Impersonation Attacks on a Mobile Security Protocol for End-to-End Communications

Reiner Dojen, Vladimir Pasca, and Tom Coffey

Data Communications Security Laboratory,
Department of Electronic & Computer Engineering,
University of Limerick,
Limerick, Ireland
{reiner.dojen,vladimir.pasca,tom.coffey}@ul.ie

Abstract. This paper presents an analysis of a cryptographic security protocol that is designed for use in a mobile communication environment. The goal of the analysed protocol is to ensure secure end-to-end communication between two mobile users that are connected to different base stations. The analysis reveals a serious flaw in the used signature scheme of the security protocol. Exploitation of this flaw enables an intruder to use algebraic simplifications to forge signatures on arbitrary messages. Two attacks, which exploit this weakness, are detailed showing the impersonation of a mobile user and a base station, respectively. Corrections to the flawed protocol are proposed and analysed. It is established that the corrected protocol is secure against the presented attacks.

Keywords: Mobile end-to-end communication, analysis of security protocols, impersonation attack, authentication and secrecy protocol.

1 Introduction

Wireless communications are driven by the need to provide network access for mobile computing devices. At the same time, the increase in available online services causes users to transmit more and more sensitive information. Security protocols are used to protect this sensitive information against many forms of attack. Basic security protocols allow agents to authenticate each other, to establish fresh session keys for confidential communication and to ensure the authenticity of data and services. Building on such basic security protocols more advanced services like non-repudiation, electronic payment and electronic contract signing are achieved. However, the design of effective security protocols is a challenging problem and often weaknesses are discovered in published protocols [1], [2], [3]. Additionally, mobile communication systems have some unique difficulties, such as limited bandwidth, high latency and unstable connections. Further, mobile devices often have low computational and storage capacities. These difficulties seriously limit the choices of applicable techniques to provide suitable protection. Thus, many security protocols have been specially designed for mobile communications [4], [5], [6], [7], [8].

A.U. Schmidt and S. Lian (Eds.): MobiSec 2009, LNICST 17, pp. 278 – 287, 2009.

This paper presents the analysis of the mobile end-to-end authentication and secrecy protocol proposed by Lee, Yang and Hwang [6]. This analysis ascertains a weakness in the employed signature scheme that allows an intruder to forge signatures on arbitrary messages. Two attacks are demonstrated that exploit this weakness and enable an intruder to successfully impersonate mobile users and base stations, respectively. Consequently, this also demonstrates that the protocol fails to achieve its goals of authentication and of privacy of the messages exchanged by the mobile users. Corrections to the flawed protocol are proposed that prevent these weaknesses and the immunity of the amended protocol against the presented attacks is established.

2 The LYH Mobile End-to-End Authentication and Secrecy Protocol

The security protocol proposed by Lee, Yang and Hwang [6] aims to provide secrecy and end-to-end authentication between two mobile users that are connected to different base stations. This protocol is divided into the following phases:

Certification Phase: A trusted certification authority (CA) distributes signed certificates to all protocol participants: Each mobile user and base station will receive only its own certificate. As the certification authority is only involved in the certification phase, it can be considered an offline certification authority.

Authentication Phase: During this phase, the two mobile users attempt to authenticate each other and to establish a secure common session key that is used in the subsequent communication phase.

Communication Phase: The two mobile users use the established session key to communicate securely with each other.

2.1 Certification Phase

Initially, the CA chooses three publicly known parameters (*p, q, g*): *p* is a large randomly selected prime number, *q* is a large prime factor of *p*-1 and $g = m^{(p-1)/q} \bmod p$, where m is an integer that satisfies 1 < *m* < *p*-1 and $m^{(p-1)/q} \bmod p >$ 1. The CA then calculates its own public/private key pair (y_{CA},x_{CA}), where x_{CA} is a randomly chosen private key and y_{CA} is the public key such that $y_{CA} = g^{x_{CA}} \bmod p$. Both x_{CA} and y_{CA} are members of GF(p)*. Then, each principal generates a public/private key pair with the corresponding properties and gives the public key as well as some proof of identity to the CA. The CA then creates the signed certificates (including certificate serial number, validity period, the identity (ID), the corresponding public key etc.) and issues them back to the owning principal.

