Hindawi Publishing Corporation
EURASIP Journal on Wireless Communications and Networking
Volume 2010, Article ID 249169, 12 pages
doi:10.1155/2010/249169

Research Article

SAM: Secure Access of Media Independent Information Service with User Anonymity

Guangsong Li,[1,2] Jianfeng Ma,[1] and Qi Jiang[1]

[1] *Ministry of Education Key Laboratory of Computer Networks and Information Security, Xidian University, Xi'an, Shaanxi 710071, China*
[2] *Department of Information Research, Zhengzhou Information Science and Technology Institute, Zhengzhou 450002, China*

Correspondence should be addressed to Guangsong Li, lgsday@gmail.com

Received 22 July 2010; Revised 11 October 2010; Accepted 19 October 2010

Academic Editor: Rodrigo C. De Lamare

Seamless handover across different access technologies is very important in the future wireless networks. To optimize vertical handover in heterogeneous networks, IEEE 802.21 standard defines Media Independent Handover (MIH) services. The MIH services can be a new target to attackers, which will be the main concern for equipment vendors and service providers. In this paper, we focus specifically on security of Media Independent Information Service (MIIS) and present a new access authentication scheme with user anonymity for MIIS. The protocol can be used to establish a secure channel between the mobile node and the information server. Security and performance of the protocol are also analyzed in this paper.

1. Introduction

Recent advances in wireless communication technologies have resulted in the evolution of various wireless networks, such as cellular network, wireless local area network, ad hoc network personal communication network, Communication in next generation networks will use multiple access technologies, creating a heterogeneous network environment [1]. Practically, a single network cannot cater for all different user needs or provide all services. Nowadays the availability of multimode mobile devices capable of connecting to different wireless technologies provides users with the possibility to switch their network interfaces to different types of networks.

Real-time multimedia services such as voice over IP and interactive streaming become more and more popular in current wireless networks, so ubiquitous roaming support for real-time multimedia traffic in an access independent manner becomes increasingly important. Seamless mobility can be achieved by enabling mobile terminals to conduct seamless handovers across diverse access networks, that is, seamlessly transfer and continue their ongoing sessions from one access network to another. Vertical handover in the heterogeneous networks is one of the major challenges for seamless mobility with ubiquitous connectivity, since each access network may have different mobility, quality of service,Fn and security requirements [2]. Moreover, real-time applications have stringent performance requirements on end-to-end delay and packet loss. In general, the vertical handover process can be divided into three main phases, namely, system discovery, handover decision, and handover execution [3]. During the system discovery phase, the mobile terminals have to determine which networks can be used and the services available in each network. These wireless networks may also advertise the supported data rates for different services. During the handover decision phase, the mobile device determines which network it should connect to. The decision may depend on various parameters or handover metrics including the available bandwidth, delay, jitter, access cost, transmit power, current battery status of the mobile device, and even the user's preferences. Finally, during the handover execution phase, the connections need to be rerouted from the existing network to the new network in a seamless manner. This phase also includes the authentication and authorization, and the transfer of user's context information.

In order to achieve seamless vertical handover in heterogeneous networks, many works have been carried out to address the issues of service continuity. Some of them made

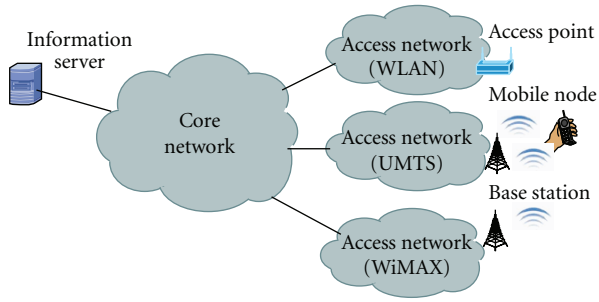


Figure 1: MIIS in heterogeneous networks.

efforts to methods about discovering neighbor networks and related information [4, 5]. Some of them focused on the issue of choosing the next network based on factors like bandwidth, cost, date rate, and so forth when the device is moving out of the current network [6–8]. Also, several approaches were published showing how to perform a fast authentication between different access technologies when handover-took place [9–11]. Apart from these, a number of works have also been carried out towards addressing other handover related issues [12–14].

Recent efforts by the IEEE 802.21 working group have designed a framework [15] to facilitate handover between heterogeneous networks by providing mobile users with information useful for making handover decisions. Examples of the information are the presence of neighboring networks, the type of their links, their characteristics, and the services supported. The heart of the framework is the Media Independent Handover Function (MIHF) which provides abstracted services to higher layers and vice versa by means of a unified interface. This is accomplished by defining a set of services, the Media Independent Handover (MIH) services, which consist of Media Independent Event Service (MIES), Media Independent Command Service (MICS), and Media Independent Information Service (MIIS). The MIES defines a solution for providing applications running above the data link layer with information about events triggered at the data link layer, such as the ones about the status of the link (link up, link down, etc.). The MICS introduces a set of commands that allows mobility functions running on the IP layer, or higher, to control the switching, scanning, and configuration functions of the data link layer. The MIIS specifies information about nearby networks useful for handover decisions and the query/response mechanism that allows mobile nodes to get that information. Users get that information from one or more information servers supporting MIH, as depicted in Figure 1. The Information Server (IS) may be located in the visited domains or in the users' home domain, that is, the domain of the service provider that holds information about the users' authentication and authorization profiles. The IEEE 802.21 working group is not trying to design a new mobility protocol, but to introduce a framework that supports the nodes involved in the mobility procedure to take handover decisions and to control the handover procedure. The IEEE 802.21 framework is complementary to existing mobility frameworks of wireless network.

As can be seen from Figure 1, MIH messages are exchanged over various wireless media between mobile nodes and access networks. Thus the MIH services can be a new target to attackers, which will be the main concern for equipment vendors and service providers [16]. Some typical threats about MIIS are listed below.

(i) *Identity Spoofing.* Attempting to gain access to information service by using a false identity.

(ii) *Tampering.* Unauthorized modification of information data exchanged.

(iii) *Information Disclosure.* Unwanted exposure of information data.

(iv) *Denial of Service.* The process of making information service unavailable to a user.

In addition, another important threat regarding the handover scenario is about user anonymity. It is desirable to hide the roaming user's identity and movements from eavesdroppers and even servers different from the home server he subscribed to. In heterogeneous wireless environments a roaming user needs to acquire neighbor network information from IS. If a user's identity is exposed to IS, the movements of the mobile user may be easily tracked by IS, since it knows the user's current location information and possible target of handover.

However, security mechanisms are not within the scope of the IEEE 802.21 standard. Security of MIH protocol currently relies on security of underlying transport protocols without a mechanism to authenticate peer MIH entities. This lack of authentication of peer MIH entities does not provide proper authorization for MIH services. Because IEEE 802.21 provides services that affect network resource, network cost, and user experience, MIH level security will be an important factor to network providers that want to deploy these MIH services in their networks. Nevertheless, there are very few security mechanisms for MIH services in the literature.

IEEE 802.21a task group was set up to address security issues of MIH services. The task of the group is [17]: (i) to reduce the latency during authentication and key establishment for handovers between heterogeneous access networks that support IEEE 802.21 (ii) to provide data integrity, confidentiality, replay protection, and data origin authentication to MIH protocol exchanges and enable authorization for MIH services. The technical requirements document [18] of the group describes usage scenarios and requirements for security signaling optimization during vertical handover and MIH protocol security. The scope of document [19] is to propose some solutions based on the requirements described in [18].

