Privacy in Mobile Web Services eHealth

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

Abstract

There are many advantages to deploy Mobile devices in eHealth systems, however; security is still an issue in particular authentication and User privacy. In this paper we propose a novel architecture which integrate 3GPP UMTS mobile technology with Web services that addresses authentication and privacy concern within eHealth environment. The architecture make use of the Generic Authentication Architecture from the 3GPP and the Single Sign-On system to authenticate mobile users to a Health Authentication Server to give access to various eHealth Service Providers, P3P is used to manage privacy in the system.

1. Introduction

eHealth addresses both changes in the access of healthcare information and services. This transformation, enabled by eHealth challenges the traditional roles of hospitals and clinics. The next phase includes the use of mobile devices to provide a user friendly interface to bring healthcare services directly into the personal space of the users. There are many other benefit than patients convenient to use mobile technologies; managed care; regulations and costs in healthcare are great gains when adopting mobile technologies in health care. However; mobile solutions also bring with them their own challenges. Given that these systems will be transmitting highly sensitive information such as patient data, implicit in their use is a need for high level of end-to-end security, confidentiality and privacy.

* This work was supported by sponsorship funding from City University, London

There are many entities involved when dealing with patient health care. This is true in terms of personal such as doctors, nurses, management staff, insurance personal, etc; or in terms of data including patient medical records, doctor prescription, X-ray images, bills, etc. Most of these data potentially very sensitive from the patient prospective. In this paper we propose a novel communication platform that allows the healthcare systems to take advantages of using mobile technologies at the same time ensuring users privacy.

The novelty of the system is in the integration of the Web Services technologies such as Single Sign-On (SSO)[7] and The Platform for Privacy Preferences Project (P3P) [1], with the Generic Authentication Architecture (GAA)from the Third Generation Partnership Project (3GPP) [4].

The rest of the paper is dedicated to describe the architecture of the system and to show how this integration can take place. A proposed communication protocol for the system is also describe with specific section that analyse User privacy in the system. The paper conclude with a conclusion summarising the proposed architecture.

2. Generic Mobile Web Services Platform for Healthcare

This section present the main actors involved in the system and describe the overall architecture of the platform. Our Platform consists of four main actors:

- Mobile Operator: This is the Mobile Phone Network Operator (MO), and it contain the Bootstrapping Function (BSF) that is part of the GAA.
- Healthcare Authentication Server: (HAS) contain two main entities; the first is the Network Application Function (NAF) which is used to communicate withe BSF. The other entity is the Iden-

tity Provider (IdP), that is used as the identity provider for the SSO system.

- Service Providers: This refers to any service provider used by the healthcare system, such as Electronic Healthcare Records (EHR), pharmacy, doctors, hospital admin, healthcare insurance, etc.
- User: the user can be anyone with 3GPP UMTS Mobile Device (MD) who want to access the healthcare system. This can be a patient, a doctor, or any other healthcare system personal.

The entities above interact with each other as described in Figure 1. When the User request access to the HAS, the HAS will request the user's GAA security credentials. The User will start the bootstrapping procedure with the BSF of the MO as described in [6]. Using the Ub interface the BSF will generate for the User a random identifier (B-TID), and using the AKA protocol both (the BSF and the USER) will generate a secret session key K_s. The user will then use B-TID and K_s as user name and password to login to the NAF located at the HAS. If successful the User will be authenticated to the IdP SSO system such as [2] to access the various healthcare service providers that are registered with the HAS.

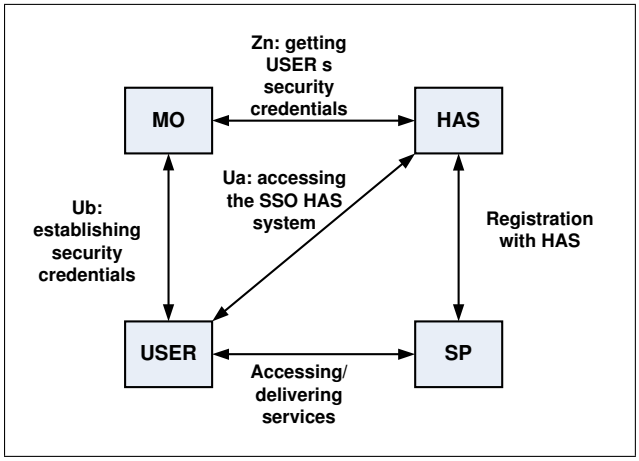

Figure 1. Mobile eHealth system architecture

The privacy of the User is managed as follows:

- All SPs to be part of the HAS must have a P3P privacy policy file. These policies describe who collects what data and for what purpose.
- All P3P privacy policy file must comply with the minimum standard set by the HAS, which is checked during the registration.
- The User must set their own P3P privacy file. When accessing a SP the two P3P files are compared (User and SP) if they match then the system will just set in the background; if not the User will be alerted and he/she can make decision to accept the service or not.

In addition the privacy of the user will be protected by encrypted all the messages sent by the user as well as changing the identity of the user each time the user accessing different SP. The details are given in the next section.

3. The proposed scheme

In our proposal the HAS acts as an Identity Provider between Web Service Providers and each of the Web Service Users (Mobile Stations). The Mobile Operator owning the SIM deployed in the Mobile Station, acts as an Authentication Authority to the HAS.

We utilise the combined Liberty & 3GPP GAA model, as defined in [5], to combine the Service Orientated Architecture of Web Services with a Mobile End User end point. We target the provision of identity-consuming services where knowledge of the user (principal) is important. In this way we address the highest value scenario; specifically:

- where the service is enhanced by knowledge of some data related to the identity of the principal (e.g. Healthcare records).
- where privacy, trust and authentication are highly relevant. (e.g. Medical insurance data)

With reference to Figure 2, in the GSM/3GPP mobile architecture, security and trust reside in two locations. These are the network HLR (Home Location Register) of the HSS (Home Subscriber System) and the Operator issued tamper resistant SIM card. We therefore consider the network HSS as the User owning entity.

A client application needs to run within the user device 'MD' in order to use the processing capabilities of the user device. However this user device is unlikely to be trusted by scheme entities to hold a valuable network level identity. [Note: This distrust is likely to increase as devices move from traditionally closed proprietary operating systems to more open operating systems capable of performing the file manipulation required by advanced 2.5G and 3G services].

The User owning entity — the network HLR — attests the identity of a particular consumer up to "a

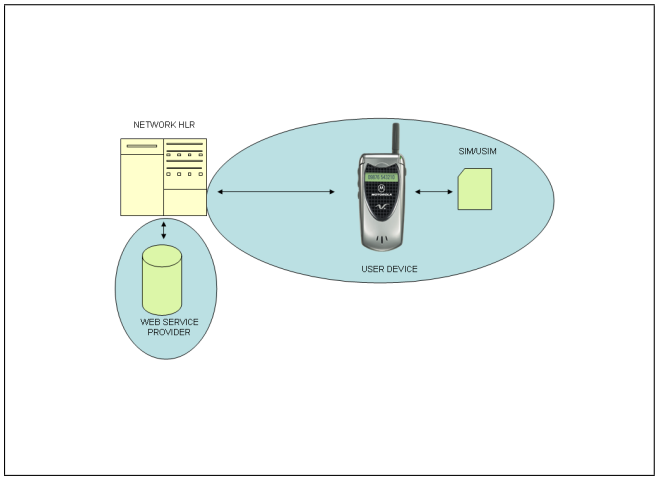

Figure 2. Security Model

certain level". Application layer credentials are bootstrapped from the (3G) cellular network mutual authentication process and provided to both the End User device and the Service Provider. This allows the Service Provider and End User to communicate securely as they now share the same secret.

We use the GBA or Generic Bootstrapping architecture of GAA as described in [5] to exploit the 3GPP Authentication and Key Agreement process to produce application credentials. The Mobile Station uses the Bootstrapping Server Function of the Mobile Operators Home Subscriber System to create these application layer credentials, i.e. GBA, over the Ub interface. These are then shared with the Identity Provider (IdP or sometimes referred to as the Network Application Function) via the Zn interface. The Mobile Station client can then communicate directly with the Service Provider using these credentials.

