Extensions to an Authentication Technique Proposed for the Global Mobility Network

Levente Buttyán, Constant Gbaguidi, Sebastian Staamann, and Uwe Wilhelm

***Abstract*—We present three attacks on the authentication protocol that has been proposed for the so-called global mobility network in the October 1997 issue of the *IEEE Journal on Selected Areas in Communications*. We show that the attacks are feasible and propose corrections that make the protocol more robust and resistant against two of the presented attacks. Our aim is to highlight some basic design principles for cryptographic protocols, the adherence to which would have prevented these attacks.**

***Index Terms*—Entity authentication, global mobility network.**

I. Introduction

In [7], an authentication technique has been proposed for use in the so-called global mobility network (GLOMONET), which provides a personal communication user with global roaming service. The proposed authentication technique consists of the following two phases:

- *roaming-service-setup phase*, in which authentication that is required to set up the roaming-service environment is performed by the visited (roamed) network, the home network, and the roaming user;
- *roaming-service-provision phase*, in which authentication that is necessary to provide the roaming service within the visited network is performed only by the visited network and the roaming user.

The motivation for this two-phase model is to have the home network involved in the authentication process only once, during the roaming-service-setup phase. In this phase, a secret key is established between the visited network and the roaming user with the help of the home network. This secret key is used later in the roaming-service-provision phase to authenticate the roaming user and the visited network to each other without any contribution from the home network. Thus, as long as the roaming user stays in the region of the visited network, authentication can be performed without contacting the home network of the roaming user (unlike, for instance, in the GSM system, where the visited network often has to obtain challenge-response pairs from the home network in order to authenticate the roaming user [3]).

The following authentication protocol is used in the roaming-service-setup phase:

1) $U \rightarrow V$: *Request*;
2) $V \rightarrow H$: rnd_1;
3) $H \rightarrow V$: $K_{vh}F(rnd_1), rnd_2$;
4) $V \rightarrow H$: $K_{vh}F(rnd_2), K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$;
5) $H \rightarrow V$: $K_{uh}F(K_{\text{tmp}}F(K_{\text{auth}}))$;
6) $V \rightarrow U$: $rnd_3, K_{\text{tmp}}, K_{uh}F(K_{\text{tmp}}F(K_{\text{auth}}))$;
7) $U \rightarrow V$: $K_{\text{auth}}F(rnd_3)$;
8) $V \rightarrow U$: $K_{\text{auth}}F(K_{\text{auth}}F(rnd_3))$;

where U, V, and H denote the roaming user, the visited network, and the home network, respectively; rnd_1, rnd_2, and rnd_3 are random numbers; K_{vh} is a long-term secret key shared by V and H; K_{uh} is a long-term secret key shared by U and H; K_{tmp} and K_{auth} are keys generated by V[1]; $KF(x)$ denotes x encrypted with the key K; and $A \rightarrow B.M$ means that A sends the message M to B.

The protocol is described in more detail below as follows.

1) The roaming user U sends a service request to the visited network V.
2) V sends the random number rnd_1 to the home network H of the roaming user. This is a challenge for authenticating H.
3) H responds to V's challenge with the random number rnd_1 encrypted with the key K_{vh} [i.e., $K_{vh}F(rnd_1)$], and sends another random number rnd_2 to V. This is a challenge for authenticating V.
4) V verifies if it has received back its random number rnd_1 encrypted with the key K_{vh}. If so, then V believes that it talks with H, since the key K_{vh} is known only to H and V, and thus, V believes that H sent message 3 (at least the first, encrypted part of it). V generates the user authentication key K_{auth} and the temporary cipher key K_{tmp}. Then, V responds to H's challenge with $K_{vh}F(rnd_2)$, and sends $K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$ to H.
5) H verifies if it has received back its random number rnd_2 encrypted with the key K_{vh}. If so, then H believes that it talks with V. H decrypts $K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$ with K_{vh} and re-encrypts the result $K_{\text{tmp}}F(K_{\text{auth}})$ with the key K_{uh}. H sends $K_{uh}F(K_{\text{tmp}}F(K_{\text{auth}}))$ to V.
6) V forwards $K_{uh}F(K_{\text{tmp}}F(K_{\text{auth}}))$ to U along with the key K_{tmp} and the random number rnd_3, which is a challenge for authenticating U.
7) U uses the key K_{uh} that she shares with H, and the key K_{tmp} that she has just received to obtain the authentication key K_{auth}. Then, U responds to V's challenge with $K_{\text{auth}}F(rnd_3)$.

