A Novel Authentication Mechanism Protecting Users' Privacy in Pervasive Systems

Mohammed Djedid and Abdallah Chouarfia
Faculty of Science, University of Sciences and Technology of Oran-Mohammed Boudiaf-, Algeria

Abstract: *Transparency of the system and its integration into the natural environment of the user are some of the important features of pervasive computing. But these characteristics that are considered as the strongest points of pervasive systems are also their weak points in terms of the user's privacy. The privacy in pervasive systems involves more than the confidentiality of communications and concealing the identity of virtual users. The physical presence and behavior of the user in the pervasive space cannot be completely hidden and can reveal the secret of his/her identity and affect his/her privacy. This paper shows that the application of state-of-the-art techniques for protecting the user's privacy still insufficient. A new solution named shadow protocol is proposed, which allows the users to authenticate and interact with the surrounding devices within an ubiquitous computing environment while preserving their privacy.*

Keywords: *Pervasive systems, identification/authentication, privacy.*

Received December 14, 2014; accepted May 2, 2015

1. Introduction

Protecting personal data and acting anonymously have always attracted the interest of people through time, and technological advances have increased their interest. Indeed, the growing number of personal data and services available in networks such as internet are often subject to attacks, to use such information and services for malicious purposes. However, they also have generated the need for sophisticated and robust security mechanisms to protect them and many have been successful in conventional networks [1] ubiquitous or pervasive computing promotes the proliferation of embedded devices, smart gadgets, sensors and actuators. Transparency of the system and its integration into the natural environment of the user are some of the important features of pervasive computing. This vision of ubiquitous computing was introduced by Weiser [26] who found that "The most profound technologies are those that disappear" he wrote: "They weave themselves into the fabric of everyday life until they are indistinguishable from it". But these characteristics that are strong points of pervasive systems, are also its weak points in terms of the users' privacy. Indeed, the transparency of pervasive systems and their integration into the natural environment make the task of data protection more complicated than for traditional systems where the user is aware of interactions with devices: Pervasive systems are generally integrated and invisible, it is difficult for users to know when, where, and how the surrounding devices collects data, and the manner in which they are treated, and expose heavily their privacy. Since, the advent of ubiquitous computing and pervasive computing, the management of privacy was the subject of active discussion [2, 4, 23]. On one hand, a pervasive computing environment needs to collect a large amount of information about its users to be able to adapt and respond better to their expectations, and without seeking them continually. On the other hand, the more information gathered by the system on its users is large, the higher risk of potential threat on the privacy of users. A wide range of definitions of privacy is found in the literature: "control over information disclosure" [8], "privilege of users to determine for themselves when, how, and to what extent information about them is communicated to others" [14] and the "ability of an individual to control the terms under which his/her personal information is acquired and used" [11]. So we locate the privacy at the point of the intersection between the protection of the user's privacy, the preservation of anonymity, and non-disclosure of the information and the specific characteristics of the user, when he/she interacts with the system. Although, encryption systems allow to have an accepted security level, by encrypting the information flowing through communication channels against a third party (for example, an intruder or a malicious system administrator), this solution nevertheless remains insufficient when we talk about open and frequently visited systems, as is the case in pervasive computing. Indeed, we must realize that our physical presence and behavior in the real world cannot be completely hidden and can reveal the secret of our identity and affect our privacy. In this paper, we present a new privacy protecting protocol that we named "The Shadow Protocol", which allows the users of an ubiquitous computing environment to authenticate and interact with the surrounding devices while preserving their privacy. Our confidentiality

protocol is designed to protect the anonymity and the unlinkability of actions made by a user vis-a-vis the intruders, curious system administrators and the system itself. Before this, we first present an overview of the main existing security systems protecting user's privacy and compare their effectiveness and their ability to meet the constraints imposed by pervasive systems. We will present a first step, the requirements in terms of privacy that the security systems must respect in the particular context of pervasive environments. Subsequently, different solutions proposed for the user's anonymity and privacy preserving will be studied. Section 4 is devoted to the presentation of our protocol followed by a synthesis about the presented solutions as well as our proposal to compare their ability to meet the privacy needs shown in section 2. Finally, we provide a formal definition of a particular type of attack based on external knowledge to evaluate and prove the effectiveness of our approach.

2. Privacy Concerns in Pervasive Computing

In this section we describe the six parameters presented by Langheinrich [18] to provide users with an acceptable level of privacy in a pervasive system, and other parameters as well, that we have introduced to further refine the evaluation of main security solutions dedicated to protect the privacy of users which we will present in section 3.

- Notice: Given the difficulty for a user to realize that a data collection is occurring, and to know how these data are processed in a pervasive system, a security system that respects the privacy of the user needs to be able to notify the user each time these personal data are about to be collected and used.
- Choice and Consent: The notion of privacy is not universal and depends on the personality of each individual: Each user should be free to choose the level of privacy that he/she wants. If the user gives less importance to his/her privacy as other parameters, his/her requirements in terms of confidentiality are less strict, in favor of an improved response time for example. However, if the user has a distrustful character, his/her choice will be to increase privacy at the expense of response time. In all cases, an effective security system should be able to meet the different sensitivities and preferences of any user, in terms of preservation of privacy.
- Anonymity and Pseudonymity: As we mentioned in the introduction, and in order to provide optimal protection of privacy to the user, a security system must be capable of preventing a link between personal data and actions of a user, and his/her identity (physical and virtual). For this purpose, concealing the real identity of the user behind a pseudonym is one of the most effective measures to ensure anonymity.
- Proximity and Locality: To face the new vulnerabilities introduced by pervasive computing, security and privacy preserving measures must be undertaken during the design process of a pervasive system, rather than being considered as additions to an existing system [7]. The proximity and locality can be seen as a lightweight confidentiality mechanism when conventional cryptographic methods cannot be supported by devices with low computing power and storage capacity. This measure will significantly reduce the number of potential attackers and even help to find the author of an attack since he/she has been physically present in the pervasive space.
- Adequate Security: In a classic computer system cryptography is considered the main mechanism to ensure confidentiality. However, cryptography is often not supported by the new low-capacity devices used in pervasive systems. Therefore, security models that require devices to have enough computing capability to support cryptography should be reconsidered.
- Access an Recourse: A security system guaranteeing privacy must be designed to ask the user for the necessary and sufficient information only, for his/her authentication and the fulfillment of the action that he/she wishes to undertake. Personal data of the user must be accessed in a simple manner through a standard interface, and the user should be informed for the use being made of his/her personal data.
- Decentralization: For better protection of privacy, a pervasive security system needs to be as decentralized as possible. Moreover, the identification and authentication data of a user should not follow the same path during the exchanges between him/her and his/her supervisory authority, to prevent eavesdropping network and thus prevent the attacker from retrieving the identity of the sender of messages after a certain number of iterations.
- Trust Management: The notion of trust is widely used in existing security systems. This property allows delegating the mechanisms of identification and authentication to several terminals. For example, a user can communicate his identification/authentication information to an intermediate entity between him/her and his/her authority, so that it cannot know the location of the sender, and the device that he/she uses thereby certifying the authenticity of his/her information.
- Transparency and Proactivity: In a ubiquitous environment, each entity must be able to authenticate itself and acquire rights in a simple