Won et al. proposed a new secure MIH message transport solution called MIHSec [20]. The idea of MIHSec is to utilize the Master Shared Key (MSK) generated by the L2 authentication procedure, for generating the MIH keys. MIHsec method though has a good performance for MIH message transportation, it introduces other issues. First, it

is closely integrated with L2 authentication, thus it is not media independent. Second, the MSK needs to be securely delivered to IS by AR (access router), which means a security association should be settled apriori between each AR and IS. So the scheme does not posses scalability. Finally, in MIHsec protocol, the AR that sends the MSK to the IS may know the key for MIH messages encryption, which degrades the level of security.

We note that user anonymity is not addressed in all above schemes. It is very important for a roaming user to keep his identity secret and movements untraceable. This paper proposes an anonymous protocol for Secure Access of MIIS, which is denoted as SAM for short. SAM not only has high level security but also obtains good performance. We give a rigorous formal analysis of its security using a modular approach. Some experiments and simulations about SAM are also done to evaluate performance of the protocol.

The rest of this paper is organized as follows. Section 2 is a quick review over some related works. In Section 3 we present our new approach in detail. Section 4 gives a formal security proof of our protocol under the CK model. Section 5 includes performance analysis. Finally, conclusions and future works are given in Section 6.

2. Related Works

2.1. 802.21a Task Group Proposals. Security is crucial for IEEE 802.21 standard to reach its market potential. Seamless mobility requires seamless security to make its applicability to government and enterprise networks. Thus 802.21a task group are making efforts to security mechanisms for IEEE 802.21 standard. In [19], proactive authentication techniques and MIH protocol level security mechanisms are elaborated.

Proactive authentication is a process by which an entity can perform a-priori network access authentication with a media independent authenticator and key holder (MIA-KH) that is serving a candidate network. The entity performs such authentication in anticipation of handover to the neighboring networks. Proactive authentication can be performed in two ways: (i) direct proactive authentication whereby the authentication signaling is transparent to the serving MIA-KH and (ii) indirect proactive authentication whereby the serving MIA-KH is aware of the authentication signaling. In each case either EAP (Extensible Authentication Protocol) [21] or ERP (EAP Reauthentication Protocol) [22] can be used as the authentication protocol.

As to MIH protocol security, two security frameworks were proposed: (i) MIH service access control applied through an authentication server and (ii) MIH service access control not applied through an authentication server.

In the first case (Figure 2), the access control may be applied by an access authentication through an EAP server or an AAA (Authentication, Authorization, and Accounting) server. Upon a successful authentication, the Mobile Node (MN) is authorized to access the MIH service through a Point of Service (PoS). The access authentication includes a key establishment procedure so that related keys are established between the MN and the Authentication Server (AS). The

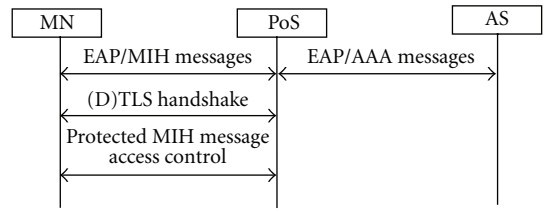


Figure 2: MIH security with access control.

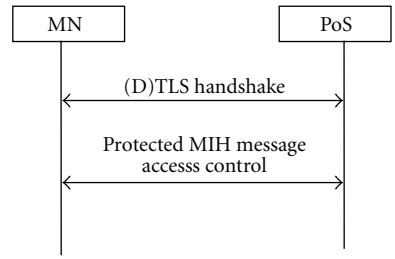


Figure 3: MIH security without access control.

method can provide MIH level protection independent to media and network access protection. Since MIH protection is end to end between the MN and the PoS, it is independent of the transport protocol for MIH. The use case is suitable for MIIS since the PoS for MIIS is more centralized. In the proposed approach, EAP framework is used over MIH protocol for carrying messages of MIH service authentication, where the PoS acts as an authenticator and also runs as an AAA client. TLS [23] or DTLS [24] is introduced to the authentication process, key establishment, and ciphering. (D)TLS handshake is carried out over MIH protocol, and a MIH SA (Security Association) is established between two MIHF peers. Once the MIH SA is established by the MIH protocol, there is no need to have MIH transport level security.

In the second case (Figure 3), the MIH service access control is not applied through any access controller. The mutual authentication may be based on a preshared key or a trusted third party like certificate authority (CA). The MN and the PoS will directly conduct a mutual authentication and key establishment protocol to setup a-MIH-specific SA. The use case allows pairwise MIH level mutual authentication and protection. This kind of MIH protection is independent of media and access technique. Since the MIH protection is end to end between the MN and the PoS, it does not rely on the transport protocol. The use case can treat MIIS, MIES, and MICS equally because no centralized server is involved.

2.2. Canetti-Krawczyk Model. A proof of security has become an essential statement for structural correctness of mutual authentication and key establishment protocols. Canetti and Krawczyk [25] proposed a model for provable security, which provided reusable building blocks for construction of new provably secure protocols. We refer to this model as the CK model in this paper. Here a description of the CK model is

given. Further details can be found in [25]. The CK model defines protocol principals who may simultaneously run multiple local copies of a message-driven protocol. Each local copy is called a session and has its own local state. Two sessions are matching if each session has the same session identifier and the purpose of each session is to establish a key between the particular two parties running the sessions. A session is expired if the session key agreed in the session has been erased from the session owner's memory.

A powerful adversary **A** attempts to break the protocol by interacting with the principals. In addition to controlling all communications between principals, the adversary is able to corrupt any principal, thereby learning all information in the memory of that principal (e.g., long-term keys, session states, and session keys). The adversary may impersonate a corrupted principal, although the corrupted principal itself is not activated again and produces no further output or messages. The adversary may also reveal internal session states or agreed session keys. The adversary must be efficient in the sense of being a probabilistic polynomial time algorithm. An unexposed session is the one such that neither it nor a matching session has had its internal state or agreed session key revealed. If the owner of the session or a matching session is corrupted, the corruption occurs after the key has expired at the corrupted party.

Two adversarial models are defined: the unauthenticated-links adversarial model (UM) and the authenticated-links adversarial model (AM). The only difference between them is the amount of control the adversary has over the communications channels between principals. The UM corresponds to the "real world" where the adversary completely controls the network in use and may modify or create messages from any party to any other party. The AM is a restricted version of the UM where the adversary may choose whether or not to deliver a message, but if a message is delivered, it must have been created by the specified sender and be delivered to the specified recipient without alteration. In addition, any such message may only be delivered once. In this way, authentication mechanisms can be separated from key agreement mechanisms by proving the key agreement secure in the AM, and then applying an authentication mechanism to the key agreement messages so that the overall protocol is secure in the UM.

To define the session key security of a key exchange (KE) protocol, the capability of the adversary is extended by allowing it to perform a test-session query. At any time during the game, **A** can issue a test-session query on a KE-session that is completed, unexpired, and unexposed. Let k be the corresponding session key. A coin $b_R \in \{0,1\}$ is tossed by the game simulator after receiving a test-session query from the adversary. If $b = 0$, k is returned to **A**; otherwise, a value chosen according to the distribution of session keys is returned to **A**. **A** can still carry out regular activities on this test-session after issuing the query but is not allowed to expose the test-session. However, the attacker is allowed to corrupt a partner to the test-session as soon as the test-session expires at that party. This captures the perfect forward secrecy property of a key exchange protocol. At the end of its run, **A** outputs a bit b' (as its guess for b).

Definition 1. A key exchange protocol π is called session key (SK)-secure in the AM if the following properties are satisfied for any AM-adversary **A**.