1. Once registered, the User Agent of the(UE) performs GBA_U with (BSF) over Ub.
2. The User Agent applet within the UICC is provided with Ub parameters.
3. The UICC component of the User Agent calculates the K_s and provides the ME with the service layer credentials ($K_s\_(int/ext)\_NAF$). The K_s always remains in the UICC.
4. The User Agent makes contact with the (NAF/IdP) to obtain a "HAS" identity.
5. Service credentials appropriate to the User Agent are communicated via Zn to the (HAS).
6. An authentication token for the "HAS" is provided to the User Agent from the (HAS).
7. ($User$) communicates with (SP) using service credentials and requests a service.
8. (SP) confirms validity of ($User$)'s service credentials.

This process allows the service provider to deliver an identity consuming web service direct to the End User, without having to resort to the use of end user certificates or setting up its own identification system.

The Mobile Station is assumed to implement a *Security Agent* function — an example of which is presented in [8]. The *Security Agent* comprises a device executed MIDlet application for I/O and computationally intensive operations, together with a tamper-resistant module (e.g. Trusted Programme Module (TPM) and/or SIM card) executed application for secure storage and cryptographic processing. The HAS is assumed to implement a Token Distribution Center.

3.1. Protocol

This section will outline the various stages of the protocol. There are mainly three stages: Registration, Authentication, and Service Delivery.

3.1.1. Prerequisites for protocol Our protocol uses both symmetric and asymmetric cryptographic techniques to provide the authentication and integrity services required.

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 $s_K(X)$.
- All actors have agreed on an asymmetric encryption algorithm, for which the encryption of data X using public key P is written $e_P(X)$.
- All actors except the consumer have encryption key pairs for encryption scheme, and all the actors possess a trusted copy of the public key of the other actors.
- All actors except the consumer have asymmetric key pair for a signature scheme, and all the actors possess a trusted copy of the public key of the other actors.

3.1.2. Registration There is two different level of registration each with its own requirement. The registration is for the SPs and Users to register with the HAS.

- SP registration (Compulsory): each service must register with the HAS to be part of the eHealth system. During the registration the SP must show

a valid P3P privacy policy file which must comply with the HAS policy.

- User registration: there are two type or user registration:
  - Compulsory User registration: this is for all the doctors or the Health service personal and for the patients who want to access their EHR or similar resources. This way the EHR can be updated by the appropriate identity.
  - Anonymous User registration (Optional): this can be done during the authentication stage, where an anonymous user will be given an temporary identifier by the HAS. This is valid only for patient User, to give them anonymity to access spacial services.

3.1.3. Authentication The SPs can authenticate to the HAS using any strong authentication schemas such as PKI, and this is out of the scope of this paper. User authentication occur to the following procedure:

1. The User sends a request to access the HAS, attaching with the request the User IMPI number.
2. Assuming the User has not been authenticated at this stage, the HAS will send a request to the User to initiate a new Bootstrapping Procedure (BSP).
3. We assume at this stage that the User does not have a valid bootstrapping session or the freshness of the key material is not sufficient. The User will initiate the BSP with the BSF via the Ub interface, the details are defined in [6].
4. The BSF will generate B-TID which is a string of base 64 random data and the BSF server domain name; it will also generate key material K_s which is the result of concatenating the Confidentiality Key (CK) and the Integrity Key (IK) resulting from the AKA protocol. The details of the generation of B-TID and the Ks are defined in [4]. The User will use the B-TID as the Username and the K_s as the password to access the HAS. B-TID will be sent to the User via the Ub interface along with the Key Lifetime, the password K_s will be generated by the user based on the AKA protocol and it will be stored in the UICC.
5. The User starts the login procedure by forwarding its 'Username' i.e. the B-TID to the HAS.
6. The HAS needs to obtain the User's password i.e. K_s that belongs to B-TID in order to be able to authenticate the User. This is done by the HAS sending the B-TID and its NAF hostname to the BSF via Zn interface, the details of this operation are defined in [4].
7. In response to step 6 the BSF will send to the HAS the User password i.e. K_s and the key lifetime (Note: other related data will be sent in this message, these data were omitted here for simplicity); the details of this operation are defined in [4] and the security of this message are defined in [3].
8. The HAS will challenge the User the possession of the password i.e. K_s. This step is required to protect against re-play attack. The HAS generates a random number RAND and sends it to the User.
9. After the User receives the RAND, the User will generate the ChallengeResponse and sends it to the HAS to prove the possession of the password i.e. K_s. The challengeResponse is a function of the RAND and K_s; ChallengeResponse = f(RAND,K_s); this operation will take place in UICC as K_s will never be reviled to the handset. It is assumed that both the User and the HAS uses the same function 'f' to generate the ChallengeResponse.
10. The HAS needs to verify the ChallengeResponse received in step 9, and if not successful it repeat step 8; if successful it will generate a UserToken (UT) and sends it to the User. The UT will be generated as follows: the IdP part of the HAS will generate a Temporary User ID (TUID), this will be used to access the SSO system in which the IdP acts as the Authentication Server. The TUID is derived by the IdP from the User ID (UID).