and transparent way. In addition, the resources used must seek the user as less as possible.

3. Related Works

This section will briefly present some of the main security systems available to protect the privacy of users.

In radius [22] a user issues an access-request that contains his/her authentication information, and sends to the server. The server processes the request locally if it recognizes the user; otherwise it acts as a radius proxy "or intermediate" by transmitting it to another server. When the request arrives at the radius server corresponding to the identification item, it validates or rejects the request. The main advantage of radius is the decentralization of the authority of authentication and the deployment of relations between servers, by propagating information from one server to another. But the forwarding of authentication data between servers increases the risk of identity disclosure of the user at several levels of radius proxy by an intruder or a curious administrator server which want to read authentication data that are not destined to it.

In WS-Security [19] seeks to encapsulate the security interactions with a set of headers of Simple Object Access Protocol (SOAP). The password is scrambled with a hash function SHA-1function that converts a large set in a smaller one called footprint. It is impossible to decipher it to return to the original set. The hashed password is the concatenation of nonce (arbitrary number), time of creation, and password by the hash function. When it is received, the client creates the same type of password with the same information at its disposal. If the two hashed passwords (the one that it received and the one that it calculated) match together, the password must be correct. This protection has the disadvantage of failing to protect the authentication data from replay attacks: We should consider including expiration time beyond which the hashed password must be regenerated for a better identity protection.

Randomized ID [25] is another cryptographic mechanism which prevents against unauthorized tracking. To do so a tag should send a different ID every time the tag reader requests it. But this solution is not appropriate for a large number of tags.

Most widely used encryption methods can also be used in order to protect sensitive personal data of the user [15].

Anonymizer [3] is a solution that aims to make the activity of a user on the internet intractable. This is made possible through a centralized proxy server acting as an intermediary and a guarantor of the user's privacy. The centralization of the anonymization server is the weak point of this approach. In the same way, an anonymity-preserving reputation framework called IncogniSense is proposed in [10], a system based on the allocation of temporary pseudonyms, or periodicals, based on a blind signature, and using a secure transfer mechanism reputation between pseudonyms.

In The Onion Routing (TOR) [21] each client must choose a random path, and then construct a "circuit" in which each node has the property to know its predecessor and its successor, without knowing more. The first node of the circuit will know the IP address of the user's machine. But from the second node, the negotiation will be through the partial circuit already built, so that the second node, for example, knows only the IP address of the first node (and the third when the third node will be added). This construction uses the concept of hybrid cryptography [13]. To route a packet to the server, the client must encrypt the package many times. (The first time, the client encrypts its TCP packet with the public key corresponding to the last node, the second time with the public key of the penultimate node etc.) The last time with the public key of the first node. When the client sends this packet to the circuit he built, the first node decrypts the packet with its private key and sends the packet to the second server; this one does the same and sends the packet to the next server etc., until the last server that decrypts the packet with its private key and obtains the original message. Generalization is a technique used in pervasive architectures such as Paws [17], SPARCLE [6], hierarchical identity-based encryption [12] and PerGym [20]. The idea is to preserve the secrecy about, the location (geographical or IP address of the device used) of the sender of a request for a service in relation to the provider of this service. This is made possible by making the context data less precise than their initial state. For example, the IP address of the device used by the issuer, could be replaced with less precise information, as the working group that owns the device for example.

In Sensitive Information Diluting Mechanism (PSIUM) [9] the client queries the service provider for more services than it really needed. The service provider responds to all requests, but only one of the queries that contain the desired service. The remaining queries (false queries) were created to hide the nature of the real service request, thereby reducing the usefulness of data collected by the server.

As PSIUM, K-anonymity [24] is a method of users' privacy protection based on the dissimulation of the sensitive data in a set of similar data. In presenting the solution K-anonymity, sweeney introduced the concept of Quasi-Identifiers (QI). QI is a set of attributes that can be potentially associated with external information to find the identity of an entity. For example, if the attacker intercepts a request containing a set of QI such as the rank of a professor, his/her age, and his/her gender, the number of potential issuers of this request is highly reduced. If in

addition the attacker is able to access some external information, such as the location of the professor in his/her office, then the link to the professor's identity will be made by the attacker as well as the access to other private information contained in the request and exchanged with the system. K-anonymity is designed in order to transform the attribute values of a query (ID, IQ) from each customer and make them indistinguishable among a set of K potential users.

