

A METHOD FOR AUTHENTICATION AND KEY EXCHANGE FOR SEAMLESS INTER-DOMAIN HANDOVERS

Rene Soltwisch, Xiaoming Fu, Dieter Hogrefe
University of Göttingen, Institute for Informatics
Lotzestrasse 16-18,
37083 Göttingen, Germany
{soltwisch, fu, hogrefe}@cs.uni-goettingen.de

Sathya Narayanan
Panasonic Technologies
2 Research Way
Princeton, NJ 08540, USA
sathya@research.panasonic.com

Abstract - With the rapid growth of the Internet and mobile wireless technologies, an ever-increasing requirement on securing services between mobile users and access networks has become especially important. When¹ a user roams into a foreign network, in addition to data confidentiality, mutual authentication between the user and the provider is also a vital issue. These concerns and the desire to stay seamlessly connected lead to the demand of fast authentication and key establishment mechanisms, which are particularly difficult in inter-domain handover scenarios. In this paper, we introduce a novel mechanism to provide a simple but effective method, which forwards the key from the previous access router to the new access router that the mobile node attaches to. With this mechanism, trust relationship can be re-established even if the access routers do not trust each other in such an inter-domain scenario. Compared with the classical authentication method used in GSM and a recently proposed EAP-based secure key exchange protocol, our approach shows advantages of faster key exchange and authentication with only minimal message exchange in the wireless link.

Keywords – Context Transfer Protocol, Authentication, Internet Key Exchange, Inter-Domain, Seamless Handover

1. INTRODUCTION

In recent years the Internet grows rapidly and is expected to support a large portion of node mobility. Meanwhile, independent of access technologies such as UMTS, 802.11x, or ultra-wideband, future mobile services are expected to be built on all-IP technologies. In such environments, a mobile node (MN) can connect to the Internet whenever and wherever possible, no matter how far from their home network. Moreover, while MNs roam between different providers, they need to set up the required service parameters such as QoS for Internet telephony as well as trust relationship in the new access network.

Trust relationship in roaming scenarios can be identified into two main aspects: (1) Authentication of the end-user or terminal (MN), e.g., by an Authentication, Authorization, and Accounting (AAA) server. This server is usually located in the home network of the MN and therefore called Home AAA server (HAAA), in contrast to the Foreign AAA server (FAAA) in the foreign network. (2) Data confidentiality between the MN and the access router (AR) it attaches to. Commonly, this is achieved by symmetric encryption mechanisms based on temporary per-session keys unique for each node. In this paper, we focus on key distribution and node authentication, which form an important building block for modern security mechanisms.

Some mechanisms have been introduced to authenticate the MN and establish the key between the MN and its AR, using two possible ways.

One approach is to transport the key from the home network to the access network like in the GSM authentication mechanism [1]. In IP host roaming scenarios, e.g., when an MN moves to 802.11x hotspots, a generic Extensible Authentication Protocol (EAP) [2] can be used together with its methods (e.g., [3, 4, 5]) for transferring credentials and providing authentication. However, key distribution can cause long latency if the provider has to prove the authenticity through the HAAA server, which can be physically located far away.

Another possibility is to forward the key from the previous access router (pAR) to the new access router (nAR), based on a Context Transfer Protocol (CTP) [6], which has been proposed by the IETF. CTP provides seamless handovers from the pAR to the nAR because it is usually faster to forward settings and credentials from the pAR than contacting the MN's home network. An assumption in this approach is that trust relationship and security association between both ARs exists before performing a handover. This assumption is acceptable for scenarios where handovers take place within a single administrative domain (intra-domain handover) but commonly does not hold in

This work is partly supported by a research collaboration between the University of Göttingen and Panasonic Inc.

case the MN traverses between different administrative domains (inter-domain handover).