Validity of CA's Signature. In general, a message signed by the CA consist of the triple (*M, s, t*), where *M* is the message itself, *s* is a random component of the signature and *t* is the message dependant component of the signature. The value s is calculated by formula (1), where *r* is a random number selected freshly for each individual signature and h(.) is a 2m-bit iterated hash function that is based on a m-bit block cipher with 2m-bit key. If the underlying block cipher has no weakness, then the hash function can be expected to have ideal computational security against the five attacks: target attack, free-start target attack, collision attack, semi-free-start collision attack and free-start collision attack [9].

$$s = g^{r} \bmod p \tag{1}$$

The message dependant component *t* of the signature is calculated according to formula (2) using the same value *r* and hash function h(.) as for *s*.

$$t = -s - h(M)x_{CA}{}^{-1} r \bmod q \tag{2}$$

Given a signed message (*M, s, t*), any principal can verify the validity of CA's signature (*s, t*) by establishing that equation (3) holds.

$$y_{CA}^{s+t} s^{h(M)} = 1 \bmod p \tag{3}$$

Assuming that (*s, t*) is a genuine signature, the calculation in (4) is performed to prove validity of the signature.

$$\begin{aligned}
& y_{CA}{}^{s+t} (s)^{h(M)} \\
& y_{CA}{}^{s-s-h(M)x_{CA}^{-1} r \bmod q} (s)^{h(M)} \\
& y_{CA}{}^{-h(M)x_{CA}^{-1} r \bmod q} (s)^{h(M)} \\
& (g^{x_{CA}} \bmod p)^{-h(M)x_{CA}^{-1} r \bmod q} (g^{r} \bmod p)^{h(M)} \\
& (g^{-h(M) r \bmod q} g^{h(M) r}) \bmod p \\
& g^{h(M) r - h(M) r \bmod q} \bmod p \\
& g^{0} \bmod p = 1 \bmod p
\end{aligned} \tag{4}$$

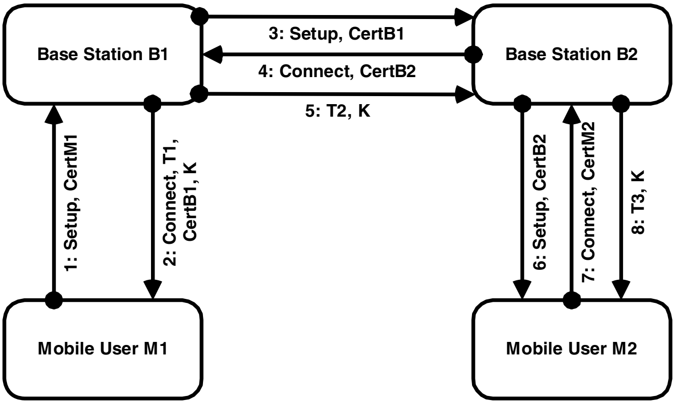

Fig. 1. LYH Protocol with Simplified Messages

2.2 Authentication Phase

In the second phase of the protocol, two mobile users M1 and M2 aim to communicate privately and mutually authenticate each other. Figure 1 outlines the protocol with simplified messages (the detailed messages are presented below in Figures 2, 3 and 4). The protocol can be divided into three parts: The first part comprises of messages 1 and 2 and establishes communication between M1 and B1. The second part consists of messages 3 to 5 and establishes communication between B1 and B2. Finally, the third part includes messages 6, 7 and 8 and establishes communication between B2 and M2.

Communication M1 to B1. The first set of messages establishes communication between the mobile user M1 and its base station B1 using the messages 1 and 2 as detailed in Figure 2.