(1) If two uncorrupted parties complete matching sessions then they both output the same key;

(2) the probability that **A** guesses correctly the bit b is no more than 1/2 plus a negligible fraction about the security parameter.

The definition of SK-secure protocols in the UM is done analogously. By distinguishing between the AM and the UM, Canetti and Krawczyk allow for a modular approach to the design of SK-secure protocols. Protocols that are SK-secure in the AM can be converted into SK-secure protocols in the UM by applying an authenticator to it. An authenticator is a protocol translator C that takes as input a protocol π and outputs another protocol $\pi' = C(\pi)$, with the property that if π is SK-secure in the AM, then π' is SK-secure in the UM. Authenticators can be constructed by applying a message transmission (MT) authenticator to each of the messages of the input protocol. Canetti and Krawczyk [25] and Tin et al. [26] provided some examples of MT-authenticators.

3. Anonymous Access Authentication of MIIS

The MIIS message exchanges are critical to handover decision phase. Therefore the process of MIIS message exchanges has to be trusted. The mobile user needs both to protect itself from threats, and to provide the IS provable trust, in order that they can exchange the information securely. The user also wants to keep his identity secret and movements untracked from eavesdroppers, particularly the IS.

This section focuses on a new proposal SAM for anonymous access authentication of MIIS. The scenario we considered is that the access control for information service is applied through an access authentication controller, namely, an AS. The new solution has the advantages of lightweight computation, low communication cost, and easy implementation.

3.1. Network Model. We consider a wireless scenario as depicted in Figure 4. There are some application servers (S_1, S_2) in core network, which provide application services like, voice over IP, video conference, interactive games, and so forth. When an MN passes the network access authentication, it establishes connection with a Point of Attachment (PoA). The MN may request a kind of application service through a certain PoS. Frequently, some kind of authentication mechanism is necessary for application service to prevent invalid access without authority. In order to support mobile users to handover seamlessly between heterogeneous networks, an IS is deployed to provide information about neighbor networks for mobile users. We assume that all MNs should register with an AS and subscribe some services they needed at network initialization. When an MN registers to the AS, it generates a random number as the long-term shared key k_M with the MN. Presumably AS has a pair of public/private keys (g^x, x), which are generated by

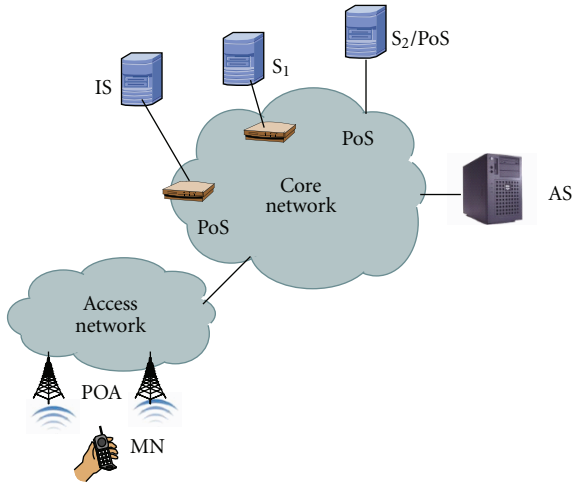
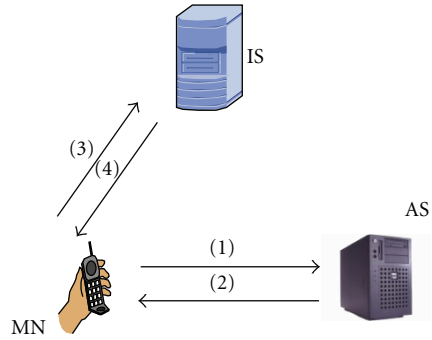


Figure 4: MIIS access control in the network.



(1) TReq, g^r, TID, ID_{IS}, $Enc_k(ID_{MN})$, ID_{AS}, t_M, MAC_M,
where TID = $h(g^r), k = prf(g^{rx})$
$MAC_M = h(k_M, TReq, g^r, TID, ID_{IS}, Enc_k(ID_{MN}), ID_{AS}, t_M)$
(2) TRes, TID, T, $Enc_{k_M}(TID, ID_{IS}, \sigma)$, ID_{AS}, t_A, MAC_A,
where $T = \{TID, ID_{IS}, Enc_{k_{AS\text{-}IS}}(TID, ID_{IS}, \sigma)\}$,
$MAC_A = h(k_M, TRes, TID, T, Enc_{k_M}(TID, ID_{IS}, \sigma), ID_{AS}, t_A)$
(3) SAReq, ID_{IS}, T, TID, t'_M, MAC'_M,
where $MAC'_M = h(\sigma, SAReq, ID_{IS}, T, TID, t'_M)$
(4) SARes, TID, ID_{IS}, t_{IS}, MAC_I,
where $MAC_I = h(\sigma, SARes, TID, ID_{IS}, t_I)$

Figure 5: Message flows of SAM.

itself. These keys are used to achieve user anonymity. In our network model, the attacker is able to corrupt any principal except for AS which is assumed beyond the attacker's control. We also assume that AS delivers k_M and public key g^x to MN using a mechanism outside of the proposed protocol, such as preloading these keys.

Here, MIIS is taken as a service at the application layer. It is assumed that MNs have no secure associations with application servers directly. In scenario where many application servers exist, Kerberos [27] is an efficient scheme for secure access of services because of its singlesign-on characteristic. We adopt a simplified version of Kerberos for easy deployment. Suppose that AS and TGS (Ticket Granting Server) are implemented by the same physical entity, which simplifies protocol design. We also assume that all application servers, (S_1, S_2, and so on, including IS) have shared some keys with the AS, respectively. For example, there is a long-term key $k_{AS\text{-}IS}$ shared between the IS and the AS for secure connection or authentication. Suppose that $prf()$ is a secure key derivation function, and $h()$ is a secure hash function. We assume that there is a time synchronization mechanism in the system. Below the new scheme is described in detail.

3.2. MIIS Access Authentication with User Anonymity. In order to handover seamlessly between heterogeneous networks while enjoying some real-time applications, each MN has to subscribe MIIS to AS when initializing. AS maintains an entry for each registered MN, which consists of the following items: ID_{MN}, k_M, service list. After an MN connects to the network, it should contact IS to get information about neighbor networks. Since the MN has no security associations with application servers (including IS), the access control of application services is applied through AS. To this end, the MN must obtain service ticket for IS. Then mutual authentication is performed between MN and IS using the service ticket. The message flows of SAM are depicted in Figure 5, in which flow (1) and (2) describe service ticket request and response flow and (3) to (4) describe mutual authentication between MN and IS.

(1) IS service ticket request (MN → AS). MN selects a random number r and computes $k = prf(g^{xr})$ as an anonymity key using public key g^x of AS. The identity ID_{MN} of MN is encrypted with k. A temporary identity TID is also computed using the equation: $TID = h(g^r)$. Then MN sends a service Ticket REQuest message (T_REQ) to AS for IS. The message content of T_REQ is as the following, {TReq, g^r, TID, ID_{IS}, $Enc_k(ID_{MN})$, ID_{AS}, t_M, MAC_M}, where TReq denotes the identifier of the request, ID_{IS} denotes the identifier of the information server, t_M is the timestamp of MN, and MAC_M is a message authentication code derived from the equation $MAC_M = h(k_M, TReq, g^r, TID, ID_{IS}, Enc_k(ID_{MN}), ID_{AS}, t_M)$.