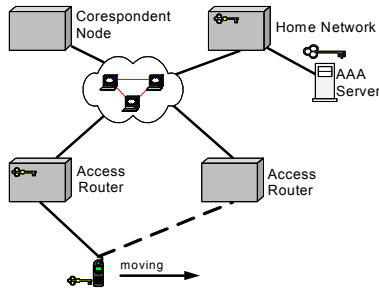


Fig. 1. A roaming scenario

Fig. 1 illustrates an MN moving to a foreign access network where neither keys nor any other service parameters for the MN exist. On the one hand, the key- and parameter-setup should occur as fast as possible in order to provide seamless handovers, and in addition the parameter-setup should occur exclusively for authorized MN and obviously in a secure way, since these parameters enable the MN to utilize the network provider’s services. On the other hand, from the providers’ point of view, it has to be ensured that the user gets the necessary permission before establishing services and therefore a trust relationship with the MN needs to be established.

In this paper we present a new mechanism to establish all necessary trust relationships and the required security associations (SA) in order to improve the CTP inter-domain handover capacity. This new method, which we denoted as Inter-Domain-Key-Exchange (IDKE) in this paper, intends to satisfy the requirements to support secure and seamless handovers. IDKE is independent of any specific access technology since it works on top of the transport layer.

The next section shortly reviews two existing mechanisms that are important for our discussion, namely the Internet Key Exchange Protocol (IKE) [7] and CTP. Details about IDKE are presented in Section III. In Section IV we analyze our approach and compare it with two other mechanisms for mobile user authentication, namely the GSM authentication approach and the W-SKE approach presented by Salgarelli, Buddhikot, *et al.* [8]. Finally, conclusions and future work are discussed.

2. BACKGROUND

In this section we shortly describe related mechanisms which IDKE will be based on.

A. Internet Key Exchange

The Internet Key Exchange (IKE) protocol is a mechanism to dynamically exchange keys between two parties, e.g., to establish an SA between them. IKE uses UDP for its

message exchanges. The amount of messages exchanged depends on the method that is used: for example 6 messages are needed in IKE main mode and 3 messages in IKE aggressive mode. In either case, IKE phase 1 is used to install its own SAs in order to protect phase 2. Since performance (e.g., key establishment delay) is very important in wireless mobile scenarios, the aggressive mode is recommended in this paper. However, our approach can also work with other modes of IKE.

B. Context Transfer Protocol

The Context Transfer Protocol (CTP) aims to enhance IP handover performance. When an MN moves to a new access network, it needs to continue certain transport-related services or services that have already been established at the previous subnet. Such services are called “context transfer candidate services”. Examples are states used in header compression, QoS reservations, AAA profile, IPsec, or firewall configurations. Re-establishing these services at the new access network will require a considerable amount of time for the protocol exchanges and as a result time-sensitive real-time traffic will suffer during this time. Alternatively, context transfer candidate services can be transferred, for example, from the pAR to the nAR so that the services will be re-established quickly. It is one means that enables the seamless IP handover operation of application streams and could reduce susceptibility to errors. Furthermore, service re-initiation to and from the MN will be avoided hence wireless bandwidth efficiency is conserved. CTP is assumed to be typically used in intra-domain handover scenarios.

3. THE IDKE APPROACH

In this section we present our mechanism called Inter Domain Key Exchange (IDKE) to address the authentication and key exchange issues in inter-domain seamless IP handover scenarios. First, we study the requirements for such mechanisms, and then we describe our approach with detailed discussions.

A. Requirements

We can divide requirements for our problem into two categories: (1) general requirements for wireless mobile environments, such as performance and reliability, and (2) security-specific requirements, for example, avoiding new security holes and providing robustness against a variety of attacks. They are elaborated as follows:

1) General Requirements

- Performance: A minimum of messages should be exchanged at low bandwidth links. In mobile wireless scenarios the wireless link is usually the bottleneck. Therefore an authentication and key exchange mechanism should try to avoid

unnecessary message exchanges over the wireless link.

- Easy implementability: Existing standards and already implemented protocols should be reused if possible.
- Statelessness: Installing states in participating nodes should be avoided whenever possible and if unavoidable should be installed as late as possible.