Paper approved by B. Jabbari, the Editor for Wireless Multiple Access of the IEEE Communications Society. Manuscript received October 15, 1998; revised May 15, 1999.

L. Buttyán and C. Gbaguidi are with the Institute for Computer Communications and Applications, Swiss Federal Institute of Technology, CH-1015 Lausanne, Switzerland.

S. Staamann is with XTRADYNE Technologies AG, D-10119 Berlin, Germany.

U. Wilhelm is with the Operating Systems Laboratory, Swiss Federal Institute of Technology, CH-1015 Lausanne, Switzerland.

Publisher Item Identifier S 0090-6778(00)02268-6.

[1]We changed the original notation for keys in [7] in order to make understanding of the protocol and the attacks presented later easier. Our notation is consistent with the notation used in the majority of cryptographic literature.

8) V verifies if it has received back its random number rnd_3 encrypted with the fresh authentication key K_{auth}. If so, then V believes that it talks with U. V sends $K_{\text{auth}}F(K_{\text{auth}}F(rnd_3))$ to U. U verifies if she has received back $K_{\text{auth}}F(rnd_3)$ encrypted with K_{auth}. If so, then U believes that she talks with V.

The following goal of the protocol is threefold.

- To authenticate the visited network V and the home network H to each other.
- To establish K_{auth} as a shared secret between the roaming user U and the visited network V. Then, K_{auth} can be used for authenticating U and V to each other in later steps of the protocol and in the roaming-service-provision phase without contacting H, as long as U stays in the region of V. We note, that according to [7], even the home network should not know K_{auth}. This is the reason why K_{auth} is encrypted with the temporary cipher key K_{tmp} in message 4.
- To authenticate the roaming user U and the visited network V to each other.

Unfortunately, the protocol has serious flaws that permit various attacks. In this letter, we present three attacks against the protocol and we propose corrections to prevent them. The first attack enables a legitimate, but malicious user to obtain the authentication key K_{auth} established between the roaming user and the visited network. In this way, the intruder can impersonate the roaming user or the visited network. The second attack allows the intruder to feed the roaming user with a compromised, old, authentication key and, thus, to masquerade as the visited network. In the third attack, we show that the home network can easily obtain the authentication key K_{auth}, which was intended by the protocol design to be kept confidential between the roaming user and the visited network, and to be hidden from the home network.

II. Attacks

A. Attack 1

In this attack, the intruder I obtains the authentication key K_{auth} of the roaming user U and the visited network V. We assume that I is a legitimate but malicious user from the same home network H as the roaming user U, and that I eavesdropped and recorded the protocol run, in which K_{auth} has been established. Thus, I knows $K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$ and K_{tmp}. The attack scenario is as follows:

1) $U \rightarrow V$: *Request*;
2) $V \rightarrow H$: rnd_1;
3) $H \rightarrow V$: $K_{vh}F(rnd_1), rnd_2$;
4) $V \rightarrow H$: $K_{vh}F(rnd_2), K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$;
5) $H \rightarrow V$: $K_{uh}F(K_{\text{tmp}}F(K_{\text{auth}}))$;
6) $V \rightarrow U$: $rnd_3, K_{\text{tmp}}, K_{uh}F(K_{\text{tmp}}F(K_{\text{auth}}))$;
7) $U \rightarrow V$: $K_{\text{auth}}F(rnd_3)$;
8) $V \rightarrow U$: $K_{\text{auth}}F(K_{\text{auth}}F(rnd_3))$
   a) $I \rightarrow V$: *Request'*;
   b) $V \rightarrow H$: rnd_1';
   c) $H \rightarrow V$: $K_{vh}F(rnd_1'), rnd_2'$;
   d) $V \rightarrow I(H)$: $K_{vh}F(rnd_2'), K_{vh}F(K_{\text{tmp}}'F(K_{\text{auth}}'))$;
   e) $I(V) \rightarrow H$: $K_{vh}F(rnd_2'), K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$;
   f) $H \rightarrow V$: $K_{ih}F(K_{\text{tmp}}F(K_{\text{auth}}))$;
   g) $V \rightarrow I$: $rnd_3', K_{\text{tmp}}', K_{ih}F(K_{\text{tmp}}F(K_{\text{auth}}))$;

where $V \rightarrow I(H)$. M means that the message M that was sent by V to H is intercepted by I, and thus, it is not delivered to H; $I(V) \rightarrow H$. M means that I sends the message M to H claiming that it is originated from V.