    Note: the NAF IdP mapping is done using a 'User map table', which maps the User's IMPI to the UID (or TUID).

    The UT will be built by concatenating the TUID to a date/time timestamp (TS), and signing the TUID||TS with the IdP digital signature private key $IdP_{ds:sk}$, and encrypting the result with the IdP encryption public key $IdP_{e:pk}$, such that the UT=$e_{IdP_{e:PK}}(s_{IdP_{ds:SK}}[\text{TUID}||\text{TS}])$.

    This UT will be sent to the User encrypted using the password K_s received in step 7.

    This concludes the authentication phase, all steps can happen at an earlier time before requesting access to any particular service provider, providing the lifetime of the keys have not been exceeded.

3.1.4. Service Delivery This phase involve the User to choose an SP that he/she require its service, select product or services, an optional payment phase if the product/service is

not free, and finally the SP will deliver the service to the User.

Service Selection:

11. Once the User receives the UT he/she can now request access to any service provider (SP) in the HAS, however to do that the User must first receive SP UserToken from the HAS. This is achieved by the RequestService message where the User sends the ID of the requested SP to the HAS concatenated with the UT.
12. The HAS now generates a specific UserToken for the User to be used only with the SP requested by the SPID from the RequestService message; this UserToken will be referred to as the SPUT.
13. The User can now talk directly with the SP requesting any services offered by this SP, the CallService message will contain the UserRequest and the SPUT. The UserRequest will be encrypted using the tsK to protect the User privacy.

    Note: it is assumed at this stage that when the User sends this message to the SP that the User is confirming his/her selection, which can be indicated in the UserRequest.
14. Once the SP receives the CallService message it decrypts the SPUT using its encryption private key $SP_{e:SK}$, it then verifies the signature of the SPUT, this is done by validating the SPUT using the HAS signature public key $s_{IdP_{ds:PK}}$, to ensure the integrity of the content of SPUT; if the validation is successful the SP compares the TUID (or UID) to its registered Users database if it exist, this option allows the SP to give customized services to its users. If payment is required then the SP gets the tsK from SPUT and use it to decrypt the UserRequest; the SP will reply with an 'Invoice', this Invoice will contain a confirmation of the UserRequest, Price, and a method of payment (e.g. Credit Cards only). The Invoice will be signed by the SP digital signature private key and encrypted with tsK.

    Payment (*Optional*):

    This phase is only necessary if a payment is required with the service.
15. The User verifies the invoice by decrypting it using tsK, and verifies the content of it. If the Invoice verification process is successful, the User now starts the payment phase. It is assumed that the User has an account with a Financial Service Provider (FSP), who will charge the User and pay the SP. However for the User to communicate with the FSP the User must obtain a SPUT for this FSP; this is done the same way as in steps 11, 12, and changing the SPID with the FSPID.
16. The HAS will send to the User SPUT and tsK to access the FSP.
17. The User forward the Invoice and the SPUT to the FSP in a CallService message, to indicate the User confirmation for the FSP to charge the User and pay the SP indicated in the Invoice.
18. The FSP will decrypt and validate the signature of SPUT (received in step 17) to obtain the tsK which will be used to decrypt the Invoice, which if successful will indicate the User confirmation to process the Invoice.