Mist [2] is composed of a set of Mist routers that forms an overlay network. Mist routers circulate communication packets using a routing protocol named hop-by-hop handle-based protocol with a public key cryptography infrastructure to protect against the listening communications of a third party. Mist introduced the concept of "portals" that are installed in each pervasive space. Portals are special mist router capable of detecting the presence of a person and objects through sensors, but without the ability to identify or authenticate these entities. The identification and the authentication of a user is done at a higher level in the hierarchy, high enough that the Authentication Server (AS) is not able to infer the physical location of the issuer or the machine that it use to perform its task. Mist is a very attractive solution for protecting the user's privacy, since the link between the user's identity, his/her location, and his/her actions is broken. However, we believe that this solution applied alone is insufficient in the context of open systems and frequently visited as is the case in pervasive systems. Indeed, from the time that the virtual world merges with the real world, and our physical presence and behavior within the system can be observed by an intruder who is physically present in the pervasive space and combine this information with the data received by the AS, this external information could reveal the missing part to the attacker to unmask our identity, and thus know the devices used and possibly the actions taken from these devices. It is in order to compensate the shortcomings of these systems that we provide our protocol called shadow protocol. Its detailed description is made in the next section.

4. Our Proposal: The Shadow Protocol

Our approach takes advantage of many benefits offered by the systems seen previously. In addition, our shadow protocol provides a more complete solution and an additional level of protection of users' privacy by minimizing the risks of disclosure of the users' identity. Indeed, the protocols seen previously do not take into account the dimension of the observation of the user's behavior by an adversary, which may reveal the missing part to the adversary so that he/she can make the link between the logical and the physical identities of the user. In other words, in the previous works, an adversary should not have visual contact that would allow him to observe the users' behavior within the active space. Otherwise, the attacker could use this external knowledge, and combined it with the identification data intercepted at the AS and accurately establish the link between the physical and the virtual identity of the user, the devices that he/she has used, and possibly the actions he/she has taken from these devices. However, the conditions above cannot be guaranteed in most usage scenarios of pervasive systems. In fact, the accessibility and the openness of the system and the frequency of visits by mobile users are fundamental characteristics of pervasive systems. Imposing restrictions to them would mean ignoring the basic problem related to the user's privacy in the particular context of pervasive systems. It is in order to take into account all these constraints that we propose the shadow protocol. Our solution will cover the requirements of a wider range of usage contexts regarding pervasive systems in terms of security and user's privacy. Shadow protocol is a protocol for identification and authentication that protects the user's privacy in a pervasive environment. Shadow protocol aims to prevent a link being made by an adversary between the user's logical identity and its physical behavior in the pervasive space, as well as the devices and activities he/she undertakes. The main challenge is to give guarantees of protection of privacy against an adversary having auxiliary knowledge, such as the observation of individual behavior (by behavior we mean movements of the user in the pervasive space). To do so, we set up an entity of preserving the user's privacy that we called Privacy Preserving Server (PPS) between the user and the AS. The PPS is the main element of our architecture; each pervasive space should have its own one. It is through the PPS that all communications will transit between the users and the AS. In addition to detecting the presence of a person in his/her space, the PPS is also responsible for the generalization of contextual data (replacing the ip address of the sending machine by the workgroup to which it belongs, for example), management of pseudonyms (assign each user a set of pseudonyms valid for a limited period), and the anonymization of requests (concealing the authentication request of a given user in a set of fake authentication requests, to make the real and the fake requests indistinguishable). For a better understanding of the system, we consider a scenario Figure 1 from those to whom the adoption of the shadow protocol may be appropriated. Bob is a teacher at a university. In addition to the courses he teaches, Bob goes regularly to the pervasive workspaces of his department in order to perform tasks using the devices available in the workspace where he is. A workspace is composed of three main elements: The AS (which is responsible for registering and assigning the users rights, to access the resources), the PPS, and the resources (the devices available to users within the workspace). We assume that each

user owns his/her smart device containing his/her information. The User's Smart Device (USD) is a small wireless device equipped with Near Field Communication technology (NFC) technology [16]. The USD is also composed of a screen to receive notifications and a keyboard for taping information and exchange with nearby NFC terminals (Proximity/locality). Smart phones are mostly equipped with an NFC chip and can also play the role of a USD. These devices have the ability to support a cryptography structure, and each user shares a secret shared key with the PPS at the registration phase.

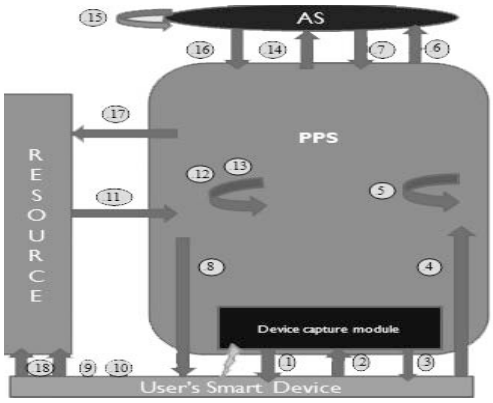

Figure 1. Usage scenario showing the interaction between a user and the pervasive space.

The PPS will be in charge of the registration of new users to the AS. Each user will be registered in the AS under different pseudonyms that the PPS has given him/her and which will be valid for one day. The PPS daily updates the pseudonyms of each old user and communicates the new pseudonyms to the AS. When Bob goes to a pervasive workspace, the Device Capture Module (DCM) of the PPS will detect his presence due to signals that Bob's smart device sends. The DCM will be able to detect the presence of the users in a pervasive space, but it will not be able to identify them. The users remain anonymous and unauthenticated for the moment. The DCM returns a request to Bob's smart device to inform him (Notice) of his detection and know whether he has visited this workspace in the past.