1. M1 → B1: SETUP, Cert_{M1}
2. B1 → M1: CONNECT, Cert_{B1}, *T1*, $y_{M1}^{-x_{B1}f(T1,DH(B1,M1))}K \bmod p$

Fig. 2. Message exchange between the initiator M1 and the base station B1

M1 sends a SETUP signal and his signed certificate Cert_{M1} to base station B1. B1 verifies the signature using equation (3). If the verification is successful, B1 sends to M1 the CONNECT signal together with B_1's signed certificate, Cert_{B1}, a time stamp *T1* and $y_{M1}^{-x_{B1}f(T1,DH(B1,M1))}K \bmod p$, where f(.) is a secure hash function and DH(B1,M1) is the Diffie-Hellman key [10] between M1 and B1. If the signature on the certificate is validated by equation (3) and the timestamp *T1* is recent, M1 retrieves the session key *K* as demonstrated in equation (5).

$$K = y_{B1}^{x_{M1}f(T1,DH(B1,M1))}\, y_{M1}^{-x_{B1}f(T1,DH(B1,M1))}K \bmod p \tag{5}$$

Communication B1 to B2. In the second part, communication is established between the base stations B1 and B2 using messages 3, 4 and 5 as detailed in Figure 3. Base station B1 sends a SETUP signal and its certificate Cert_{B1} to B2. Base station B2 verifies the signature of B1's certificate using equation (3).

On successful verification, B2 replies by sending a CONNECT signal and its certificate. If B1 validates B2's certificate, then B1 sends a new timestamp, *T2*, along with $y_{B2}^{-x_{B1}f(T2,DH(B1,B2))}K \bmod p$ to B2.

3. B1 → B2: SETUP, Cert_{B1}
4. B2 → B1: CONNECT, Cert_{B2}
5. B1 → B2: *T2*, $y_{B2}^{-x_{B1}f(T2,DH(B1,B2))}K \bmod p$

Fig. 3. Messages exchanged between the two base stations B1 and B2

Upon receiving message 5 with a valid timestamp, B2 computes the session key *K* using equation (6).

$$K = y_{B1}^{x_{B2} f(T2, DH(B1,B2))} y_{B2}^{-x_{B1} f(T2, DH(B1,B2))} K \bmod p \tag{6}$$

Communication B2 to M1. In the third part, communication is established between the base station B2 and mobile user M2 using messages 6-8 as detailed in Figure 4.

6. B2→ M2: SETUP, Cert_{B2}
7. M2→ B2: CONNECT, Cert_{M2}
8. B2→ M2: *T3*, $y_{M2}^{-x_{B2} f(T3, DH(B2,M2))} K \bmod p$

Fig. 4. Messages exchanged between the responder M2 and the base station B2

In the third step, B2 sends a SETUP signal to mobile user M_2 along with his certificate Cert_{B2}. The mobile user verifies the certificate and sends its certificate along with the CONNECT signal to B2. The base station verifies M2's certificate and sends a new timestamp T_3 along with $y_{M2}^{-x_{B2} f(T3, DH(B2,M2))} K \bmod p$ to M2. The mobile user verifies the validity of the timestamp and computes the session K using equation (7).

$$K = y_{B2}^{x_{M2} f(T3, DH(M2,B2))} y_{M2}^{-x_{B2} f(T3, DH(M2,B2))} K \bmod p \tag{7}$$

At this stage, the mobile users M1 and M2 have established a common session key that can be used for private communications.