2) Security Requirements

Obviously security mechanisms should guarantee that only the corresponding parties have knowledge about the key. Further requirements on keying material are as follows:

- Key strength: The keying material should be cryptographically strong. Furthermore, keys and user IDs should be fresh, random and unique.
- Session-key establishment: As the key exchange mechanism only matters once after an MN moves to a new access networks, only temporary session security associations should be established for setting up trust relationship between the MN and the access network. Thus the mechanism should generate per-session keys for each user, which we call Session Master Secrets (K_{sms}).
- Prevent session hijacking: The mechanism should prevent users from seizing control of a communication session previously established by another user.
- DoS attack prevention: The mechanism should not be vulnerable against Denial of Service (DoS) attacks [9]. For example, setting states and performing high computation should be avoided when a request is issued by some untrusted node and, if unavoidable, done as late as possible.

B. IDKE Overview

As described above, the main goal of IDKE is to establish a trust relationship and a shared key between the MN and the nAR for encryption and authentication on the wireless link in order to provide seamless IP handovers. The basic idea of IDKE is to avoid re-establishing keys from scratch but to forward them from the pAR, thus reducing the delay and enabling seamless handovers. We reuse CTP for this purpose.

According to CTP, a secure channel and a trusted correspondent are required before using CTP for key exchange. However, this generally is not the case in inter-domain scenarios. Therefore, we introduce a simple but effective mechanism (herein called IDKE) by allowing CTP to securely forward the key and utilizing IKE to establish the required SA between the ARs. The mechanism consists of four steps: (1) trust establishment between the ARs, (2)

transfer-key negotiation required for a secure channel establishment, (3) transfer of MN's credentials (session-key) between the ARs, and (4) the acknowledgement and home registration process. Step 2 and 3 are basically well-known protocols, while step 1 forms the core of IDKE. First we give an overview of all three steps and then we examine the most important step separately (Fig. 3c illustrates the message flow).

(1) In the first step we need to trigger the ARs to set up the required security associations prior to performing CTP and thus the key forwarding. For that purpose, once an MN gets IP connectivity from the nAR, it sends an IDKE_initiation message to the nAR, which includes a token (see Section III.C for detailed discussion) and a request for the pAR. Upon receipt of this message, the nAR assembles a new IDKE_forward message based on the MN's request but adds its own token before sending it to the pAR. The pAR confirms the validity of the token and, according to the result, sends an IDKE_acknowledge message to the nAR to trigger the SA establishment phase.

(2) The second step is to establish an SA between pAR and nAR by utilizing a key exchange between the ARs. Therefore we set up a secret key shared by pAR and nAR called $K_{\text{par,nar}}$. It is important that this key is only known to the two ARs. Therefore the ARs need to create their security association independently of the MN based on common key negotiation algorithms such as Diffie Hellmann. Afterwards, the SA and thus the secure channel between the trusted parties can be reused between the ARs. This can provide performance benefits since there might be some additional nodes roaming between the same <pAR, nAR> pair. For IDKE we decided to use IKE as illustrated in step 4-6 of Fig. 2 when IKE is used in aggressive mode. From now on a secure channel between pAR and nAR can be assumed. All data is encrypted based on the symmetric key $K_{\text{par,nar}}$. For that purposes IPsec [10] in tunnel mode can be used to provide data confidentiality and authentication.

(3) The session-key K_{sms} , as well as other session-related parameters such as QoS reservations if necessary, are transferred from the pAR to the nAR (message 7-8 in Fig. 2). This can also improve the performance because we avoid overloading the bottleneck – the wireless link – for transferring potentially large sizes of keying- or other service-parameters.

