Timestamp Authentication Protocol for Remote Monitoring in eHealth

Kalid Elmufti, Dasun Weerasinghe, M Rajarajan, Veselin Rakocevic, Sanowar Khan
Information Engineering Research Centre
School of Engineering and Mathematical Sciences
City University,
Northampton Square, London, EC1V 0HB, UK
k.elmufti@city.ac.uk

Abstract— **Remote monitoring is fundamental in eHealth and introducing mobile devices in the remote monitoring process can provide additional benefits to both patients and medical personnel. For mobile remote monitoring systems to be successful, however, the authentication process must be in place to prevent the misuse of the system. In this paper we analyse the use of timestamps in the authentication process, showing many advantages timestamps have over other authentication methods. The paper presents the design principles for timestamp based authentication protocols in remote monitoring systems and proposes a specific protocol to implement such a system.**

I. Introduction

The recent technical advances in communication systems have had impact on all aspects of our everyday life, and the field of healthcare is no exception. From here the term ehealth or electronic health was developed; it is defined as the use of a wide range of hardware and software in support of health care.

The need for ehealth varies. In some places health centers and hospitals are hundreds of miles from patients homes, the presence of ehealth systems allows timely access to quality health services like tele-monitoring, tele-diagnosis, and e-prescription at low costs leading to improved quality of life of citizens and greater economic productivity.

Chronic disease of all types of obesity, diabetes, asthma, and cardiovascular illness are on the increase, especially as western populations age. Research has shown that clinical outcomes and patient well being are significantly enhanced by self management. Regular measurement of vital signs, glucose levels or blood oxygen levels and other key parameters mean that diet, exercise or medication can be controlled. This is one of the main objectives of ”Remote Monitoring”. Remote monitoring allows for an individual to the various physiological parameters and send it to a remote server, where the general practitioner, specialist nurse or a consultant is able to view the data.

A possible scenario here could be: a patient using a sensor to measure vital signs, which then are transferred to their mobile phone via bluetooth. Using the mobile phone network then gets uploaded to a remote server or a Healthcare Authentication Server (HAS). At the server side a specialist can view the data and take the appropriate action.

PERVASIVEHEALTH 2008, 30 Jan - 1 Feb. Tampere, Finland

DOI 10.4108/ICST.PERVASIVEHEALTH2008.2542

As ehealth remote monitoring is dealing with sensitive data, the security of the system and the privacy of the data are important issues that must be considered for such applications. The following are key security issues to be addressed when developing such systems:

- Authenticating the patient to the HAS.
- Authenticating the specialist to the HAS, with the appropriate level of access.
- Protecting the confidentiality of data during transmission and while stored.
- Protecting the integrity of data during transmission and while stored.
- Prevent replay attacks.

In this paper we focus on the first two points, highlighting the issue of authentication. Depending on the application in concern, the authentication process can be a costly process. We propose a case of authentication scheme/protocol that is both secure and efficient. We propose the use of timestamp based authentication protocol.

One of the key factors to determine the meaning of a message in an authentication protocol is the “Time variant parameters”. There is a need to believe that the message is fresh before action upon it; and that the message is not a reply of an old instance.

Time variant parameters may be used in authentication protocols to prevent replay and interleaving attacks, to provide uniqueness or timeliness guarantees, and to prevent certain chosen-text attacks [4].

II. Motivation for using Timestamps in Authentication Protocols

The notion of time is fundamental for describing and verifying security properties related to the expiration of keys and the freshness of messages [3]. Timestamps are necessary in authentication protocols that support multiple authentication without multiple request to an authentication server [6]. The motivation for using Timestamps in Authentication protocols can be summarized as follows:

- Timestamps may be used to provide timeliness and uniqueness guarantees, which guarantees the freshness of a message.

- As a result of the above point message replay and forced delay can be detected.
- Timestamps may also be used to implement time-limited access privileges.
- Timestamps in authentication protocols offer the advantage of fewer messages (typically by one) than other challenge-response protocols. This is of special importance in mobile networks such as GSM/UMTS where there is usually a cost associated with each message sent.