(2) IS service ticket response (AS → MN). Upon receiving the T_REQ message from MN, AS extracts g^r then computes $k = prf(g^{rx})$ using g^r and its private key x. AS decrypts the ciphertext $Enc_k(ID_{MN})$, and gets the identity of MN. AS finds the item related to MN in its database, namely, the entry (ID_{MN}, k_M, service list). Then AS checks if the timestamp t_M is within some allowable range compared with its current time. If t_M is not valid, the request message is dropped because of staleness. Otherwise, AS computes the value $h(k_M, TReq, g^r, TID, ID_{IS}, Enc_k(ID_{MN}), ID_{AS}, t_M)$ using k_M. If the value matches with MAC_M in T_REQ, AS believes the message is really originated from MN. AS checks service list of MN to find whether it has subscribed service of IS. If MN has not subscribed the service of IS, AS will respond a reject message to MN. Otherwise,

a service ticket T will be generated for MN. AS chooses a random number σ as the service key used by MN and IS for secure connection. The format of service ticket is as follows: $T = \{\text{TID}, \text{ID}_{\text{IS}}, \text{Enc}_{k_{\text{AS-IS}}}(\text{TID}, \text{ID}_{\text{IS}}, \sigma)\}$, where $\text{Enc}_{k_{\text{AS-IS}}}(\text{TID}, \text{ID}_{\text{IS}}, \sigma)$ denotes the cipertext encrypted with the key $k_{\text{AS-IS}}$ shared between AS and IS.

AS generates a service Ticket RESponse (T_RES) message. The T_RES message consists of the following items $\{\text{TRes}, \text{TID}, T, \text{Enc}_{k_M}(\text{TID}, \text{ID}_{\text{IS}}, \sigma), \text{ID}_{\text{AS}}, t_A, \text{MAC}_A\}$, where TRes denotes identifier of the response, t_A is the timestamp of AS, and MAC_A is a message authentication code derived from the equation: $\text{MAC}_A = h(k_M, \text{TRes}, \text{TID}, T, \text{Enc}_{k_M}(\text{TID}, \text{ID}_{\text{IS}}, \sigma), \text{ID}_{\text{AS}}, t_A)$.

Afterwards, T_RES message is transmitted to MN by AS.

(3) IS service access request (MN → IS). When MN receives the T_RES message from AS, MN first validates t_A. If the result is positive, it calculates the value $h(k_M, \text{TRes}, \text{TID}, T, \text{Enc}_{k_M}(\text{TID}, \text{ID}_{\text{IS}}, \sigma), \text{ID}_{\text{AS}}, t_A)$ and compares the value with MAC_A in the T_RES message. If the two values are identical, MN believes the message is generated by AS. MN decrypts $\text{Enc}_{k_M}(\text{TID}, \text{ID}_{\text{IS}}, \sigma)$ to get the service key σ.

Now MN is able to contact with IS for MIIS. MN needs to send an information Service Access REQuest message (S_Acce_REQ) to IS. The message format of S_Acce_REQ is as the following: $\{\text{SAR}_{\text{eq}}, \text{ID}_{\text{IS}}, T, \text{TID}, t'_M, \text{MAC}'_M\}$, where SAReq denotes identifier of the request, T is the service ticket generated by AS, and t'_M is current timestamp of MN. MAC'_{M} is calculated using $\text{MAC}'_{\text{M}} = h(\sigma, \text{SAR}_{\text{eq}}, \text{ID}_{\text{IS}}, T, \text{TID}, t'_M)$.

(4) IS service access response (IS → MN). On receiving the IS_Acce_REQ message, IS validates t'_M and decrypts T using the key $k_{\text{AS-IS}}$ shared with AS to obtain the service key σ. It also gets the identifiers in the service ticket to determine whether the ticket is for TID and IS. Then IS computes $h(\sigma, \text{SAR}_{\text{eq}}, \text{ID}_{\text{IS}}, T, \text{TID}, t'_M)$ and compares it with the value of MAC'_M. If the two values are identical, IS believes the requestor is a valid client. IS then computes $k_s = prf(\sigma, \text{TID}, \text{ID}_{\text{IS}})$ as the service session key. IS generates an information Service Access RESponse message (S_Acce_RES) and sends to MN. The message has the following items: $\{\text{SARes}, \text{TID}, \text{ID}_{\text{IS}}, t_{\text{IS}}, \text{MAC}_I\}$, where SARes denotes the identifier of the response and $\text{MAC}_I = h(\sigma, \text{SARes}, \text{TID}, \text{ID}_{\text{IS}}, t_I)$.

After MN receives S_Acce_RES message, MN first validates t_I then computes $h(\sigma, \text{SARes}, \text{TID}, \text{ID}_{\text{IS}}, t_I)$ and compares it with the value of MAC'_M. If the two values are identical, IS passes the authentication to MN. MN computes $k_s = prf(\sigma, \text{TID}, \text{ID}_{\text{IS}})$ as the session key of information service. Afterwards, MN uses the service session key to secure access MIIS.

For accessing services other than the MIIS, the user needs to obtain the corresponding service ticket from AS. The user then sends an authentication request message directly to the application server which runs the authentication process as depicted in Figure 5. Based on the user credentials, the application server authenticates the user, which means that it checks user's service ticket and decides whether to grant access or not according to the authentication result. The application server and the user can use the shared secret key resulting from successful authentication to set up IPSec security at IP level or simply use the key to perform symmetric-cryptography based security at application level.

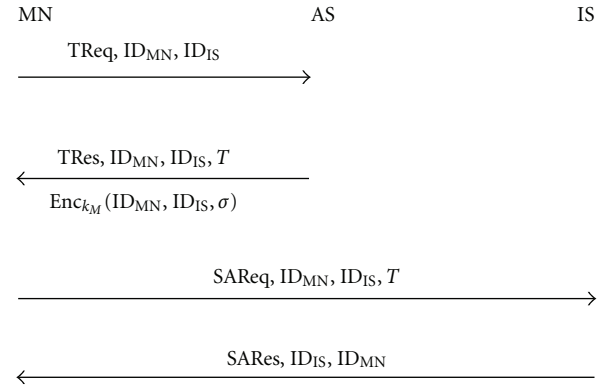


Figure 6: Flow chart of SKD protocol for MIIS access.

4. Formal Security Proof of SAM Protocol

In this section, we will give a rigorous proof for security of SAM under the CK model. We first present a basic SK-secure protocol in AM. Second, we extend it to achieve user anonymity. Third, we apply authenticators to the protocol to derive a protocol that is automatically secure in UM. Finally, we get our new protocol by reordering and reusing message components to optimize the resulting protocol.

4.1. Secure Key Distribution (SKD) Protocol in AM. We propose a key distribution protocol in AM where MN and IS rely on a trusted server AS for service key generation. This protocol uses only symmetric encryption. Figure 6 shows the flow chart of the protocol.

(1) IS service ticket request (MN → AS). MN sends a service ticket request message (T_REQ) to AS for IS. The message content of T_REQ is as $\{\text{TReq}, \text{ID}_{\text{MN}}, \text{ID}_{\text{IS}}\}$.

(2) IS service ticket response (AS → MN). Upon receiving the T_REQ message from MN, AS validates if MN and IS are the correct entities which have proper contractions with it. Then AS checks service list of MN to find whether MN has subscribed service of IS. If MN has subscribed the service of IS, AS chooses a random number σ as the service key used by MN and IS for secure connection. AS generates a service ticket as follows: $T = \{\text{TID}, \text{ID}_{\text{IS}}, \text{Enc}_{k_{\text{AS-IS}}}(\text{TID}, \text{ID}_{\text{IS}}, \sigma)\}$. Then AS sends to MN a service ticket response message (T_RES). The T_RES message consists of the following items: $\{\text{TRes}, \text{ID}_{\text{MN}}, \text{ID}_{\text{IS}}, T, \text{Enc}_{k_{\text{M}}}(\text{ID}_{\text{MN}}, \text{ID}_{\text{IS}}, \sigma)\}$.