(4) The nAR creates a new key $K_{\text{sms-new}}$ by performing a logical exclusive OR operation on a randomly created value K_{ran} and the old key K_{sms} . The new value of $K_{\text{sms-new}}$ is computed as $K_{\text{sms-new}} := (K_{\text{sms}} \text{ XOR } K_{\text{ran}})$. The random value is attached to the IDKE_tmp_key_established message which is sent to the MN (message 9) and provides a temporary key for the MN and therefore temporary access to the nARs services. According to the logical operation the MN also computes the new session-key $K_{\text{sms-new}}$ by performing the logical operation. In parallel the nAR sends a message to the MN's HAAA server in order to inform the

MN's home about the AR change (message A). The message to the home network contains the random value K_{ran} and is signed by the old session-key K_{sms} .

The authorization is temporary since the nAR is still expecting the HAAA server to send an acknowledgement (message B) to confirm the key change and the nAR has to decide how long this key should be valid and may request the MN to re-authenticate after some timeout. Another possibility is that the HAAA server might neglect the new key and request the MN to perform authentication from scratch which is the commonly used challenge-response procedure. Once the HAAA server acknowledges the new key, an IDKE_accomplished is sent to the MN (message 10).

After the registration procedure as described above, the MN can perform its binding procedure by sending a binding update to its home network. These steps are not considered here in more detail since this process has to be performed anyway for mobility management purposes.

C. New IDKE Messages

In this section we explore the mechanism in more detail. The focus is on the new messages introduced in the first step of IDKE, since the others are covered by already well-known protocols. As mentioned above the first of the three steps consists of three messages.

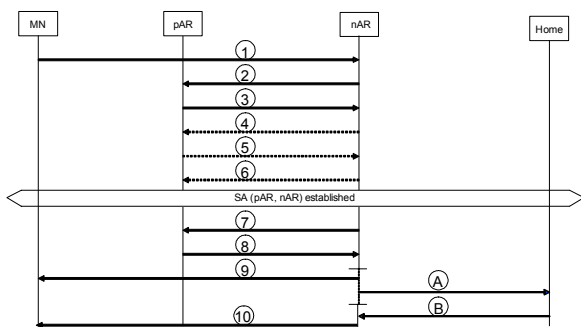


Fig. 2. Message sequence chart for message exchange in IDKE

Before explaining their purposes in detail, we examine the knowledge of all three nodes before the mechanism starts:

- MN: The mobile node has knowledge about the identifiers of the pAR, typically its IP address. The MN and the pAR share the session master secret (K_{sms}). Furthermore, the MN knows the identifier of the nAR, e.g., the nAR's IP address, since it got an advertisement from the nAR. Also, the MN has knowledge about a unique session identifier.
- pAR: The previous AR obviously knows about the MN's home address, the shared K_{sms} and the unique session identifier. It has knowledge about the QoS

settings and other parameters if any. In the scenario we consider here it has no security association to the nAR and might not even be aware of it.

- nAR: The new AR knows the local address of the MN it advertised, e.g., the MN's care-of-address (CoA) which is the IP address for routing purposes. We assume that the nAR has no knowledge about the MN's home IP address or at least does not trust the MN.

New messages introduced by IDKE are grouped as follows: 3 messages to initiate the transport of credentials between the pAR and the nAR, 2 messages for home registration, and 2 messages to notify the MN about the key status. First, the three messages that belong to initiation of the key-transport are explained in more detail:

The first message called IDKE_initiation is sent from the MN to the nAR after the MN got a router advertisement from the nAR and therefore knows the nAR's IP address. The message type is 1, which identifies the initiation message. The message consists of sender and receiver ID and a Session Identifier that uniquely identifies the prior session of the MN when communicating over the pAR. Furthermore, there is a timestamp created by the MN. The key holder ID is the pAR since the nAR might not be aware of the pAR at all. The MN_Token is a Message Authentication Code (MAC) for all important data and encrypted by the shared secret K_{sms} between the MN and the pAR. The detailed message format is as follows:

```

Message 1: [Message_Type=1; SenderID=MN_IP;
ReceiverID=nAR_IP; Session_ID; MN_Timestamp;
Key_Holder_ID=pAR_IP;
MN_Token=MAC(K_sms){SenderID=MN_IP; Session_ID;
MN_Timestamp}]
  