- *Step* 1: If the answer is "no".
- *Step* 2: The DCM then asks Bob to choose a Password (PWD).
- *Step* 3: If the user wants to interact with the pervasive space (Choice and consent). This password will be hashed by the user's private key (K_{usd}) stored on the USD (which is automatically generated and provided by the application during its installation on the USD). The USD also provides the PPS a shared secret key (K'_{usd}) which will be used by the PPS to encrypt Bob's data before they are transmitted to his USD. A synchronization phase precedes the packet transmission, with the addition of a TimeStamp (T_{usd}) to prevent against replay attacks (The most used method to synchronize two entities wishing to communicate in a client/server architecture is one in which the server broadcasts its clocks continuously with a Message Authentication Code (MAC) and every client that wants to communicate with that server must adjust its clock according to the clock issued by the server). When a client wants to send a message to the server, the client indicates the time T of the clock in his message (timestamp), which is also authenticated by an MAC. The server can decide, after verification of T on the basis of tolerance time if the request will be processed or not. All this information will be encrypted by the public key of the PPS (E_{kpps}) and sent to it. USD→PPS: the packet $P_{usd/pps}$: E_{kpps} (K'_{usd}, h_{Kusd}(PWD), T_{usd}).
- *Step* 4: No personal data will be required for Bob's registration or future interactions (Access and Recourse). After this registration phase, no module of PPS and the network in general will be able to precisely know the presence of Bob in the workspace during his next passages (Transparency/ Proactivity). Once the PPS receives $P_{usd/pps}$, it decrypts it and first checks the validity of TimeStamp T_{usd}. If T_{usd} is incorrect, the session ends, otherwise the process continues. The PPS then generates a set of pseudonyms that Bob may borrow during the day (ex: M1, M2, M3).
- *Step* 5: The PPS will hash every one of Bob's pseudonyms concatenated with the hashed password of Bob h_{Kusd}(PWD) using Bob's secret shared key K'_{usd}. As seen before, a synchronization phase between the AS and the PPS precedes the packet transmission, with the addition of a TimeStamp (T_{pps}) to prevent against replay attacks. The PPS sends separately and periodically the packets $P_{pps/as}$ that contain Bob's pseudonyms to the AS and also generates other registration packets containing fictitious pseudonyms, as a matter of fact the AS cannot distinguish the pseudonyms reserved for Bob and the pseudonyms of other potential users.
- *Step* 6: All this information will be encrypted by the public key Kas of the AS and sent to it. Example with the bob's pseudonym "M1": PPS→AS: the packet $P_{pps/as}$: $E_{Kas}(h_{K'usd}(M1||h_{Kusd}(PWD)),T_{pps})$. Once, the AS receives $P_{pps/as}$, it decrypts it and first checks the validity of TimeStamp T_{pps}. If T_{pps} is incorrect, the session ends, otherwise the process continues. Once the registration of Bob's new pseudonyms is done and confirmed by the AS.
- *Step* 7: The PPS encrypts the pseudonyms of Bob with his secret shared key K'_{usd}, and sends them to Bob's smart device.

- *Step* 8: PPS→USD: The packet $P_{pps/usd}$: $E_{K'usd}(M_1, M_2, M_3)$. In addition, and at the end of the day (00 p.m), the PPS proceeds to update the pseudonyms, by recalculating the new ones using a function that takes as entrees the old pseudonyms and the date of the current day. This is done in order to protect users against traceability. The PPS is also responsible for disseminating the updated pseudonyms to users when they come back. As soon as DCM detects the presence of an already registered user (The answer is "yes" at step 2, the PPS sends to his/her USD all encrypted pseudonyms in its possession. The user's smart device can decrypt only the pseudonyms signed by his/her secret shared key (adequate security). Other pseudonyms will be indecipherable and will be rejected by the user's smart device. Once Bob receives the list of pseudonyms that have been attributed to him, he can pick one whenever he decides to authenticate himself before performing a task. When Bob wants to use a resource within the workspace, he must be close enough to the resource that he wants to use.
- *Step* 9: Must be authenticated by the AS in a pull mode (In this mode, the resource sought by the user is directly requested (Note: If the resource is "not intelligent" like a printer or scanner, the USD will interact with the computer that is connected to this resource). The resource refers to AS to check if the user is allowed to use it. Thus, if the answer is positive, the resource provides access to the user based on the rights approved by the AS). Bob sends his identification/authentication data to the resource R by choosing one of the pseudonyms that the PPS has assigned to him (Pseudonymity/anonymity), and by tapping his password on the USD. The USD will hash this password with the Bob's private key (K_{usd}) and concatenate this hash with the chosen pseudonym. The result will be hashed again but this time with Bob's secret shared key. The generated hash will be encrypted with the AS public key and the PPS public key respectively. A synchronization phase precedes the packet transmission, with the addition of a TimeStamp (T'_{usd}).
- *Step* 10: USD→R→PPS: the packet: $P_{usd/r/pps}$: $E_{Kpps}(E_{Kas}(h_{K'usd}(M1 \parallel h_{Kusd}(PWD))),T'_{usd})$. Since the resource is unable to decrypt Bob's message because it is encrypted with the PPS's public key, the resource transmits the data received from Bob as it is to the PPS.
- *Step* 11: (Decentralization). The PPS then performs a K-Anonymity by generalizing the context data of Bob in order to make the data concerning him indistinguishable from a set of K potential users. For example by replacing the IP address of the resource that the user seeks, by the IP address of the PPS.
- *Step* 12: Therefore, the PPS will be the only entity in the entire architecture to know the exact source of the messages sent by the users, but without knowing the content of the packets and as a result without knowing their identities. The PPS will try to collect enough identification/authentication requests issued by other users in the same workspace as Bob at the same time that Bob wants to authenticate himself to send them in a group, in order to reduce the chances of an adversary with an eye on the data received by the AS and another on the movements and behaviors of the users within the workspace to connect the physical identity of a user to his/her logical identifier (Pseudonym). This condition is not easily feasible; the PPS will proceed mostly to the creation of fake authentication requests with fictitious pseudonyms registered previously on the AS in order to conceal the real request of Bob in a set of fake requests.
- *Step* 13: This phase will significantly reduce the chances for an adversary in possession of auxiliary information (the observation of users' movements in the workspace) to establish a precise link between the sender of the identification request (physical entity) and the received identifier by the AS (logical entity). At this point, the requests for identification/authentication (true and fake) including those of Bob will be sent simultaneously to the AS.
- *Step* 14: PPS→AS: The packet Ppps/as: $(E_{Kas}(h_{K'usd}(M1 \parallel h_{Kusd}(PWD))),T'_{pps})$. The AS checks if the received authentication/ identification data, including those of Bob are correct.
- *Step* 15: If it is the case, the AS will send a message to the PPS indicating that this user is authenticated.
- *Step* 16: Finally, and based on the positive response from the AS, the PPS will order the resource to perform the task that the user requests.
- *Step* 17: And the resource provides access to the user based on the authentication validated by the AS.
- *Step* 18: In summary Figure 2, Bob may identify/authenticate himself to the AS, which does not know his true identity (the identifier of Bob being substituted for one of the pseudonyms available for the day), and cannot distinguish between true and fake requests; therefore, it cannot provide useful information to the adversary.