Let us assume that U, V, and H successfully run the protocol. In step 4) and in step 6), I eavesdrops $K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$ and K_{tmp}, respectively. Then, I starts the protocol with V. Since I is a legitimate user, she can start the protocol with V. I lets the protocol run until step 4). In step 4), I exchanges the second part of the message $K_{vh}F(K_{\text{tmp}}'F(K_{\text{auth}}'))$ to $K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$. H decrypts $K_{vh}F(K_{\text{tmp}}F(K_{\text{auth}}))$ and re-encrypts the result $K_{\text{tmp}}F(K_{\text{auth}})$ with K_{ih}, which is the long-term key shared by I and H. When I receives message 6 from V, she obtains K_{auth}, since she knows both K_{ih} and K_{tmp}. Later, I can use K_{auth} to impersonate U or V in the roaming-service-provision phase.

Obviously, attack 1 is a coordinated activity of several physically dispersed malicious entities that we consider jointly to be the intruder. Two of these entities eavesdrop the communication between U and V, and between V and H, respectively. A third one starts the protocol with V, and a fourth one modifies message 4 in transit between V and H. We assume that the intruder entities communicate with each other (possibly in a proprietary way). The attack requires that the intruder can associate messages, which are observed on different interfaces, to each other. In particular, I has to eavesdrop message 4 of the original protocol run on the V-H interface and message 6 from the same protocol run on the V-U interface. Similarly, I has to catch and modify message 4 on the V-H interface, which belongs to the protocol run initiated by I on the I-V interface. Considering that message 4 has to contain information about the initiator of the protocol[2] (otherwise H would not know which key to use to encrypt message 5), the problem of associating the right messages to each other does not seem to be too difficult.

Furthermore, the attack does not require the modification of messages on the air interface (between U and V), which would be quite difficult to do. Whereas, eavesdropping of messages sent over a wireless connection is considered to be rather simple, because of the broadcast nature of wireless communication. Eavesdropping and modifying messages that are sent between the visited network and the home network is technically possible, since these networks are usually connected via a fixed network. Therefore, we believe that attack 1 is feasible.

However, the fixed network is usually assumed to be physically protected, and thus, the communication between the visited network and the home network is considered to be secure. If we accept this assumption, then neither eavesdropping nor modification of messages is possible between V and H, and attack 1 no longer works. But we note, that a small modification of the protocol would also prevent attack 1, and we would not need the assumption of the security of the fixed network. We would

[2]This information is not shown explicitly in the protocol description.

prefer to make the protocol itself more robust than to rely on strong assumptions.

Correction: Attack 1 is possible, because message 4 is not explicit enough: it does not contain any information about the intended recipient of the authentication key K_{auth}. This weakness allows the intruder to replace a part of message 4 with the corresponding part from another protocol run. Thus, following [1, Principle 3], which says that it is prudent to mention a participant's name explicitly in the message if the identity of that participant is essential to the meaning of the message, we propose to include the identifier of the roaming user explicitly in message 4. For similar reasons, we also suggest to include the identifier of the visited network in message 5. Although these modifications would be enough to prevent attack 1, in order to make the protocol even more robust, we propose to encrypt the first and the second part of message 4 together. Binding the random number rnd_2 to the rest of the message prevents the replay of message 4 or any parts of it. Furthermore, replacing parts of this message is no longer possible. Therefore, our correction is the following:

...
4) $V \rightarrow H$: $K_{vh}F(rnd_2, U, K_{\text{tmp}}F(K_{\text{auth}}))$
5) $H \rightarrow V$: $K_{uh}F(V, K_{\text{tmp}}F(K_{\text{auth}}))$
6) $V \rightarrow U$: $rnd_3, K_{\text{tmp}}, K_{uh}F(V, K_{\text{tmp}}F(K_{\text{auth}})) \cdots$
...

Although it is not mentioned explicitly in [7], the original protocol is probably intended to be implemented with a block cipher used in electronic code book (ECB) mode. Our modifications clearly rule out such a mode of encryption, because they require the encryption of several concatenated parts, which probably exceed the block size of the cipher. The usual solution in this case is to use a block cipher in cipher-block chaining (CBC) mode. This, however, involves use of an initial parameter (initial vector or IV) that has to be known by the parties in order to be able to communicate properly. Since this initial parameter need not be secret, it can be sent along with the encrypted message.[3] The integrity of IV is protected by the mechanism described in the next paragraph.