```

The second message called IDKE_forward is similar to the first one. The message type is 3, the sender is the nAR, the receiver is the pAR, and the initiator is the MN. MN_Token and the NM_Timestamp are just copies from message 1. The detailed packed format is as follows:

```

Message 2: [Message_Type=3; SenderID=nAR_IP;
ReceiverID=pAR_IP; Initiator=MN_IP; Session_ID;
MN_Timestamp; nAR_Token=MAC(K_nar){pAR_IP,
MN_ID, Session_ID; MN_Timestamp};
MN_token=MAC(K_sms){...}]
  
```

The third message called IDKE_acknowledge is sent from the pAR to the nAR. The message type is 5, the sender is the pAR, and the receiver is the nAR. The nAR token from message 2 is sent back to the nAR and the MN timestamp is still the same as in message 1.

```

Message 3: [Message_Type=5; SenderID=pAR_IP;
ReceiverID=nAR_IP; Initiator=MN_IP; Session_ID;
MN_Timestamp; nAR_Token=MAC(K_nar){...}]
  
```

The message types 1, 3, and 5 are used to identify these messages. Other message types are for error handling but not considered in this paper. The reader might just imagine that whenever an error occurs the MN will be notified and uses another mechanism for authentication.

Message 4 to 6 belong to the IKE protocol (aggressive mode) establishing the SAs required for IPsec. Message 7 and 8 are CTP_request and CTP_data to transfer the requested context / key from pAR to nAR.

The second group of messages consists of two optional messages: IDKE_home_req (message A) and IDKE_home_ack (message B). These messages are exchanged between the nAR and the home network. The first message is sent to the home network for two purposes. The first reason is to prove the validity of the key received by the pAR. The second reason is to inform the home network about the AR change and to change the key at the home network.

Message A: [Message_Type=15; SenderID=nAR_IP;
ReceiverID=HA_IP; Session_ID; K_{ran} ;
Signature=MAC(K_{sms}){ K_{ran} , MN_IP, Session_ID, pAR_ID}]

The message contains the random value K_{ran} so that the home network gains knowledge about the new key. A MAC on the random number and the identities of the MN and the nAR are also included in the message. This guarantees the validity of the key K_{sms} since it is checked again by the home network. The random number K_{ran} sent to the home network is to change the key while an MN moves from one AR to another. The home network replies by sending message B. This message is not described in detail here since it basically just contains an acknowledgement.

Thirdly we describe the last two messages that belong to the third group. These messages are for establishing the new key at the MN and to confirm the registration process. Message 9 is the so-called IDKE_tmp_key_established which informs the MN about the success of the key transfer from the pAR to the nAR. It also includes the random number to enable the MN to generate the new key. This message could either be sent before or after message A.

Message 9: [Message_Type=19; SenderID=pAR_IP;
ReceiverID=MN_IP; Session_ID; MN_Timestamp; K_{ran} ;
Signature=MAC(K_{sms}){ K_{ran} , MN_IP, Session_ID, pAR_ID}]

When the home registration procedure has been successful the nAR sends message 10 (IDKE_accomplished) to the MN. This message is not described in more detail here since it only contains the acknowledgement information sent from the home network to the nAR.

D. Detailed Discussion

The first three messages introduced above belong to the first group. These three messages are exchanged prior to the key exchange. Here we explain how these three messages achieve trust establishment between the ARs. All important

parameters that are used in this process are also examined in more detail. Furthermore, we describe the concept of key regeneration which is performed at the nAR after the key has been transferred.