3 Analyzing the LYH Protocol

The authors of the protocol claim that their protocol correctly achieves its goals of authentication and secrecy [6]. Authentication is ensured by the validity of the certification authority's signature on the public key certificates. To bind a public key to its owner, principals establish whether equation (3) holds, where M is replaced by Cert_X as detailed in equation (8). If this is true, then principals will accept that the key contained in the certificate belongs to the principal named in the certificate.

$$y_{CA}^{s+t} s^{h(Cert_x)} = 1 \bmod p \tag{8}$$

Binding principals' IDs to their public keys is crucial for the establishment of the Diffie-Hellman keys, which are used in the distribution of the session key. However, in this paper it will be demonstrated that the employed signature scheme is not secure. It will be shown, that an intruder (also called attacker) can create seemingly correct and valid signatures on any arbitrary message. This is possible without the intruder knowing the Ca's private key x_{CA}. Thus, an intruder can forge the CA's signature on any bogus certificate. Therefore, the security requirements of the protocol are not met.

3.1 Forging Certificates for the LYH Protocol

Consider any arbitrary message M and its signature (s,t) as defined by equations (1) and (2). A principal will consider this signature valid, if it satisfies equation (3). In order to forge CA's signature on message M, the intruder needs to establish the pair (s, t) such that (M, s, t) satisfies equation (3). This can be achieved by selecting s equal to CA's public key y_{CA}. This is equivalent to setting the random value r equal to CA's private key x_{CA} as shown in equation (9) and (10). Note, that it is not required for the attacker to know the value of x_{CA}: To calculate t the attacker simply uses the value 1 for the factor $x_{CA}^{-1}r(= x_{CA}^{-1}x_{CA} = 1)$ and it follows that s is equal to CA's public key y_{CA}.

$$s = y_{CA} \bmod p \tag{9}$$

$$t = -y_{CA} - h(M)x_{CA}^{-1}x_{CA} \bmod q = -y_{CA} - h(M) \bmod q \tag{10}$$

A principal that tries to establish the validity of the signature (s, t) on the message M attempts to ascertain if equation (3) holds by performing the calculations shown in (11). As demonstrated, the forged signature satisfies the equation and any principal will accept the forged signature as a legitimate signature of CA. Forging the signature of CA is possible, as the random number r in t appears only as a factor of a term containing x_{CA}^{-1}.

$$\begin{aligned} &y_{CA}^{s+t}s^{h(M)} = 1 \bmod p \\ &y_{CA}^{y_{CA}-y_{CA}-h(M) \bmod q}\,y_{CA}^{h(M)} \bmod p = 1 \bmod p \\ &y_{CA}^{h(M)-h(M) \bmod q} \bmod p = y_{CA}^{0} \bmod p = 1 \bmod p \end{aligned} \tag{11}$$

The intruder can now easily create arbitrary certificates for malicious activities. As trust among principals is established through certificates, an intruder can create certificates to obtain illegitimate privileges. Below, two attacks are presented, in which it is assumed that the intruder has forged a certificates in the presented way to impersonate principals.

3.2 An Attack to Impersonate a Mobile User

The intruder chooses any private key x_I from GF(p)* with its corresponding public key $y_I = g^{x_I} \bmod p$ and creates a mobile-type certificate $\mathrm{Cert_I}$ that includes the chosen bogus public key $\mathrm{y_I}$ and the identity of the mobile M1. The signature $(\mathrm{s_I},\mathrm{t_I})$ for this bogus certificate is obtained as per equations (9) and (10), i.e. $s = y_{CA} \bmod p$ and $t_I = -y_{CA} - h(Cert_I) \bmod q$. Using this bogus certificate the attacker can initiate a protocol run as M1 to communicate with the mobile user M2 as shown in Figure 5. Neither the mobile user M2 nor any of the base stations B1 or B2 can detect that an illegal certificate is used. Note that the same bogus certificate could be used to take part in the protocol as responder.