(3) IS service access request (MN → IS). When MN receives the T_RES message from AS, MN needs to send an information Service Access REQuest message (S_Acce_REQ) to IS. The message format of S_Acce_REQ is as the following: $\{\text{SAReq}, \text{TID}, \text{ID}_{\text{IS}}, T\}$.

(4) IS service access response (IS → MN). On receiving the IS_Acce_REQ message, IS decrypts T using the key $k_{\text{AS-IS}}$ to obtain the identity of MN (which is confirmed by AS) and service key σ. IS then computes $k_s = prf(\sigma, \text{ID}_{\text{MN}}, \text{ID}_{\text{IS}})$ as the service session key. IS generates an information Service Access RESponse message (S_Acce_RES) and sends it to MN. The message has the following items: SARes, ID_{MN}, ID_{IS}.

After MN receives S_Acce_RES message, MN computes $k_s = prf(\sigma, \text{ID}_{\text{MN}}, \text{ID}_{\text{IS}})$ as the session key of information service. Afterwards, MN uses the service session key to secure access MIH information service.

Theorem 1. *The protocol SKD is SK-secure in the authenticated links model (AM) if the encryption algorithm Enc () used in SKD is a CCA-(chosen ciphertext attack-) secure symmetric encryption scheme.*

Proof sketch. It is easy to see that both parties MN and IS are in possession of the same session key upon the completion of the protocol execution, and therefore the protocol satisfies condition 1 of SK-security in Definition 1. So we concentrate on proving condition 2 of the SK-security.

Let **A** be an adversary against the protocol SKD. Let ε be the advantage of **A** indistinguishing between a session key and a random value of the same length. We show that if ε is nonnegligible, we can construct an algorithm **D** to break the encryption algorithm Enc (). **D** sets up a virtual scenario for the run of SKD and activates **A**. Virtual players include user MN, information server IS and authentication server AS. The scheduled operations are performed by **D** on behalf of all virtual players for SKD. We use x (resp., y and z) to denote the maximum number of MN (resp., IS and AS) that can be invoked. Let l denote the maximum number of sessions between the chosen parties. By running **A** as a subroutine, **D** can break the encryption algorithm Enc () with overall probability $1/2 + \varepsilon/lxyz$. The advantage $\varepsilon/lxyz$ is non-negligible. This contradicts our assumptions in Theorem 1.

4.2. Anonymous SKD Protocol in AM. Now we focus on extending the SKD protocol to achieve user anonymity. In [28], the authors proposed a general security framework to capture user anonymity and untraceability. They introduced a security definition for anonymity and untraceability in UM. Different to [28], we will define anonymity and untraceability in AM.

Let l be a system-wide security parameter. Let $M(l) = \{M1, \ldots, M_{Q1(l)}\}$ the set of mobile users in the system, $I(l) = \{I_1, \ldots, I_{Q2(l)}\}$ the set of information servers in the system, and $A(l) = \{A_1, \ldots, A_{Q3(l)}\}$ be the set of authentication servers in the system, where Q_1, Q_2, and Q_3 are some polynomials and M_t, I_u, and A_v are the corresponding identifiers of the parties, for $1 \le t \le Q_1(l)$, $1 \le u \le Q_2(l)$ and $1 \le v \le Q_3(l)$. First we depict a game of attacker similar to [28].

Anonymous Game: The game is carried out by a simulator **S** which runs an adversary **A**. It is based on the adversarial model AM.

(1) **S** sets up a system with users in $\text{M}(l)$, information servers in $\text{I}(l)$, and authentication servers in $\text{A}(l)$.

(2) **S** then runs **A** and answers **A**'s queries.

(3) **A** can execute the SKD protocol on any parties in the system by activating these parties and making queries.

(4) Among all the parties in the system, **A** picks two users M_t, $\text{M}_u \in \text{M}(l)$, an information server $\text{I} \in \text{I}(l)$, and an authentication server $\text{A} \in \text{A}(l)$, such that M_t, and M_u are the registration users of A.

(5) **A** sends a test query by providing M_u, M_v, I, and A.

(6) The simulator **S** simulates one SKD protocol run among M_u, I and A, and another one among M_v, I and A. **S** also updates the state information of each party due to the simulation. Then **S** tosses a coin b, $b_R \leftarrow \{0, 1\}$. If $b = 0$, the simulation transcript with M_u is returned to **A**, otherwise, that with M_v is returned to **A**.

(7) After receiving the response of the test query, **A** can still launch all the allowable attacks through queries and also activate parties for protocol executions as before.

(8) At the end of **A**'s run, it outputs a bit b' (as its guess for b).

A wins the game if (1) A, M_u, and M_v are uncorrupted, (2) for the one session above, **A** can only perform session-state reveal, session-key reveal,and session expiration queries to I. (3) A guesses correctly the bit b (i.e., outputs $b' = b$).

$$\text{Define } \text{Adv}\mathbf{A}(l) = \Pr[\mathbf{A} \text{ wins the game}] - \frac{1}{2}. \quad (1)$$

Definition 2. (user anonymity and untraceability) An SKD protocol provides user anonymity and untraceability if for sufficiently large security parameter l, $\text{Adv}\mathbf{A}(l)$ is negligible.

The formulation of Definition 2 is very powerful and can be shown to ensure both user anonymity and user untraceability required by a good SKD protocol. It guarantees that as long as the authentication server is uncorrupted, the adversary can neither tell the identity from the messages of one session nor link that session to another one.

Based on the secure SKD protocol (in AM), we now modify it so that it also provides user anonymity and untraceability. To provide user anonymity, the identity of the user should not be sent in clear. In addition, the identity should not be known to the information server according to the anonymity definition above. To do so, we use an identity hiding mechanism. Figure 7 depicts the message flows of the anonymous SKD protocol.

(1) IS service ticket request (MN → AS). MN selects a random number r computes $k = prf(g^{xr})$ as an anonymity key using the random number r and public key g^x of AS. The identity ID_{MN} of MN is encrypted with k. A temporary identity TID is also computed using the equation $\text{TID} = h(g^r)$. Then MN sends a service ticket request message (T_REQ) to AS for IS.

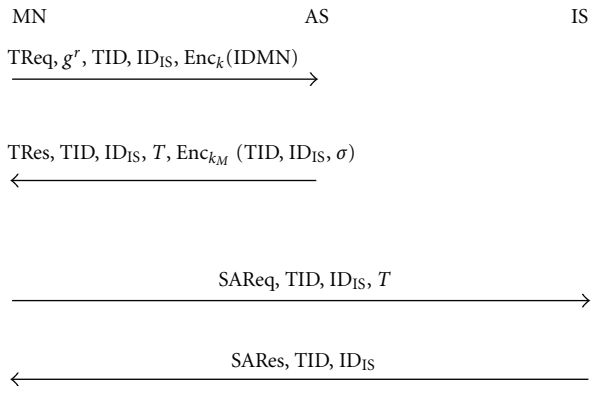


Figure 7: Flow chart of anonymous SKD protocol for MIIS access.

The message content of T_REQ is as the following: {TReq, g^r, TID, ID_{IS}, $Enc_k(ID_{MN})$}.