1) Token Generation in the MN

The MN computes a token by computing a MAC over all relevant data. This MAC is encrypted by K_{sms} which is the key shared between MN and pAR. This token will eventually reach the pAR since it is forwarded by the nAR. Once the pAR gets the token it knows that it was sent by the MN, even if the MN is not connected to the pAR any more. The pAR has knowledge about K_{sms} and all other encryption parameters such as encryption algorithms and is therefore able to re-compute the token. This token acts as a digital signature which the MN uses to sign the message. Since the MN knows the pAR's ID before creating the token, it even authenticates the nAR for the pAR.

2) Token Processing in the nAR

The role of the nAR is basically to forward the MN's token to the pAR. In order to stay stateless the nAR computes its own token and attaches it to the message as well. The token generation is comparable to the MN's token but the nAR uses a secret key – K_{nar} – to create the MAC. K_{nar} is only known by the nAR and its only purpose is to create such tokens. When the pAR receives the token it will send it back to the nAR by attaching it to message 3. Therefore the token authenticates the pAR for the nAR. After message 3 the two access routers are aware of each other. Now they can continue by utilizing IKE for SA establishment and CTP for session-key forwarding.

3) Timestamp

The MN computes a timestamp before sending the first message. This timestamp persists during the transfer from the MN to the nAR, from the nAR to the pAR and back. This timestamp ensures that the message is fresh and no replay attack is performed. The scope of this timestamp can be set up to the granularity of minutes because a replay attack is not expected to occur within this scope. Actually, an attacker would need several days to re-compute the MAC that signs the message. This granularity ensures that even in case of manually set time – one might imagine a user setting up the time of its device manually – the mechanism is still secure and applicable. Whenever receiving a message, both the nAR and the pAR will check the timestamp, elaborate whether the age of the message is acceptable or not and drop the message according to their decision.

4) Key regeneration

The idea behind the key regeneration is to keep it fresh and unique. Therefore the nAR performs the logical operation to modify the key when receiving it from the pAR. Here it is important to mention that this operation should be

considered as optional. Since the random number K_{ran} is always sent as clear text even over the wireless link, the nAR might even choose a random number equal to zero that does not change the key at all. The purpose is just to limit the validity of a key in order to put additional obstacles for an attacker since he needs to record all random numbers as well and therefore follow the MN or need to have connectivity to the core network near the MN's home network.

4. ANALYSIS

In this section the properties of IDKE is analyzed in more detail, then we compare IDKE with some other approaches.

A. IDKE Properties

We show that IDKE meets all requirements mentioned in Section III.A. First, from performance aspects IDKE satisfies the requirement of sending a minimum number of messages along the wireless link. Only three messages are exchanged over the wireless link: the IDKE_initiation message, the IDKE_tmp_key_established and the IDKE_accomplished or alternatively an error notification message. Thus IDKE is expected not to consume much additional bandwidth and not to introduce more security hole on the bottleneck wireless link. Second, IDKE is easy to implement since it utilizes IKE and CTP which are well-known IETF protocols. Third, because of IDKE's token concept no state is set at the nAR until the three IDKE messages are exchanged. Since states are set as late as possible IDKE is robust against DoS attacks that try to block the AR by initiating unnecessary states. Furthermore, the requirement of a secure key exchange is met since IDKE forwards K_{sms} over a secure channel between the pAR and nAR. Finally an intruding or malicious sniffing MN does not have a chance to steal the K_{sms} since it will never be transferred over the wireless link and the concept of key regeneration provides additional security. Session hijacking or unauthorized encryption is therefore impossible.

Key exchange by IDKE is therefore as strong as the weakest points of our chain; that are: K_{sms} itself, the SA established by IKE and the MN token. Like other strong approaches IDKE avoids sending the K_{sms} to the MN and uses strong encryption on the forwarding path. Therefore IDKE can be considered as invulnerable if high encryption is used on the channel between pAR and nAR. However, we recommend choosing the order of magnitude of MN-token and IKE-key appropriately higher than K_{sms} .