In this attack the intruder makes a request to base station B1. The certificate that is presented to the base station appears to be valid. Therefore, B1 is fooled into believing that the intruder is the legitimate mobile user M1. The base station follows the protocol faithfully and generates the session key K, which the intruder I receives in the second message. As the base station B1 used the bogus key in the forged

1. M1(I) → B1: SETUP, Cert$_{\mathrm{I}}$

2. B1 → M1(I): CONNECT, Cert$_{\mathrm{B1}}$, *T1*, $y_I^{-x_{B1} f(T1,DH(B1,M1))} K \bmod p$

3. B1 → B2: SETUP, Cert$_{\mathrm{B1}}$

4. B2 → B1: CONNECT, Cert$_{\mathrm{B2}}$

5. B1 → B2: T_2, $y_{B2}^{-x_{B1} f(T2,DH(B1,B2))} K \bmod p$

6. B2 → M2: SETUP, Cert$_{\mathrm{B2}}$

7. M2 → B2: CONNECT, Cert$_{\mathrm{M2}}$

8. B2 → M2: *T3*, $y_{M2}^{-x_{B2} f(T3,DH(B2,M2))} K \bmod p$

Fig. 5. Attack to impersonate a mobile user

certificate to protect the session key, the intruder is able to retrieve it. The base station B1 continues the protocol run with B2 and eventually the mobile user M2 receives the session key *K*. M2 believes that *K* is shared with a legitimate user when in fact it is shared with the illegitimate intruder.

3.3 An Attack to Impersonate a Base Station

In another scenario the attacker can forge the signature on a base station-like certificate and compromise the privacy of the communication between two mobile users. The attacker can pose as any of the base stations B1 or B2. When the identity of B1 is assumed, then the intruder is in fact generating the session key *K*. This attack, presented in Figure 6, assumes that the intruder selects a public/private key pair x_I,y_I, creates a base station-like certificate Cert$_{\mathrm{I}}$ and signs it as outlined in (9) and (10). Cert$_{\mathrm{I}}$ contains the selected public key and the identity of B1. Alternatively, the intruder can also masquerade as base station B2. In this case, both B1 and M2 are convinced to accept the intruder as a legitimate base station. As a result, the intruder is able to obtain the session key *K* generated by B1. In both variants of the attack, the intruder possesses the session key *K* and can listen on in the communication between the mobile users M$_1$ and M$_2$, whereas they believe that they share a secret session key known only to themselves and legitimate base stations. Consequently, the privacy of the messages exchanged by the mobile users is compromised.

1. M1 → B1(I): SETUP, Cert$_{\mathrm{M1}}$

2. B1(I) → M1: CONNECT, Cert$_{\mathrm{I}}$, *T1*, $y_{M1}^{-x_I f(T1,DH(B1,M1))} K \bmod p$

3. B1 → B2: SETUP, Cert$_{\mathrm{I}}$

4. B2 → B1: CONNECT, Cert$_{\mathrm{B2}}$

5. B1 → B2: *T2*, $y_{B2}^{-x_I f(T2,DH(B1,B2))} K \bmod p$

6. B2 → M2: SETUP, Cert$_{\mathrm{B2}}$

7. M2 → B2: CONNECT, Cert$_{\mathrm{M2}}$

8. B2 → M2: *T3*, $y_{M2}^{-x_{B2} f(T3,DH(B2,M2))} K \bmod p$

Fig. 6. Attack to impersonate a base station

4 Fixing the LYH Protocol

The previous section demonstrated that flaws in the signature scheme of the LYH protocol allow an attacker to use algebraic simplifications to impersonate mobile users or base stations. A secure signature scheme is required to avoid the presented attacks.

One ad-hoc solution to avoid these attacks is to discard the signatures that can be generated by the intruder, i.e. to disallow any signature where *s* equals the certification authority's public key y_{CA}. While this prevents the attacks presented in this paper, the signature scheme would still be subject to other weaknesses [11]. Thus, we propose to employ the Elgamal signature scheme [12], which does not allow the algebraic simplifications required for the presented attacks.