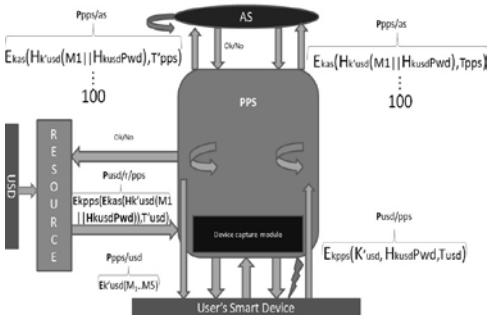

Figure 2. A detailed scheme.

Where K_{as}: Public key of the AS, K'_{usd} : Secret shared key of a user, K_{usd}: Private hash key of a user, K_{pps}: Public key of the PPS.

The AS cannot know the exact place where Bob is situated when he authenticates or the device he seeks, thanks to the K-anonymity and generalization phase performed by the PPS. Only the PPS will be able to know the exact location of the issuer of a request, but without knowing his/her identity so far. The PPS will be able to detect the entry of a user within his/her area but no entity within the pervasive workspace can recognize the effective identity of this user. If an attacker is physically present in the workspace, or if he/she has a different way to have a view on the workspace (cameras, sensors etc.,) enabling him/her to perform inference attacks, his/her chances of successfully disclosing the identity of the users and affecting their privacy will be highly reduced. This is made possible thanks to our joint application of K-anonymity, the generation of fictitious queries by the PPS, and the way we combine cryptographic methods to ensure that no sensitive and reusable information is transmitted, or stored on the various entities of the pervasive space. Our system will see the user like our perception of a person's shadow: We know when he/she is close, but we do not know who he/she really is. This is the reason why it is called the shadow protocol.

5. Synthesis

Before validating the results, we present in Table 1 the various solutions described in section 3 and our proposal, to compare their ability to meet the needs in terms of privacy, identified in section 2, and situates our solution in relation to previous works. The following mentions are used:

- √ √: The solution satisfies this need.
- √: The solution partially satisfies this need.
- ¬: The solution does not satisfy this need.
- Blank: No relationship between the solution and this need.

Table 1. Comparison between the security solutions regarding the needs in terms of privacy.

Privacy Needs	Notice	Choice/ Consent	Anonymity/ Pseudonym	Proximity/ Locality	Adequate Security	Access/Recourse	Decentral	Trust Manageme	Transparency/ Proacti
RADIUS		¬	¬	¬	¬	¬	√√	√√	√√
WSS		√√	¬	¬	¬	√√	¬	¬	√√
Randomized	¬	¬	√√	√√	√√	¬	¬	¬	√√
Anonymi		¬	√√	¬	√	√	¬	√√	√√
IncogniSense	¬	√	√√	√√	√√	¬	√√	√√	√√
Tor	√√	√√	¬		√√		√√	√√	√√
Generali	√√		√			√√	¬	¬	√√
PSIUM	√√	√	¬		¬	√	¬	¬	√√
K-Anon			√√		√√	√√	¬	¬	√√
Mist		√√	¬	√√	√√	√√	√√	√√	¬
Shadow	√√	√√	√√	√√	√√	√√	√√	√	√√

As we have stated previously, the shadow protocol's driving idea, is to ensure on one hand, the identification/authentication of people and allow them to use resources without any personal, confidential, or reusable data by a third party, can flow in the network or stored in the various entities of the pervasive space, on the other hand, to prevent a link to be made between the movements/ behavior of a user and data/messages received by the AS which may reveal to which user owns these received data. Indeed, if we take for example, the case where the administrator of the AS is the adversary or an external adversary enters the AS and retrieves the users' database. The only information the administrator or the malicious intruder may find in this database is a long table of "hashes" of considerable size, corresponding to the result of the hashed concatenation of (Username and password) by the hash key K'usd. In other words, the adversary could only find less important and completely useless information because:

- Do not reveal in any case the real identities of users.
- Do not reveal the pseudonyms used by users.
- Do not reveal users' passwords.
- Do not allow the adversary to use this information to impersonate a registered user (to use a resource, the application executed by the USD, requires the user to choose a pseudonym and to enter a password the application will handle by itself the concatenation and the hashing process.

With regard of PPS, even if it is considered as a trusted entity "Trusted Third Party", the information stored there is just as useless for an opponent, and not allowing it to harm the identity and the privacy of the user. Indeed, the information stored on the users table in the PPS corresponds to the same information present in the AS, and enjoys, thus, the same advantages mentioned above vis-à-vis the protection of identity and privacy.

In the case where the opponent gets into a resource of pervasive space, the caught data during the authentication request will be "unreadable" because encrypted by the public Key of the PPS (Kpps), the secret random key (K"usd) and the public key of the AS (Kas).

The same applies to the phases of data exchange between the other entities of the system:

- USD→PPS: Packets encrypted by the public key of the PPS (Kpps) and the secret random key K"usd.
- PPS→AS: Packets encrypted by the public key of the AS (Kas).
- PPS→USD: Packets encrypted by the shared secret key (K'usd).