B. Comparison with Other Approaches

As mentioned above, when an MN moves from a pAR to an nAR, it has to establish mutual trust relationship with the nAR: nowadays not only foreign end devices, even 802.11x hotspots (the network) might be malicious or untrustworthy. One common mechanism to authenticate the MN is to

request the required information from its home network. An example of this obvious and simple mechanism is the GSM authentication mechanism (a detailed description can be found in [1]). Different from IDKE, it does support for the MN to authenticate the network. Under performance aspects, the authentication phase in GSM authentication requires expensive end-to-end communication over long distances and wireless media might be critical for fast re-authentication. For GSM authentication, a challenge response mechanism authenticates the MN. Basically a random number is sent from the Home Location Register (HLR) to the MN. The MN computes a session-key (K_c) corresponding to our K_{sms} by generating it from the random number by the aid of K_i that we would call Master Secret Key. K_c is sent back to the HLR and authenticates the MN (Fig. 3a). The HLR will send an acknowledgement to the Access point if K_c is valid. This authentication mechanism is fast but insecure since it authenticates the MN for the network but not vice versa.

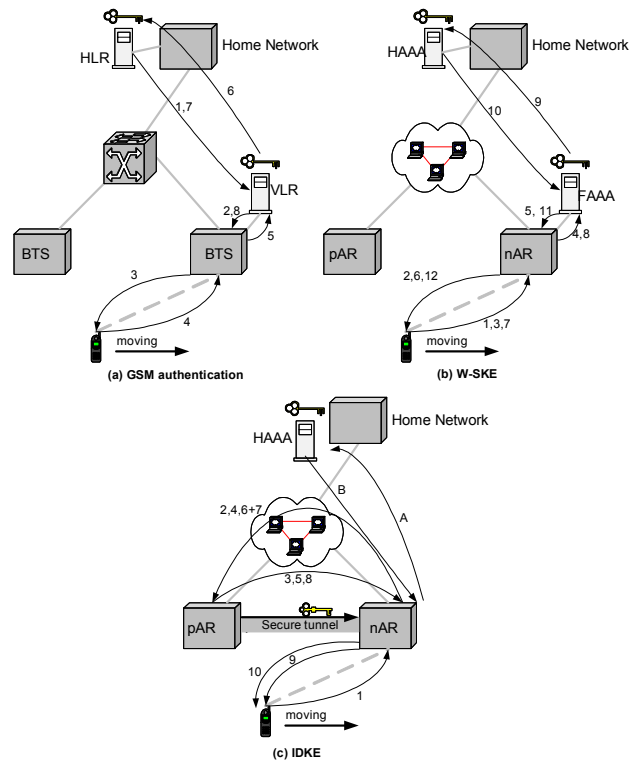


Fig. 3. Message flow comparison

A recent proposal, the W-SKE [5, 8] protocol, tries to provide mutual authentication and minimizes the amount of data exchanged between the foreign network and the HAAA server. However, it requires a high amount of messages exchanged on the wireless link. Intuitively, this protocol can be assumed to be secure but causes a high overhead in terms of low bandwidth on the wireless link. W-SKE utilizes EAPOL for the wireless link for the first message and RADIUS to communicate with AAA servers. The messages

as shown in Fig. 3b correspond to two challenge response mechanisms one for the FAAA server and one for the HAAA server. IDKE has the advantage of reducing the amount of messages that are transferred on the wireless link and the amount of messages exchanged with the HAAA server. It uses three messages on the wireless link and just one message exchange with the HAAA server.

Fig. 3 illustrates the typical message exchanges in all three approaches: GSM, W-SKE and IDKE and therefore gives an impression where dataflow occurs mainly. Furthermore, in Table 1 we give a summary about the properties of all three mechanisms. It can be concluded that IDKE has the advantages over the other two approaches by providing both fast key exchange and mutual authentication, and meanwhile raising a minimal requirement on wireless bandwidth.