III. The Proposed Protocol

In this section we propose a timestamp based authentication protocol for ehealth remote monitoring system. The protocol benefits from features provided by timestamp based authentication protocol, namely: providing freshness of the messages, and protecting from replay attacks. Further to this, the protocol can improve the efficiency of the authentication process, since timestamp based protocols use least one less message than challenge/response systems [5].

A. The Environment

This section presents the main actors involved in the system and describes the overall architecture of the platform. Our Platform consists of there main actors:

- The Healthcare Center (HC) which consists of Electronic Healthcare Records Database (EHRD), which contains and stores patient data, and Healthcare Authentication Server (HAS), that grants the appropriate access to the EHRD and provides the remote access to both the staff users and the patient users.
- Staff Users (Staff): This refers to any user working for the healthcare center, such as doctors, hospital administrators, etc.
- Patient Users (Patient): This refers to the patients who use the remote monitoring service provided by the healthcare center to send their. The patient user is equipped with a sensor (Sen) to measure the vital signs, and a GSM/UMTS mobile phone or a mobile device (MD) . The data collected by the sensor are transferred to the mobile phone via bluetooth connection.

The entities above interact with each other as described in Figure 1.

B. Prerequisites for protocol

The proposed protocol uses digital cryptography to protect the integrity and the confidentiality of the messages in the system, such techniques are detailed in [4]. The following requirements must be met prior to the use of the protocol.

- All actors have agreed on a specific signature algorithm. The signature on data X using private key K is written $s_K(X)$.
- The HAS has an asymmetric key pair for a signature scheme, and all the actors have a trusted copy of the public key of the HAS.

The proposed protocol has the following assumption for the environment:

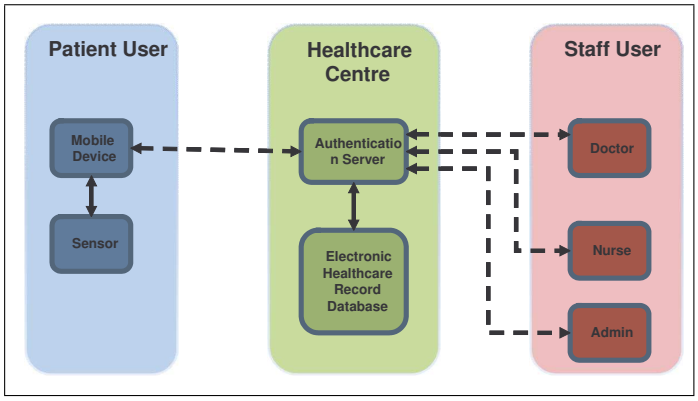

Fig. 1. Remote Monitoring eHealth system architecture

- The clocks used by the communicating devices are not precisely synchronized. And the differences in clock values are less than some threshold value *T*.
- Messages sent from one device to another are subject to a maximum transit delay of *D*.
- Given the current time t_c, and the timestamp in the message is t_m, the message is considered 'fresh' if t_c-*T*-*D*$\leq t_m \leq t_c$+*T*.
- From the above, the time of acceptance interval or the 'window of acceptance' is [t_c-*T*-*D*,t_c+*T*].
- It is assumed that each device in the system possesses a clock that is reasonably reliable (e.g. accurate to within a few seconds a day).
- A timestamp based authentication protocol is used between pairs of devices (i.e. Staff/Patient Users and HAS) in the system to guarantee the freshness of the messages, and that the protocol is designed in a way that the recipient of a timestamped protocol message can guarantee its origin and integrity by cryptographic means.
- Each device maintains a 'clock offset' value, used purely for the purposes of entity authentication. Specifically, the time value used for the security protocol purposes is always the sum of the clock value and the clock offset value. If the clock value is ever adjusted, then the clock offset value must also be adjusted to ensure that the sum of the clock value and the clock offset is never decreased. This would most easily be achieved by ensuring that if the clock value is moved back by δ seconds, then the offset is simultaneously increased by δ seconds. To avoid unnecessary increases in the sum of the clock value and the offset, in the same way if the offset is reduced by δ seconds whenever the clock is moved forward by δ seconds. Since it was assumed that the clock offset value is never reduced (except as described immediately above), the sum of the clock value and the clock offset never decreases.

Therefore in order to detect message replays within this window, the recipient must retain copies of all messages received and accepted within this window.