Since much of the data that is being encrypted is random, an additional mechanism is needed to protect the integrity of messages (i.e., to be able to detect modifications and replacements of blocks in the cipher-block chain as well as of IV). For this reason, we suggest that first a cryptographic hash value is computed from the data to be sent using a hash function such as, for instance, MD5 [6], and then the data is encrypted together with the hash value using a block cipher such as, for instance, DES [4] in CBC mode.[4]

Although our modifications require some additional mechanisms, none of these are considerably complex or time consuming operations. We believe that these modifications would result in more gain (robustness and resistance against attacks) than loss (complexity and performance).

[3]For the sake of simplicity, we do not make this explicit in the protocol description.

[4]For simplicity, we do not make this explicit either.

B. Attack 2

This attack, which enables an intruder I to impersonate the visited network V to the roaming user U, exploits the fact that U does not receive any fresh message in the protocol. Attack 2 is directly related to [1, Principle 9], which says that a key may have been used recently, yet be quite old and possibly compromised, and is essentially the same as the attack against the Needham–Schroeder protocol [5] described in [2]. In attack 2, we assume that K^*_{auth} is a compromised, old, authentication key and that the intruder I recorded the protocol run that established K^*_{auth}. Thus, I possesses the old authentication key K^*_{auth}, the corresponding temporary cipher key K^*_{tmp}, and the ciphered message $K_{uh}F(K^*_{\text{tmp}}F(K^*_{\text{auth}}))$.

The attack scenario is as follows:

1) $U \rightarrow I(V)$: *Request*;
6) $I(V) \rightarrow U$: $rnd'_3, K^*_{\text{tmp}}, K_{uh}F(K^*_{\text{tmp}}F(K^*_{\text{auth}}))$;
7) $U \rightarrow I(V)$: $K^*_{\text{auth}}F(rnd'_3)$;
8) $I(V) \rightarrow U$: $K^*_{\text{auth}}F(K^*_{\text{auth}}F(rnd'_3))$.

When U starts a new instance of the protocol with the *Request* message, I plays back message 6 from the old protocol.[5] U thinks that the authentication key is K^*_{auth}, so she sends $K^*_{\text{auth}}F(rnd'_3)$ to V. This message is intercepted by I. I generates the last message $K^*_{\text{auth}}F(K^*_{\text{auth}}F(rnd'_3))$ and sends it to U. In this way, I can impersonate the visited network V.

In attack 2, the intruder has to be able, with regard to the network technology, to play the role of the visited network and to make the roaming user send messages to her instead of the visited network. Although, this requirement seems to be strong, satisfying it is not impossible. There are commercially available devices called "IMSI catchers," the functionality of which is very similar to that needed by the intruder in attack 2. From the side of the mobile phone, an "IMSI catcher" behaves as a base station of the mobile network. A mobile phone, which is closer to an "IMSI catcher" than to a base station, can be coerced by the "IMSI catcher" to establish a connection with it rather than with the base station. The mobile phone does not even know that it talks with an "IMSI catcher," instead of a base station. The "IMSI catcher" can relay communication between the mobile phone and the base station and stays unnoticed. We believe that such a device would enable attack 2.

Correction: To prevent attack 2, the roaming user should receive something fresh in message 6. A possible solution is that the roaming user generates a random number at the beginning of the protocol and sends this number to the visited network together with the service request. The visited network forwards this random number to the home network, which can include it in message 5. The modified messages are as follows:[6]

1) $U \rightarrow V$: *Request*, rnd_0
...

[5]I may change the random number rnd_3 to rnd'_3.

[6]The messages include the modifications proposed earlier to prevent attack 1.

4) $V \rightarrow H$: $K_{vh}F(rnd_2, U, K_{\text{tmp}}F(K_{\text{auth}}), rnd_0)$
5) $H \rightarrow V$: $K_{uh}F(V, K_{\text{tmp}}F(K_{\text{auth}}), rnd_0)$
6) $V \rightarrow U$: $rnd_3, K_{\text{tmp}}, K_{uh}F(V, K_{\text{tmp}}F(K_{\text{auth}}), rnd_0)$
...