(2) IS service ticket response (AS → MN). Upon receiving the T_REQ message from MN, AS extracts g^r, then computes $k = prf(g^{rx})$ using g^r and its private key x. AS decrypts $Enc_k(ID_{MN})$, and gets identity of MN. AS finds the item related to MN in its database, namely, the entry (MN, k_M, service list). AS checks service list of MN to find whether it has subscribed service of IS. If MN has not subscribed the service of IS, AS will respond a reject message to MN. Otherwise, a service ticket T will be generated for MN. AS chooses a random number σas the service key used by MN and IS for secure connection. The format of service ticket is as follows: T = {TID, ID_{IS}, $Enc_{k_{AS\text{-}IS}}$(TID, ID_{IS}, σ)}. AS generates a service ticket response (T_RES) message. The T_RES message consists of the following items: {TRes, TID, ID_{IS}, T, Enc_{k_M} (TID, ID_{IS}, σ)}.

(3) IS service access request (MN → IS). When MN receives the T_RES message from AS, MN decrypts Enc_{k_M}(TID, ID_{IS}, σ) to get the service key σ. MN needs to send an information Service Access REQuest message (S_Acce_REQ) to IS. The format of the message is as: {SAReq, TID, ID_{IS}, T}.

(4) IS service access response (IS → MN). On receiving the IS_Acce_REQ message, IS decrypts T using the key $k_{AS\text{-}IS}$ to obtain the temporary identity of MN (which is confirmed by AS) and service key σ. IS then computes $k_s = prf(\sigma, TID, ID_{IS})$ as the service session key. IS generates an information Service Access RESponse message (S_Acce_RES) and sends to MN. The message has the following items: SARes, TID, ID_{IS}.

After MN receives S_Acce_RES message, MN computes $k_s = prf(\sigma, TID, ID_{IS})$ as the session key of information service. Afterwards, MN use, the service session key to secure access MIH information service.

Theorem 2. *If Enc () is CCA-secure and CDH (compute diffie-helleman) problem is difficult, the advantage Adv***A**(*l*) *that* **A** *wins the anonymity game is negligible.*

Proof. We prove it by contradiction. Namely, if the protocol is not anonymous, that is, if **A** wins the game with non-negligible advantage, Adv**A**(*l*), over random guess (which is half chance), we construct a distinguisher **D** to break Enc () or to solve CDH problem. □

We start by describing a game for the distinguisher **D**. First, **D** adaptively queries a decryption oracle with any ciphertext. Then **D** chooses two messages msg_0 and msg_1 and asks the game simulator for a ciphertext. The simulator randomly picks $b_R \leftarrow \{0, 1\}$ and gives **D** the ciphertext c such that $c = Enc_k(msg_b)$.

After receiving c, **D** adaptively queries the decryption oracle with any ciphertext except c. **D** is to output a value $b' \in \{0, 1\}$ as its guess for b. Now we construct **D** which simulates anonymous game. First, **D** sets up the system appropriately by creating a set M(l) of users, a set I(l) of information servers, and a set A(l) of authentication servers. It then initializes all the users in M(l) and information servers with randomly chosen symmetric keys from $\{0, 1\}^l$, and initializes all the authentication servers in A(l) with randomly chosen public key pairs for encryption. Afterwards, **D** randomly picks an authentication server A, and replaces its encryption public key and private key corresponding to g^x and x.

D runs **A** as a subroutine and answers all its queries and simulates all the responses of party activation due to protocol execution. If **A** picks M_u, M_v as two users, A as the authentication server, and I as the information server during the test query, **D** answers the query by providing the transcript of a protocol constructed as follows.

First, **D** randomly chooses a session ID s in $\{0,1\}^k$, and constructs two messages msg_0 and msg_1 as follows: $msg_0 = ID_{Mu}$, and $msg_1 = ID_{Mv}$.

D queries the CCA-security encryption oracle with msg_0 and msg_1. Suppose the CCA-security oracle returns g^r and a ciphertext c, which satisfies $c = Enc_k(msg_b)$, where $k = prf(g^{rx})$. Then, **D** constructs

message 1: TReq, g^r, TID, ID_{IS}, c

message 2: TRes, TID, ID_{IS}, T, Enc_{k_M}(TID, ID_{IS}, σ)

message 3: SAReq, TID, ID_{IS}, T

message 4: SARes, TID, ID_{IS}

The transcript returned by **D** to **A**, as the response for **A**'s test query is (message 1, message 2, message 3, message 4). **D** continues the game by answering all the queries made by **A** and simulating all the responses of party activation due to protocol execution. If **A** corrupts I, the simulator returns the long-term keys of I, and the internal state of I which includes the state information of session s, to **A**.

When **A** outputs a bit value b as its guess, **D** outputs b' and halts. If **A** does not pick A as the authentication server in his test query, **D** just randomly picks a value $b'_R \leftarrow \{0, 1\}$, outputs it and halts.

Analysis. Let E be the event that **A** picks A as the authentication server in its test query. Since **D** chooses A from A(l) in the game uniformly at random, $Pr[E] = 1/Q_3(l)$.

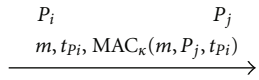


Figure 8: One-pass timestamp based MT-authenticator

Hence we have

$$\begin{aligned}\Pr[\mathbf{D} \text{ guesses } b \text{ correctly}] &= \left(\frac{1}{2} + \mathrm{Adv}\mathbf{A}\ (l)\right)\Pr[E] \\ &\quad + \frac{1}{2}(1 - \Pr[E]) \\ &= \frac{1}{2} + \mathrm{Adv}\mathbf{A}(l)/Q_3(l),\end{aligned} \tag{2}$$

which is non-negligible over random guess.

D may win the game by the following means.

(1) **D** analyses CCA-secure encryption scheme with the help of adaptive query to plaintext of any chosen ciphertext except to the challenge c.

(2) **D** computes the key $k = prf(g^{rx})$ with the knowledge of g^r and g^x, then decrypt the ciphertext c to get $\mathrm{msg_b}$;

(3) **D** guesses b directly with correct probability 1/2.

Assume probability of case (1) is $\mathrm{Adv_{Enc}}$ and probability of case (2) is $\mathrm{Adv_{CDH}}$.

Thus, $\mathrm{Adv_{Enc}} + \mathrm{Adv_{CDH}} \geq \Pr[\mathbf{D} \text{ guesses b correctly}] - 1/2 = \mathrm{Adv}_{\mathbf{A}}(l)/Q_3(l)$.

If $\mathrm{Adv}_{\mathbf{A}}(l)$ is non-negligible, at least one of $\mathrm{Adv_{Enc}}$ and $\mathrm{Adv_{CDH}}$ is non-negligible. So we have constructed a distinguisher **D** to break Enc () or to solve CDH problem.

4.3. Anonymous SKD Protocol in UM. Now we come to the anonymous secure key distribution protocol in UM. Since the adversary can forge and modify any message, the identities of the user, the information server, and the authentication server all should be authenticated in the scenario.

An anonymous SKD protocol in UM can be derived by applying certain MT-authenticators to the SKD protocol in AM according to the CK approach [25]. Here we apply the one-pass timestamp based-MT-authenticator to the message flows of the protocol depicted in Figure 7.

The one-pass timestamp based MT-authenticator is depicted as Figure 8. Though the authenticator is very simple, it is widely used in synchronized system. It helps simplify the authentication procedures and improve the protocol efficiency.

