EUC'06), Lecture Notes in Computer Sciences LNCS 4096, pp. 345–354, 2006, http://dx.doi.org/10.1007/11802167_36.

[18] Neuman B.C., Stubblebie S.G., "A Note on the Use of Timestamps as Nonces", ACM SIGOPS Operating Systems review, Vol. 27, Issue 2, pp. 10-14, April 1993.

[19] Neuman B.C., Is'o T., "Kerberos: An Authentication Service for computer Network", IEEE Communications Magazine, Vol. 32, Issue 9, pp. 33-38, September 1994.

[20] Neuman C., Yu T., Hartman S., Raeburn K., "The Kerberos Network Authentication Service (V5)", RFC 4120, July 2005.

[21] Patel B., Crowcroft J., "Ticket based service access for the mobile user", Proceeding of the 3rd annual ACM/IEEE international conference on Mobile computing and networking (MobiCom"97), Budapest, Hungary, pp. 223–233, 1997, http://dx.doi.org/10.1145/262116.262150.

[22] Jeong Y., Yoon K., Ryou J., "A Trusted Key Management Scheme for Digital Right Management", ETRI Journal, Vol.27, Issue 1, pp. 114-117, February 2005, http://dx.doi.org/10.4218/etrij.05.0204.0043.

[23] Lee D.G., Han J.W., Park D.S., Lee I.Y., "Intelligent Pervasive Network Authentication: S/Key Based Device Authentication", 6th IEEE Consumer Communications and Networking Conference (CCNC 2009), Las Vegas, USA, 2009, http://dx.doi.org/10.1109/CCNC.2009.4784994.

[24] Boyle J., Cohen R., Durham D., Herzog S., Rajan R., Sastry A., "The COPS (Common Open Policy service) Protocol", Request For Comment 2748, Network Working Group, January 2000.

[25] Case J., Mundy R., Partain D., Stewart B., "Introduction to version 3 of the Internet – Standard Network Management Framework", Request For Comments 2570,Network Working Group, April 1999.

[26] Stevens M., Weiss W., Mahon H., Moore B., Strassner J., Waters G., Westerinen A., Wheeler J., "Policy Framework." The Internet Engineering Task Force IETF draft, Policy Framework Working Group, September 1999.

[27] Loscocco P., Smalley S., "Integrating Flexible Support for Security Policies into the Linux Operating System", Proceedings of the FREENIX Track of the 2001 USENIX Annual Technical Conference, pp. 29-42, 2001.

[28] Gehrmann C., Nyberg K., Mitchell C.J., "The personal CA-PKI for a personal area network", IST Mobile and Wireless Telecommunications, Summit 2002, pp. 31-5, 2002.

[29] Hwang J.B., Lee H.K., Han J.W., "Efficient and User Friendly Inter-domain Device Authentication/Access control in Home Networks", Proceeding of the 2006 IFIP International Conference on Embedded and Ubiquitous Computing (EUC 2006), Seoul, Korea, Lecture Notes in Computer Sciences LNCS 4096, pp. 131-140, August 2006.

[30] Fadlullah Z.M., Fouda M.M., Kato N., Takeuchi A., Iwasaki N., Nozaki Y., "Toward Intelligent Machine-to-Machine Communications in Smart Grid", IEEE Communications Magazine, Vol. 49, Issuer 4, pp. 60-65, April 2011, http://dx.doi.org/10.1109/MCOM.2011.5741147.

[31] Fouda M.M., Fadlullah Z.M., Kato N., Lu R., Shen X.S., "A Lightweight Message Authentication Scheme for Smart Grid Communications", IEEE Transactions on Smart Grid, Vol. 2, Issue 4, pp. 675-685, December 2011, http://dx.doi.org/10.1109/TSG.2011.2160661.

[32] Djellali B., Chouarfia A., Belarbi K., Lorenz P., " Authenticated key exchange protocol for wireless sensor networks", To appear in the International Conference on Telecommunications and ICT (ICTTELECOM 2015), Oran, Algeria, 16-17 May 2015.

Copyright Disclaimer