TABLE I
COMPARISON OF GSM, W-SKE, AND IDKE

	GSM	W-SKE	IDKE
Authenticate the MN	Yes	Yes	Yes
Authenticate the access network	No	Yes	Yes
Key exchange	Yes	Yes	Yes
Messages over the wireless link	2	6	3
Messages within the access network	3	4	6
Messages to the home network	3	2	2

5. CONCLUSIONS AND FUTURE WORK

In this paper we introduced and elaborated a new key exchange mechanism. We showed how this approach could be used to establish a session-key at the new access router by forwarding the key from the previous router. We found that utilizing the context transfer protocol (CTP) is a potential way for establishing such keys but that currently the security association required for CTP is missing in inter-domain scenarios. The simple but elegant way that our mechanism uses is to establish the SA prior to CTP by setting up a shared key at the two ARs. In order to do so, we present a temporary token concept and show how IKE can be used to guarantee appropriate security against DoS and session hijacking. Through analytical comparison with other approaches such as GSM authentication and W-SKE, we found out that our mechanism supports mutual authentication and uses minimal message exchanges at the wireless link as well as to the home network, therefore having the advantage of being fast and requiring low bandwidth consumption in wireless environments. Since it utilizes IETF standards for key exchange and context forwarding, it is more likely to be able to get reasonable acceptance from an engineering point of view, and can be a promising way to establish necessary keys and mutual authentication between the MN and the access networks in IP handover scenarios.

Since one of the main advantages of IDKE is the improved performance, we are currently performing simulation studies by using OPNET Modeler [11] in order to compare several key exchange mechanisms when moving in different wireless access networks. The focus is on improving the performance parameters such as message latency, key computation times, and the overall key exchange delays in different scenarios. We consider inter- vs. intra-domain scenarios as well as reusing SAs and secure tunnels between two ARs for several MN and therefore reducing the amount of messages exchanged between the ARs. Another interesting scenario is an extremely fast moving MN and the chain-of-trust problem since the transitivity of trust can not be assumed. Beyond this simulation, we interoperate with AAA architectures (e.g., RADIUS [12] and Diameter [13]). All these aspects are not described in more detail in this paper due to space limitation.

6. REFERENCES

- [1] C.-H. Lee, M.-S. Hwang, and W.-P. Yang, "Enhanced Privacy and Authentication for the Global System for Mobile Communications", *Wireless Networks*, Volume 5, Issue 4, pp.231 - 243, 1999.
- [2] L. Blunk and J. Vollbrecht, "Extensible Authentication Protocol (EAP)", Internet Draft (work in progress), IETF, Feb. 2004.
- [3] H. Haverinen and J. Salowey, "EAP SIM Authentication", Internet Draft (work in progress), IETF, Oct. 2003.
- [4] J. Arkko and H. Haverinen, "EAP AKA Authentication", Internet Draft (work in progress), IETF, Oct. 2003.
- [5] L. Salgarelli, "EAP SKE authentication and key exchange protocol", Internet Draft (work in progress), IETF, May 2003.
- [6] J. Loughney M. Nakhjiri, C. Perkins, and R. Koodli, "Context Transfer Protocol", Internet Draft (work in progress), IETF, Apr. 2004.
- [7] D. Harkins and D. Carrel, "The Internet Key Exchange (IKE)", RFC 2409, IETF, Nov. 1998.
- [8] L. Salgarelli, M. Buddhikot, J. Garay, S. Patel, and S. Miller, "Efficient Authentication and Key Distribution in Wireless IP Networks", *IEEE Wireless Communications*, Vol. 10 No. 6, Dec. 2003.
- [9] C. Kaufman, R. Perlman, and B. Sommerfeld, "DoS Protection for UDP-based Protocols", 10th ACM Conference on Computer and Communication Security, Washington D.C., USA, Oct 2003.
- [10] S. Kent and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, IETF, Nov. 1998.
- [11] The OPNET Modeler, <http://www.opnet.com/products/modeler>.
- [12] C. Rigney and S. Willens, "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, IETF, June 2000.
- [13] P. Calhoun, J. Loughney, "Diameter Base Protocol", RFC 3588, IETF, Sept. 2003.