    The FSP then charge the Users with the amount stated in the Invoice, and generate an InvoiceConfirmation, which is the Invoice concatenated with the FSPID and a status flag to indicate the statues of the charging, which can only be True (successful operation) or False (unsuccessful operation).

Once the SP receives the InvoiceConfirmation message from the FSP, it validate the message signature and then checks the StatusFlag, which if set to True, the SP will deliver the service to the User; an optional message can be sent to the FSP to confirm service delivery.

The scheme above supports both On-Phone and Off-Phone payment mechanisms. The protocol described above is for the Off-Phone payment mechanism. The On-Phone payment mechanism refers to the case when the user uses the Mobile Operator as a FSP by charging the user's phone bills.

The payment protocol will be exactly as in the Off-Phone case, with the main difference that the FSP will be the MO — the entity that contains the BSF.

4. Privacy consideration

This section examine User privacy in the proposed system, and identifying possible threats.

It is assumed that both the HAS and the MO are trusted entities and their system is very hard to compromise or brake.

- Privacy requirements and policies: This is achieved first by the HAS setting a general privacy policy guide lines, which all the SPs must comply with in their P3P privacy policy files. The main issue here is that it is not easy to technically to force the

SP to comply with its P3P policy, however protection can be put in place in a form of legal contract during the registration with the HAS.

- User Identity: the User will access various SPs with different IDs only known to the HAS (trusted entity), therefore for any one spoofing on the network will not be able to follow the User's activities even if more than one 'bad' SPs joins and share their knowledge about the system users. This is achieved because of the SSO system used by the HAS.
- Communication privacy: all communication messages are encrypted and can only be read by the intended receiver.
- Single point of failure: the IdP server part of the HAS is arguably the single point of failure in the system as it manages the Users identities, approve the policy files, and generate the security token for the system. If this is compromised then the system will collapse, however; we assumed that it was secure.

5. Conclusion

In this paper we have introduced a mobile eHealth scheme for the direct consumption of Web services by a Mobile Station; we described how SSO system such as Liberty Alliance ID-FF model can make use of the extended authentication services offered by 3GPP GAA to provide a *Healthcare Authentication Server* environment to mobile phone users and Healthcare Service Providers.

The protocol and the system structure described in this paper, were used to study and analyse Users authentication and privacy issues in Mobile Web Services eHealth. The results of the analysis showed that the integration of 3GPP GAA with SSO and P3P can improve the User privacy in eHealth systems. In addition the architecture is very flexible to include various payment mechanism and to cope with anonymous user scenario.

References

[1] The Platform for Privacy Preferences 1.0 Specification. Technical report, W3C, February 2002. W3C Recommendation.

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

[3] Access to network application functions using hypertext transfer protocol over transport layer security. Technical report, ETSI European Telecommunications Standards Institution, June 2005. UMTS, Generic Authentication Architecture.

[4] Generic bootstrapping architecture. Technical report, ETSI European Telecommunications Standards Institution, June 2005. UMTS, Generic Authentication Architecture.

[5] Interworking of Liberty Alliance ID-FF, ID-WSF and Generic Authentication Architecture. Technical report, 3GPP 3rd Generation Partnership Project, July 2005. 3GPP TR 33.980; Technical Specification Group Services and System Aspect, Release 4.

[6] Support for subsriber certificates. Technical report, ETSI European Telecommunications Standards Institution, June 2005. UMTS, Generic Authentication Architecture.

[7] Markus Hillenbrand, Joachim Gotze, Jochen Muller, and Paul Muller. A Single Sign-On Framework for Web-Services-based Distributed Applications. University of Kaiserslautern.

[8] John A. MacDonald, William G. Sirett, and Chris J. Mitchell. Overcoming channel bandwidth constraints in secure SIM applications. In *Security and Privacy in the Age of Ubiquitous Computing*. Springer Science and Business Media, 2005.