Now, when a device receives a protocol message containing a (protected) timestamp, this timestamp is compared with

the current clock value for the device in the following way. Suppose the timestamp in the message is t_m, the current clock value is t_c, the current clock offset value (stored within the device) is t_o, and *T* and *D* are as above. Then the message is accepted as fresh as long as the following inequality holds:

$$t_c + t_o - T - D \leq t_m$$

Moreover, if $t_m > t_c + t_o$, then $t_m - t_c - t_o$ is added to the clock offset value. When a device sends a protocol message containing a timestamp, this timestamp is set equal to $t_c + t_o$.

All devices maintain a list of all messages received with a timestamp t_m satisfying $t_c + t_o - T - D \leq t_m$. As soon as the timestamp t_m in a stored message no longer satisfies this inequality it can be discarded. Each received message is compared with the current list of stored messages, and discarded if it matches. Note that this was achieved without any assumptions abut routine clock resynchronization.

There are possible problems with some attacks. However, depending on the environment in which the protocol is used, these problems may not be significant. For networks where devices have limited communications resources, e.g. mobile and ad hoc networks, the proposed timestamp management scheme may be an efficient alternative to the use of nonce based protocols [3].

C. Issues with using Timestamps in Authentication Protocols

The topic of timestamps is not new in the security literature. [8], [5], [2] highlight various issues when using timestamps in communication protocols. The following are the main issues of relevant concern.

1) Clock Synchronization: Timestamp based protocols require that time clocks be both synchronized and secured. The possibility of adversarial modification of local time clocks is difficult to guarantee in many distributed environments [5], [2]. While technical solutions exist for synchronizing distributed clocks, if synchronization is accomplished via network protocols, such protocols themselves must be secure, which typically requires authentication; this leads to a circular security argument if such authentication is itself timestamp based [4].

2) Trusted Clocks: As mentioned before, the timestamps based protocol can help to ensure freshness of messages [2], [8]. The freshness is usually adhered as follows. The party originating a message obtains a timestamp from its local clock, and cryptographically binds it to a message. Upon receiving a timestamped message, the second party obtains the current time from its own clock, and subtracts the timestamp received. The received message is valid provided the timestamp difference is within the acceptance window.

Therefore, there is a need for a level of trust that the other party clock is functioning correctly. In a server/client communication it is generally assumed that the server clock is trusted to function as expected. However, the same can't be easily assumed at the client side. This is an issue when mutual authentication is required and if the authentication protocol is timestamp based. The clocks in most clients handsets in GSM/UMTS systems can be easily modified by the users (for good reasons such as setting the time of their mobile phones). Therefore if such devices (e.g. mobile phone) are used in such protocol, there will be a need to establish the trust ensuring that all clocks are behaving as expected.

D. Protocol Operation and Analysis

In order to use a timestamp based authentication, a degree of trust must be in place between the various clocks used in the system. One of the main objectives behind the scheme is to increase the level of trust in the Patient's clock, by getting the HAS to approve the time from the user's clock (both the staff and Patient). When the Patient (i.e MD) authenticates to the HAS to obtain a 'user token' (UT) to access an EHRD. The HAS will attach a signed value of the MD current time $t_{c_{MD}}$ and the HAS current time $t_{c_{HAS}}$ to the UT. So when the user accesses an EHRD, the EHRD can compare the MD timestamps, the one attached with the UT and the timestamp sent with the message t_m. This way the EHRD will be able to detect if the users clock has changed significantly since the session started (i.e. since the user authenticated to the HAS) to take an appropriate action. The proposed scheme takes the following steps:

Message 1
Patient → HAS: t_m,Access Info
Message 2
HAS → Patient: $s_{SK_{HAS}}(t_{Patient}, t_{HAS})$, UT
Message 3
Patient → EHRD: $s_{SK_{HAS}}(t_{Patient}, t_{HAS})$, UT, t_m

In message 1 the user will attach its current timestamp t_m with its security credentials (i.e. Access Info) to access the federated environment through the HAS. The HAS will compare the user's t_m with its local clock t_{HAS}. If the different in times (both directions) is bigger than a threshold value *T*, the HAS will ask the user to adjust its clock by sending a reply message with the its current timestamp t_{HAS}. On the other hand if the difference in time is less than *T* and the verification of the Access info (e.g. Username/Password) is successful, then the HAS will do two things:

- Generate a User Token UT.
- Attach its timestamp t_{HAS} to the user timestamp t_{user} (note that at this stage $t_m = t_{user}$) and sign them with its signature private key SK_{HAS}.