At the end, and in the case of theft of the USD of a legitimate user by an adversary, and despite the fact that this adversary can receive pseudonyms of the legitimate user on the USD from the PPS (the adversary designates that he/she is already registered by pressing the button "Old"), the adversary cannot nevertheless impersonate the legitimate user, for the simple reason that he/she does not know the user's password and cannot impersonate the legitimate user. This is known as a two factors authentication method because the right user needs his USD and the corresponding password to successfully authenticate correctly, if one of the two factors is missing, authentication becomes impossible. Conversely, if the opponent gets the user's password, he/she cannot access the resources, on behalf of the latter for not having the USD of that user, which is the only device to contain the unique key K'usd permitting to hash the concatenation Pseudonym/password and thereby reconstruct the corresponding hash that which was previously stored at the AS during the registration phase.

6. Evaluation of Proposed Protocol

Consider the following example: we suppose that in a given period of time, three users, u_1, u_2 and u_3 submit requests r_1, r_2 and r_3, respectively to the PPS, including their pseudonyms, passwords, and contextual data such as time and identity of the sought resource. Then the PPS applies a K-anonymity with K=10, by generalizing the contextual data (time, resource IP, etc.,) to expand the set of resources potentially issuers of such queries. We denote the generalized queries by r_1', r_2', r_3'. This phase will make the issuers devices indistinguishable in what we called an anonymity set of ten potential devices $\{d_1, d_2, d_3, ..., d_{10}\}$. Once the three generalized requests r_1', r_2' and r_3' are received by the AS (or intercepted), an adversary or a malicious server administrator may not accurately associate a request r_i' with a issuer device d_j. Indeed, for an adversary, each request r_i initially has a probability of 1/10 to have been emitted by a device d_j. The example above shows how k-anonymity can be advantageous for the preservation of privacy against attacks based on the analysis of the queries issued by users, and external knowledge. However, in the context of pervasive systems another class of attacks must be considered. This is the type of attacks based on the observation of the user's behavior and the server responses following this behavior. To ensure this kind of attacks, the adversary must (by electronic or physical way) be able to observe the movements and the behavior of the potential authentication request issuers, belonging to the same anonymity set, within the pervasive workspace.

6.1. Formal Definition of Attack Based on External Knowledge

Before evaluating and demonstrating the effectiveness of our protocol, it is necessary to establish a platform to formalize concepts such as attacks, external knowledge, the user's behavior and the server responses to the user's queries. For this reason, the formal framework proposed in [5] is taken into account, and is extended to consider the attacks based on external knowledge. Given a set of queries and generalized requests R, a set of users' identities I, and external knowledge Γ in possession of the adversary. An attack is defined in [5] as follows: An attack based on knowledge Γ is a function $Att\ \Gamma$: $R \times I \rightarrow [0, 1]$ such that for each generalized request r'.

$$\sum_{i \hat{I} I} Att_{\Gamma}(r', i) = 1 \quad (1)$$

This means that the value of $Att_{\Gamma}(r', i)$ is the probability which an adversary can know that the user i is the issuer of the generalized request r', by performing an inference attack on the basis of external knowledge Γ. This definition formalizes the attacks based on listening to the generalized requests and the external knowledge that the attacker posses about the users. But, the movements and actions, and more generally the behavior of the users, as well as the responses from the AS as a result of these behaviors are not included in this definition. Even if these data can be considered as part of the external knowledge available to the adversary, it is necessary to extend the definition proposed in [5] to cover all of the properties characterizing the attacks based on the observation of the behavior of individuals within the pervasive space and analysis of server responses to requests for identification/authentication. Therefore, the following definitions are then added: Given external knowledge Γ available to the adversary, the anonymity set θ=Ano $\Gamma(r)$ corresponding to the potential issuers (i.e., issuers whose context data contained in their requests are similar to those of other users of the same anonymity set), and the time interval t, the user's behavior is represented by $\Omega_{\Gamma, \theta, t}$, which corresponds to the context data of the users belonging to the set θ (actions or movements in space), in a time interval t. Considering an AS, a generalized request r' intended to this AS, the external knowledge Γ available to the adversary, and the anonymity set θ. $\mu_{as, \theta, t'}$ represents all the authentication requests sent by the issuers belonging to θ during the time interval t'. Attack by inference based on the user's behavior $\Omega_{\Gamma, \theta, t}$, and

analysis of the authentication requests μ_{as}, θ, t' (t' immediately following t) can then be formalized by:

$$\sum_{i \hat{I} I} Att'_{\Omega \Gamma, \theta, t, \mu \hat{a} s t', \theta, t'}(r', i) = 1 \quad (2)$$

From the moment that the adversary can observe the physical behavior of users within the pervasive space and receive or intercept the user's requests, including their identifiers (pseudonyms/password) and other context data, the adversary can then perform an attack to reduce the level of anonymity provided by the system, or in the worst case, definitely associate the physical identity of the user to his/her virtual identity (pseudonym) in the system, as well as the devices that has used and the personal data they have left after passing through the system. In the case where users u_1 and u_2 decide to authenticate in the same time interval t to use a resources of the pervasive space, an adversary cannot be sure that u_1 is at the origin of the request r_1', and that u_2 issued the request r_2' or vice versa. However, the degree of anonymity initially guaranteed to u_1 and u_2 decreases from 10-Anonymity to 2-nonymity. In addition, if the user u_3 is the only person who moves to a resource to authenticate before he/she can use it, the adversary can unambiguously associate u_3 to his/her request r_3' and affect his/her anonymity and privacy. Table 2 shows the probability that an adversary can know with success that a user u_j is the issuer of the generalized request r_i, thanks to an attack by inference $Att'_{\Omega \Gamma, \theta, t \mu as, \theta, t'}(r', i)$. The left column represents the queries issued in the same time interval t ("¬" means that no query was recorded during the time interval t), whereas the right column shows the probability that a user can be recognized as the issuer of the request r_i.

Table 2. Probability that an individual u_j is the issuer of the request r_i.