Suppose that a party P_i shares a random key κ with another party P_j. There exists a time synchronization mechanism between P_i and P_j. The one-pass timestamp based MT-authenticator λ_t proceeds as follow:

(i) Whenever P_i wants to send a message m to P_j, P_i extracts its timestamp t_{Pi}, sends m, t_{Pi}, $\mathrm{MAC}_\kappa(m, P_j, t_{Pi})$ to P_j, where MAC is a message authentication function, and adds a message "P_i sent m to P_j" to P_i's local output.

Table 1: Cryptographic operations and computational costs.

Computation operations	Notation	Time (ms)
Certificate validation	T_{CV}	10.5
DH key generation	T_{DH}	14.2
Random number generation	T_{RG}	0.09
Hash value computation	T_{HC}	0.03
Key derivation	T_{KD}	0.03
Symmetric encryption	T_{SE}	0.12
Symmetric decryption	T_{SD}	0.12

(ii) Upon receiving m, t_{Pi}, $\mathrm{MAC}_\kappa(m, P_j, t_{Pi})$, P_j verifies that the $\mathrm{MAC}_\kappa(m, P_j, t_{Pi})$ is correct and t_{Pi} is within allowable range. If all verifications are correct, P_j outputs "P_j received m from P_i."

After deriving the anonymous SKD protocol in UM, an optimization [26] of message flows can be applied. As a result, we obtain a UM anonymous SK-secure protocol SAM in Figure 5, which provides secure access for information service with user anonymity.

5. Performance Analysis

Protocol performance has become an increasingly important concern in wireless computing and networking environments. It is always desirable to make an authentication protocol more efficient. Our protocol may be quite efficient, since it relies mainly on symmetric key operations and a few rounds of message exchanges during access authentication process. The computational cost of our protocol is very reasonable, especially for the mobile node. The computation operations in our protocol are negligible compared to any strong public-key authentication. In the proposal of 802.21a task group [19], EAP framework is suggested to fulfill mutual authentication between peers for the centralized MIH service. EAP-TLS [29] is a typical and widely applied authentication protocol in EAP protocol family. We take it as an example for comparison.

To evaluate our protocol and 802.21a proposal, we implemented all cryptographic operations required in the two schemes using the Crypto++ Library (version 5.6.1) [30]. The cryptographic experiments were executed on a laptop with PIII 1.6 GHz CPU and 128 MB RAM. The results are listed in Table 1, where SHA-1, AES, and RSA are used for analysis. The computational costs required by MN, AS, and IS (or PoS) are given in Table 2. Compared with SAM, 802.21a proposal is a rather complex and high-cost process because of using public key certificates. That method adds too much load to entities involved (consuming much time and energy). According to Table 2, we can conclude that the computational cost of MN, AS and IS can be reduced nearly by 41.7%, 40.8% and 30.0% in SAM, respectively.

As to communication performance, in the first phase of SAM (service ticket request), only a 2-way handshake is executed between MN and AS. It fulfils tasks of data origin authentication and service ticket distribution. In the second

Table 2: Computational costs in 802.21a and SAM.

	802.21a	SAM
MN	$T_{CV} + T_{DH} + T_{RG} + 2T_{HC} + 2T_{KD} = 24.91$	$T_{DH} + T_{RG} + 4T_{HC} + T_{KD} + T_{SE} + T_{SD} = 14.68$
AS	$T_{CV} + T_{DH} + T_{KD} + T_{SE} = 24.85$	$T_{DH} + T_{RG} + 2T_{HC} + 2T_{SE} + T_{SD} = 14.71$
IS	$T_{RG} + 2T_{HC} + T_{KD} + T_{SD} = 0.3$	$2T_{HC} + T_{KD} + T_{SD} = 0.21$
Total time	50.06	29.60

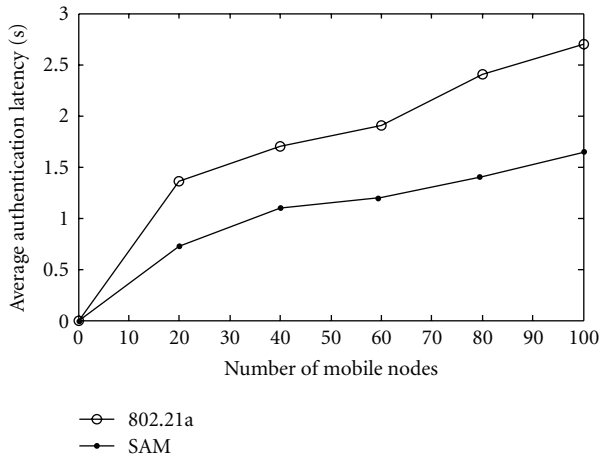


Figure 9: Comparison about average authentication latency.

Table 3: Simulation parameters.

WLAN area	300 m*300 m
The number of AP	10
Coverage of AP	100 m
The number of MNs	20~100
The number of MIIS request for each MN	10
Simulation time	5 minutes

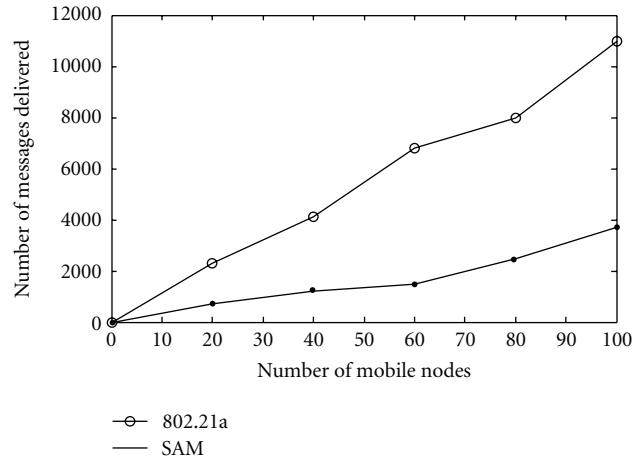


Figure 10: Comparison about number of messages delivered

phase (information service access request), mutual authentication between MN and IS is also carried out through a 2-way handshake procedure. Nevertheless in 802.21a proposal, a full EAP-TLS procedure requires 8 message flows between MN and AS for their mutual authentication, afterwards it has to perform mutual authentication between PoS of IS, and MN (at least 3 message flows). The whole process of 802.21a needs so many message flows that it consumes too much bandwidth and time. Thus our protocol performs better than the proposal of 802.21a task group.

We carried out some simulation experiments of SAM and 802.21a proposal using OPNET 10.5 [31] to verify analysis above. For simplicity, only a WLAN was used as the access network in the topology, and one AS and one IS were deployed, where the two servers were both connected to the Internet as in Figure 4. The simulations run with 20~100 MNs and 10 APs uniformly distributed in the WLAN area for 5 minutes of simulation time. For the MIIS authentication request pattern, each MN made 10 requests randomly distributed over the whole simulation period. The simulation parameters are listed in Table 3. Here we mainly focus on the measurements of average authentication latency and the number of messages delivered in the network.

Figure 9 shows the average authentication latency of the two schemes as the number of MNs changes. We can see that the average authentication latency of SAM and 802.21a both become larger as the number of MNs increases. The reason is that the number of packets generated in the network increases as the number of MNs increases, which makes packets collision and retransmission happen more often. The average authentication latency obtained using SAM is about 60% to that obtained using 802.21a in all scenarios. This suggests that SAM is highly effective in authentication latency. Figure 10 shows the changes of the number of messages delivered in the network when the number of MNs changes. As we can see from the results, the number of messages delivered of 802.21a increases sharply while that of SAM increases smoothly as the number of MNs increases. The number of messages delivered of SAM is about 30% to that of 802.21a in all scenarios.

The simulation results indicate that SAM has advantages in communication performance compared with 802.21a.