In message 2 the HAS will send to the user the information generated above $s_{SK_{HAS}}(t_{Patient}, t_{HAS})$, UT.

In message 3 the Patient will try to access the EHRD by providing its UT, its current timestamp t_m, and the message from the HAS $s_{SK_{HAS}}(t_{Patient}, t_{HAS})$. The EHRD now verifies the Patient's data in the following way (note: it is assumed that the EHRD clock is synchronized with the HAS clock):

- Verify the UT (this is subject to the federated system deployed).
- Verify the message $s_{SK_{HAS}}(t_{Patient}, t_{HAS})$ using the HAS signature public key PK_{HAS} and if successful

- Compare the Patient's timestamps; $t_{Patient}$ from the HAS message with t_m which is the Patient current time. Message will be accepted if the difference between t_m and $t_{Patient}$ is less than *T*. This will ensure to the EHRD that the Patient's clock is correct (i.e. synchronized with the HAS clock)

If the above conditions are met then a timestamp based authentication protocol can be used after this stage to authenticate the user to the EHRD. Note that no extra message were introduced, since these are the normal message exchange in the federated environment scenario [1].

Some of the major issues with timestamp authentication protocols are clock synchronization presented in section III-C.1 and trusted clock issues which were presented in section III-C.2. The proposed scheme solves the Patient's trusted clock issue, in another word the EHRD can now trust the Patient's clock as its value has been tied to the HAS trusted clock at the beginning of the session. That enables the EHRD to detect any changes in value to the user's clock.

Once the authentication process is completed, the Patient now can upload the data obtained by the Sensor to the EHRD via the MD.

The Staff users (e.g. doctors, specialist nurse or a consultant) will go through the same steps of authentication as the Patient user to obtain access to the EHRD, with the appropriate level of access [7]. Further more; as the authentication is a timestamp based it can be utilized to implement time-limited access privileges for the Staff user's. If the authentication is successful a Staff user (e.g a doctor) can now review the Patient data (e.g. vital signs) and take the appropriate action.

IV. Conclusion

The efficiency gain that timestamp based authentication protocol can offer over other authentication protocols can be of great value, especially when mobile phones are used in the system, since there is usually an associated cost related to the number of messages sent. However the issue of clock synchronization and trusted clock are of great concern in such schemes. In this paper we demonstrated how to address these concerns and propose a mechanism to use timestamp authentication protocol in remote monitoring even if mobile devices with untrusted clocks are used. Next we plan to build a system demonstrating the proposed system to allow for more accurate measurements, to determine the practical efficiency gain by timestamp authentication protocols.

References

[1] Liberty ID-FF Architecture Overview. Technical report, Liberty Alliance, April 2003.

[2] Dorothy E. Denning and Giovanni Maria Sacco. Timestamps in key distripution protocols. *Communications of the ACM*, 24(8):533–536, August 1981.

[3] Nevin Heintze and J. D. Tygar. Timed models for protocol security. Technical Report CMU-CS-92-100, 1992.

[4] A.J. Menezes, Paul van Oorschot, and Scott A. Vanston. *Handbook of Applied Cryptography*. CRC Press Inc, 1996.

[5] Chris J. Mitchell. Timestamps and authentication protocols. Technical report, Royal Holloway, University of London, February 2005.

[6] B. Clifford Neuman and Stuart G. Stubblebine. A note on the use of timestamps as nonces. *Operating System Review*, 27(2):10–14, April 1993.

[7] Weerasinghe, D.,Elmufti, K., Rajarajan, M. and Rakocevic, V. Securing electronic health records with novel mobile encryption schemes. *Int. J. Electronic Healthcare*, 3(4):395 416, 2007.

[8] Eun-Jun Yoon and Kee-Young Yoo. Efficient mutual authenticaion scheme with smart card. *Lecture Notes in Artificial Intelligence*, 4088:813–818, 2006.