	$Att'_{\Omega \Gamma, \theta, t, \mu as, \theta, t'}$ (r', i)	u_1	u_2	u_3
t0-t1	r_1' r_2'	0, 1	0, 1	¬
t1-t2	r_3'	¬	¬	0, 1
t2-t3	r_1' r_2' r_3'	0, 1	0, 1	0, 1

This table clearly shows that the application of K-anonymity by the PPS is not enough to ensure an acceptable level of privacy against an inference attack, especially when the number of queries issued in a given time interval is reduced. In the purpose to address this lack and enhance our protocol against this type of attacks, that we developed a process of generating fake authentication requests, at the PPS before sending all requests (real and fake ones) simultaneously. If we take for example the 3rd row of the Table 2, instead of transmitting three requests to the AS, the PPS generates fake identification/authentication requests (for example, seven fake requests) using existing profiles of users registered in its database before transmitting them to the AS. The probability that an adversary successfully infer a request r_i has been issued by a user u_j is then reduced from 1 chance/ 3 (≈ 0.33) to 1 chance/10 (0.1) for this example. We obtain the same result for the worst case scenario, which means, when u_3 is the only issuer during the time interval t. While the adversary could unambiguously associate u_3 to his/her request r_3' and affect his/her privacy, the generation of false requests by the PPS reduces by 10 the probability that an adversary correctly associates u_3 to his/her request r_3'.

Table 3. Probability of disclosure of privacy after generation of fake requests by the PPS.

	$Att'_{\Omega \Gamma, \theta, t, \mu as, \theta, t'}$ (r', i)	u_1	u_2	u_3
t0-t1	r_1' r_2'	0, 5	0, 5	¬
t1-t2	r_3'	¬	¬	1
t2-t3	r_1' r_2' r_3'	0, 33	0, 33	0, 33

As we can see in Table 3, the use of fake requests significantly improves the level of protection of the user's privacy. Indeed, for 10 queries (true and false) sent in the same time interval t, the probability that an adversary makes a correct relationship between the query and the user declines to 0.1 for each user in the system.

Obviously we can easily find that if only one user is present in the pervasive space, its location (the resource being used) cannot be concealed for this type of attack. But the role of the PPS remains nevertheless important because the real user query containing his identifiers and possibly other information can at least be obfuscated in a batch of fake requests received by the AS, which will be unable to discern which of the request was really submitted by the user present in the pervasive space. The novelty and advantages of the shadow protocol can then be summarized as: Its ability to preserve the anonymity of users and this, thanks to the mechanism of registration and updating pseudonyms, making a given user indistinguishable from other potential users during its passage through the system. Its ability to conceal the user's identity to the PPS, and conceal the resource from which the user is authenticated to the AS, thanks to the K-anonymity phase which makes the generalization of the query so that the exact origin of the request is merged into a set of potential resources.

Its original way to combine different cryptographic techniques to prevent network attacks without the need to recourse to secure communication methods such as SSL. Its robustness against attacks based on observation of physical behavior and analysis of the users' queries, through the generation of fake requests by the PPS that significantly reduce the chances for an adversary to correctly associate a user (physical) to its logical identifier and messages (requests).

The opportunity for the user to choose from a list of valid pseudonyms for the day instead of using the same pseudonym for identification/authentication, in

addition to the daily update of the list of pseudonyms, prevents traceability of the users by a third malicious party. Its ability to guarantee exclusive access to legitimate users, and ensure their total anonymity and privacy through the two factors authentication system (USD and the associated pseudonym and password).

The ability to authenticate legitimate users and give them the rights to access resources, without any identifier being required or stored in any final or intermediate entity of the system (trusted or not), and opens the way to a new paradigm of authentication without identification.

7. Conclusions

Through this study we arrive at the conclusion that the notion of identity is not the same in a pervasive environment as in a conventional computer system. The protection of personal data (name, social security number, date and place of birth, etc.,) is no longer sufficient to ensure the anonymity and privacy of users. Some data that seems harmless can identify an individual (quasi-identifiers) if they are combined together, such as physical movements and behavior of an individual. The application of state of the art solutions dedicated to the security does not meet all the needs expressed in terms of protection of the user's privacy.

K-anonymity and Mist are the solutions that most resemble the ideal solution, but the first has difficulty to guard against inference attacks, and the second does not take into account the possibility of an adversary beign able to observe the users' behavior within the pervasive space. This would destroy the effectiveness of such a system. This latest possibility is highly probable, as soon as we talk about open and frequently visited systems such as pervasive systems. In order to address this failure we proposed the shadow protocol, an authentication protocol providing a real alternative to pervasive systems that must support frequent visits while protecting the privacy of its users. Indeed, the idea of conducting the shadow protocol is to allow the user to login/authenticate and acquire rights in the pervasive space without an adversary, or a malicious system administrator being able to discern him/her from other users (real or fictitious) registered in the AS.

It seems obvious that greater the number of requests (true or false) is sent, the less likely an adversary associates these requests to their issuers. It is for this reason that our next perspective is to find a good balance (threshold) between a good ratio of privacy and an acceptable response time. The perspective of connecting multiple pervasive spaces to the same PPS is also in study, in order to increase the number of participants and therefore increase the anonymity set for a better requests obfuscation.

References

[1] Al-Jaroodi J., "Routing Security in Open/Dynamic Mobile Ad Hoc Networks," *The International Arab Journal of Information Technology*, vol. 4, no. 1, pp. 17-25, 2007.

[2] Al-Muhtadi J., Campbell R., Kapadia A., Mickunas M., and Yi S., "Routing through the Mist: Privacy Preserving Communication in Ubiquitous Computing Environments," *in Proceeding of 22nd International Conference on Distributed Computing Systems*, Vienna, pp. 74-83, 2002.

[3] Anonymizer, http://www.anonymizer.com, last visited 2012.