6. Conclusions and Future Works

The IEEE 802.21 standard aims at optimizing handovers among heterogeneous wireless networks. In this paper, we propose an anonymous access authentication protocol for MIIS defined in the 802.21 standard. We adopt a modified version of Kerberos featuring of user anonymity in service ticket distribution and service access authentication. The security and performance analyses show that the proposed

scheme has good characteristics. In fact, our work can be applied to offer integrated authentication and authorization functionalities for any type of application service.

By ensuring a robust access authentication for MIIS, our scheme can be a step forward from best-effort to support seamlessly mobility in wireless world. Now we are making an effort to put up a real testbed to evaluate performance of our protocol. There are also some interesting works deserving considerations. The information server may not have a previously established security association with the mobile user's authentication server, then how to implement secure access for MIIS at this scenario? The mobile user and the information server may belong to different security domains, thus cross-domain authentication schemes ought to be established. In the future heterogeneous networks, there may exist several information servers deployed by different providers; the mobile user needs an efficient method to choose a more trusted one from a set of information servers.

Acknowledgments

The authors would like to thank the anonymous reviewers and the editor for their constructive comments that have helped them to improve this paper. This work is supported by the National Natural Science Foundation of China (60872041, 60633020, 60702059, 60803154), the National High Technology Research and Development Program of China (2007AA01Z429, 2009AA01Z417), and the China Postdoctoral Science Foundation (20100471604).

References

[1] N. Nasser, A. Hasswa, and H. Hassanein, "Handoffs in fourth generation heterogeneous networks," *IEEE Communications Magazine*, vol. 44, no. 10, pp. 96–103, 2006.

[2] G. Karopoulos, G. Kambourakis, and S. Gritzalis, "Survey of secure handoff optimization schemes for multimedia services over all-IP wireless heterogeneous networks," *IEEE Communications Surveys and Tutorials*, vol. 9, no. 3, pp. 18–28, 2007.

[3] J. McNair and F. Zhu, "Vertical handoffs in fourth-generation multinetwork environments," *IEEE Wireless Communications*, vol. 11, no. 3, pp. 8–15, 2004.

[4] W.-I. Kim, B.-J. Lee, Y.-S. Shin, and Y.-J. Kim, "Battery efficient wireless system discovery scheme for inter-system handover," in *Proceedings of the 25th IASTED International Conference on Parallel and Distributed Computing and Systems (PDCN '07)*, pp. 28–32, ACTA Press, Innsbruck, Austria, 2007.

[5] F. Siddiqui and S. Zeadally, "An efficient wireless network discovery scheme for heterogeneous access environments," *International Journal of Pervasive Computing and Communications*, vol. 4, no. 1, pp. 50–60, 2008.

[6] E. Stevens-Navarro, Y. Lin, and V. W. S. Wong, "An MDP-based vertical handoff decision algorithm for heterogeneous wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 57, no. 2, pp. 1243–1254, 2008.

[7] Y. Nkansah-Gyekye and J. I. Agbinya, "A vertical handoff decision algorithm for next generation wireless networks," in *Proceedings of the 3rd International Conference on Broadband Communications, Informatics and Biomedical Applications (BroadCom '08)*, pp. 358–364, Gauteng, South Africa, 2008.

[8] S. K. Lee, K. Sriram, K. Kim, Y. H. Kim, and N. Golmie, "Vertical handoff decision algorithms for providing optimized performance in heterogeneous wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 2, pp. 865–881, 2009.

[9] D. Nikitopoulos, N. Papaoulakis, A. Trakos, A. Giamas, E. Sykas, and M. Theologou, "Authentication platform for seamless handover in heterogeneous environments," in *Proceedings of the Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services (ICAS/ICNS '05)*, p. 36, Papeete, Tahiti, October 2005.

[10] S. C.-H. Huang, H. Zhu, and W. Zhang, "SAP: seamless authentication protocol for vertical handoff in heterogeneous wireless networks," in *Proceedings of the 3rd International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine '06)*, vol. 191 of *ACM International Conference Proceeding Series*, pp. 231–241, ACM, Waterloo, ON, Canada, 2006.

[11] A. A. Shidhani and V. C. M. Leung, "Reducing reauthentication delays during UMTS-WLAN vertical handovers," in *Proceedings of the IEEE 19th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC '08)*, pp. 1–5, Cannes, France, September 2008.

[12] R. G. Garroppo, S. Giordano, S. Lucetti, G. Risi, and L. Tavanti, "An experimental cross-layer approach to improve the vertical handover procedure in heterogeneous wireless networks," *Journal of Communications Software and Systems*, vol. 2, no. 1, pp. 40–50, 2006.

[13] N. Shenoy and R. Montalvo, "A framework for seamless roaming across cellular and wireless local area networks," *IEEE Wireless Communications*, vol. 12, no. 3, pp. 50–57, 2005.

[14] H. Kwon, K.-Y. Cheon, and A. Park, "Analysis of WLAN to UMTS handover," in *Proceedings of the IEEE 66th Vehicular Technology Conference (VTC '07)*, pp. 184–188, Baltimore, Md, USA, October 2007.

[15] IEEE 802.21 standard, Media Independent Handover Services, 2009.

[16] Y. Ohba, "Five criteria for security extensions to media independent handover services," http://www.ieee802.org/21/802_21a_5C.pdf.

[17] 802.21a PAR, "Amendment for security extensions to media independent handover services and protocol," http://www.ieee802.org/21/802_21a_Par.pdf.

[18] S. Das, M. Meylemans, Y. Ohba et al., "Security SG," Tech. Rep. IEEE 802.21, 2008, https://mentor.ieee.org/802.21/documents.

[19] S. Das, A. Dutta, and T. Kodama, "Proactive authentication and MIH security," 2009, https://mentor.ieee.org/802.21/documents.

[20] J. Won, M. Vadapalli, C. Cho, and V. C. M. Leung, "Secure media independent handover message transport in heterogeneous networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2009, Article ID 716480, 15 pages, 2009.

[21] B. Aboba, D. Simon, and P. Eronen, "Extensible Authentication Protocol (EAP) key management framework," RFC 5247, 2008.

[22] V. Narayan and L. Dondeti, "EAP extensions for EAP reauthentication protocol (ERP)," RFC 5296, 2008.

[23] T. Dierks and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2," RFC 5246, 2008.

[24] E. Rescorla and N. Modadugu, "Datagram transport layer security," RFC 4347, 2006.

[25] R. Canetti and H. Krawczyk, "Analysis of key-exchange protocols and their use for building secure channels," in *Proceedings of the Advances in Cryptology—Eurocrypt*, vol. 2045 of *Lecture Notes in Computer Science*, pp. 453–474, Springer, 2001.
[26] Y. S. T. Tin, C. Boyd, and J. G. Nieto, "Provably secure key exchange: an engineering approach," in *Proceedings of the Australasian Information Security Workshop*, pp. 97–104, 2003.
[27] C. Neuman, T. Yu, S. Hartman, and K. Raeburn, "The Kerberos network authentication service (V5)," RFC 4120, 2005.
[28] G. Yang, D. S. Wong, and X. Deng, "Formal security definition and efficient construction for roaming with a privacy-preserving extension," *Journal of Universal Computer Science*, vol. 14, no. 3, pp. 441–462, 2008.
[29] D. Simon, B. Aboba, and R. Hurst, "The EAP TLS authentication protocol," RFC 5216, 2008.
[30] Crypto++ Library, http://www.cryptopp.com/.
[31] OPNET, http://www.opnet.com/.