C. Attack 3

In this attack, we show that the home network H can easily obtain the authentication key K_{auth}, which is intended to be a shared secret between the roaming user U and the visited network V. The attack requires a collaborator of the home network, which eavesdrops the communication between the roaming user and the visited network. Let us assume that U started the protocol. In step 4), H stores $K_{\text{tmp}}F(K_{\text{auth}})$. In step 6), the collaborator of H eavesdrops K_{tmp} and sends it to H. H decrypts the previously stored message $K_{\text{tmp}}F(K_{\text{auth}})$ and obtains the key K_{auth}.

In attack 3, a participant of the protocol, the home network, instead of an intruder, mounts the attack. The described behavior of the home network seems to be unusual, and we hasten to note that we would not consider this to be an attack if the authors in [7] had not considered that "it is possible that the entities concerned take illegal action in roaming-service provision. Therefore, it is desirable not to leak the authentication keys needed for their authentication to the other networks." It is clear that this desire is not satisfied by the protocol.

We cannot suggest a correction to prevent this attack because we believe that it cannot be prevented in the given situation, where the sole pre-established secure channel between the visited network and the roaming user goes through the home network. Therefore, we advise not to use this protocol if the home network cannot be trusted (i.e., if it is possible that the home network tries to acquire the key K_{auth} and to misuse it).

On the other hand, it should be noted that if we assume that the home network is trusted, then there is no particular reason anymore to hide K_{auth} from it. In that case, we can go without the temporary key K_{tmp} and obtain a simplified protocol as follows that uses less encryption and decryption operations:

1) $U \rightarrow V$: *Request*, rnd_0;
2) $V \rightarrow H$: rnd_1;
3) $H \rightarrow V$: $K_{vh}F(rnd_1), rnd_2$;
4) $V \rightarrow H$: $K_{vh}F(rnd_2, U, K_{\text{auth}}, rnd_0)$;
5) $H \rightarrow V$: $K_{uh}F(V, K_{\text{auth}}, rnd_0)$;
6) $V \rightarrow U$: $rnd_3, K_{uh}F(V, K_{\text{auth}}, rnd_0)$;
7) $U \rightarrow V$: $K_{\text{auth}}F(rnd_3)$;
8) $V \rightarrow U$: $K_{\text{auth}}F(K_{\text{auth}}F(rnd_3))$.

III. Conclusion

In this letter, we presented three attacks on the authentication protocol that has been proposed for the so-called global mobility network in [7]. We showed that the attacks are feasible, and that careful design would have prevented them. We used the basic principles for the design of cryptographic protocols described in [1] to correct the protocol and to make it resistant against two of the attacks presented. In particular, we directly used Principle 3, which says that it is prudent to mention the participant's name explicitly in the message if the identity of that participant is essential to the meaning of the message, and Principle 9, which says that a key may have been used recently, yet be quite old and possibly compromised. We hope that the attacks presented in this letter remind designers of these principles, and we advise adherence to them. The third attack allowed us to discover the need for an assumption, which has to be made in order for the protocol to be correct. Moreover, making this assumption allowed us to simplify the protocol. We emphasize that all the assumptions, which the correctness of the protocol depends on, should be stated explicitly and clearly, so that someone reviewing the design can see whether they are acceptable or not.

References

[1] M. Abadi and R. Needham, "Prudent engineering practice for cryptographic protocols," in *Proc. IEEE CS Symp. Res. Security and Privacy*, 1994, pp. 122–136.
[2] D. Denning and G. Sacco, "Timestamps in key distribution protocols," *Commun. ACM*, vol. 24, no. 8, pp. 198–208, 1981.
[3] A. Mehrotra and L. Golding, "Mobility and security management in the GSM system and some proposed future improvements," *Proc. IEEE*, vol. 86, pp. 1480–1496, July 1998.
[4] *National Bureau of Standards. Data Encryption Standard*, Federal Information Processing Standards Pub. 46, 1977.
[5] R. Needham and M. Schroeder, "Using encryption for authentication in large networks of computers," *Commun. ACM*, vol. 21, no. 12, pp. 993–999, 1978.
[6] R. Rivest, "The MD5 message-digest algorithm," in *Internet Request for Comments 1321 (presented at Rump session of Crypto '91)*.
[7] S. Suzuki and K. Nakada, "An authentication technique based on distributed security management for the global mobility network," *IEEE J. Select. Areas Commun.*, vol. 15, pp. 1608–1617, 1997.