[4] Bellotti V. and Sellen A., "Design for Privacy in Ubiquitous Computing Environments," *in Proceeding of 3rd European Conference on Computer Supported Cooperative Work*, Milan, pp. 77-92. 1993.

[5] Bettini C., Mascetti S., Wang X., and Jajodia S., "Anonymity in Location-based Services: Towards a General Framework," *in Proceedings of the 8th International Conference on Mobile Data Management (MDM)*, Washington, pp. 69-76. 2007.

[6] Brodie C., Karat C., Karat J., and Feng J. "Usable Security and Privacy: a Case Study of Developing Privacy Management Tools," *in Proceedings of the 2005 Symposium on Usable Privacy and Security*, Pennsylvania, pp. 35-43. 2005.

[7] Campbell R., Al-Muhtadi J., Naldurg P., Sampemane G., and Dennis M., "Towards Security and Privacy for Pervasive Computing," *in Proceeding of 3rd IEEE Conference on Pervasive Computing and Communications Workshops*, USA, pp. 194-198, 2005.

[8] Cardoso R. and Issarny V., "Architecting Pervasive Computing Systems for Privacy: A Survey," *in Proceeding of IEEE/IFIP Conference on Software Architecture*, pp. 26-29, 2007.

[9] Cheng H., Zhang D., and Tan J., "Protection of Privacy in Pervasive Computing Environments," *in Proceeding of International Conference on Information Technology: Coding and Computing*, NV, pp. 242-247, 2005.

[10] Christin D., Roßkopf C., Hollick M., Martucci L., and Kanhere S., "IncogniSense: An Anonymity-preserving Reputation Framework for Participatory Sensing Applications," *in Proceeding of IEEE International Conference on Pervasive Computing and Communications*, Lugano, pp. 135-143, 2012.

[11] Culnan M., "Protecting Privacy Online: is Self-Regulation Working?," *Journal of Public Policy and Marketing*, vol. 19, no. 1, pp. 20-26, 2006.

[12] Hengartner U. and Steenkiste P., "Exploiting Hierarchical Identity-Based Encryption for Access Control to Pervasive Computing Information," *in Proceeding of 1st International Conference on Security and Privacy for Emerging Areas in Communications Networks*, Athens, pp. 384-396, 2005.

[13] Hofheinz D. and Kiltz E., "Secure Hybrid Encryption from Weakened Key Encapsulation," *in Proceeding of the Twenty-Seventh Annual International Cryptology Conference*, California, pp. 553-571, 2007.

[14] Hong J. and Landay J. "An Architecture for Privacy-Sensitive Ubiquitous Computing," *in Proceeding of 2nd international Conference on Mobile Systems*, Massachusetts, pp.177-189. 2004.

[15] Inald R., Lagendijk L., Erkin Z., and Barni M., "Encrypted Signal Processing for Privacy Protection," *IEEE Signal Processing MAGAZINE*, pp. 82-105, 2013.

[16] Jovanovic M. and Munoz M., "Analysis of the Latest Trends in Mobile commerce using the NFC Technology" available at http://uowdnfc.blogspot.com/2012/07/journal-1-analysis-of-latest-trends-in.html, last visited 2012.

[17] Langheinrich M. "A Privacy Awareness System for Ubiquitous Computing Environments," *in Proceedings of the 4th International Conference on Ubiquitous Computing*, Sweden, pp. 237-245. 2006.

[18] Langheinrich M. "Privacy by Design-Principles of Privacy-Aware Ubiquitous Systems," *in Proceedings of the 3rd International Conference on Ubiquitous Computing*, Georgia, pp. 273-291, 2001.

[19] Organization for the Advancement of Structured Information Standards (OASIS), available at: https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wss, last visited 2004.

[20] Pareschi L., Riboni D., Agostini A., and Bettini C., "Composition and Generalization of Context Data for Privacy Preservation," *in Proceedings of 6th Annual IEEE International Conference on Pervasive Computing and Communications*, Hong Kong, pp. 429-433. 2008.

[21] Reed M., Syverson P., and Goldschlag D., "Anonymous Connections and Onion Routing," *IEEE Journal on Selected Areas in Communication*, vol. 16, no. 4, pp. 482-494, 1998.

[22] Rigney C., Rubens A., Simpson W., and Willens S., "RFC 2138: Remote Authentication Dial in User Service (RADIUS)," *Technical Report*, 1997.

[23] Satyanarayanan M., "Privacy: The Achilles Heel of Pervasive Computing?," *IEEE Pervasive Computing*, vol. 2, no. 1, pp. 2-3, 2003.

[24] Sweeney L., "k-Anonymity: A Model for Protecting Privacy," *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, vol. 10, no. 5, pp. 557-570, 2002.

[25] Wang J., Floerkemeier C., and Sanjay E., "Session-Based Security Enhancement of RFID Systems for Emerging Open-Loop Applications," *in Proceeding of Identification Information and Knowledge in the Internet of Things Beijing*, China, pp. 18-20, 2013.

[26] Weiser M., "The computer for the 21st century," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 3, no. 3, pp. 3-11, 1999.

Mohammed Djedid is a PhD student at University of Sciences and Technology of Oran. His fields of research are user's privacy and anonymity preserving in Pervasive systems. He obtained his Master of Information Systems Engineering in 2009. He has published in several peer reviewed journals, and participated at specialized conferences in the field of pervasive systems (Ex: Int' Conf. on Pervasive and Embedded Computing and Communication Systems, and forward thinking colloquy –Milwaukee, USA-). He is also a reviewer in the Wireless Personal Communications journal edited by Springer.

Abdallah Chouarfia is a Professor in the Department of computer sciences at University of Sciences and Technology of Oran (USTO-MB) Algeria. He received the B.E. degree in Computer science from CERI Algiers and Ph.D degree in software engineering from Paul Sabatier University Toulouse France (1983). He has more than 30 years of teaching experience. He has published more than 20 papers in International, National journals and conference proceedings. His areas of research are Software engineering and Networking including mobile Ad-Hoc and security.