4.1 Using the Elgamal Signature Scheme in the LYH Protocol

The Elgamal signature scheme uses the parameters (*p, q, g*) with the same properties as in the LYH scheme: *p* is a large prime number, *q* is a large prime factor of *p*-1 and $g = m^{(p-1)/q} \bmod p$, where *m* is an integer that satisfies 1 < m < p-1 and $m^{(p-1)/q} \bmod p$ > 1. A certification authority CA selects its secret key x_{CA}, the corresponding public key $y_{CA} = g^{x_{CA}} \bmod p$ and generates a fresh random number *r* for each signature. The signature on a message *M* is then given by the pair (*s, t*) where *s* and *t* are as defined in (12) and (13). If either *s* or *t* evaluates to 0 then the signature algorithm is restarted.

$$s = g^r \bmod p \tag{12}$$

$$t = r^{-1}(h(M) - x_{CA}s) \bmod q \tag{13}$$

Principals will accept a signature, if it satisfies equation (14).

$$g^{h(M)} = y_{CA}^s s^t \bmod p \tag{14}$$

The security of this signature scheme lies in the difficulty of solving the discrete logarithm problem and in the difficulty of finding collisions for the employed hash function h(.).

4.2 Resistance of Fixed Protocol to the Presented Impersonation Attacks

In the impersonation attacks presented in section 3 the intruder sets the value of *s* equal to the CA's public key – thus, implicitly, setting the value of r to CA's private key x_{CA} without explicit knowledge of the value of x_{CA}. In the Elgamal scheme, r^{-1} is a factor of the two terms *h(M)* and ($-x_{CA}s$). Thus, if *s* is chosen to be y_{CA} – which implies that *r* is equal to x_{CA} – then the second term reduces to -*s*. However, the attacker still needs to know the explicit value of *r* to calculate the first term $r^{-1}h(M)$. However, this is equivalent to knowing CA's private key x_{CA}, which is a contradiction to the basic assumption that nobody other then CA knows CA's private key. Simply choosing the values according to equations (9) and (10) also fails. The calculations in (15) demonstrate that this will not satisfy equation (14).

$$\begin{aligned} s &= y_{CA} \bmod p \\ t &= -y_{CA} - h(M) \bmod q \\ g^{h(M)} &= y^s s^t \\ g^{h(M)} &= (g^{x_{CA}})^{g^{xCA}} (g^{x_{CA}})^{-g^{xCA}-h(M)} \\ g^{h(M)} &= (g^{x_{CA}})^{-h(M)} \\ g^{h(M)} &\neq g^{-x_{CA}h(M)} \end{aligned} \tag{15}$$

5 Conclusions

This paper analysed the LYH mobile end-to-end authentication and secrecy protocol. The analysis revealed a serious new flaw in the used signature scheme that allows an intruder to use algebraic simplifications to forge signatures. In particular, it was demonstrated how to forge the signature of the certification authority (CA). Further, two attacks were presented where an attacker can impersonate a mobile user and a base station, respectively. In both attacks, neither any honest mobile user nor base station can detect that a forged certificate has been used. In consequence, the LYH protocol fails to achieve is goals of authentication and secrecy.

To avoid the presented attacks, modifications to the LYH protocol were proposed incorporating the use of the Elgamal signature scheme. It was further shown that the use of the Elgamal scheme protects the protocol against the presented attacks.

Acknowledgements. This work was partially funded by Science Foundation Ireland - Research Frontiers Programme SFI 07/RFP/CMSF 631.

References

1. Ventuneac, M., Dojen, R., Coffey, T.: Automated Verification of Wireless Security Protocols using Layered Proving Trees. WSEAS Transactions on Communications 5(2), 252–258 (2006)
2. Dojen, R., Zhang, F., Coffey, T.: On the Formal Verification of a Cluster Based Key Management Protocol for Wireless Sensor Networks. In: 27th IEEE International Performance Computing and Communications Conference – Workshop of Information and Data Assurance, pp. 499–506 (2008)
3. Dojen, R., Lasc, I., Coffey, T.: Establishing and Fixing a Freshness Flaw in a Key-Distribution and Authentication Protocol. In: IEEE International Conference on Intelligent Computer Communication and Processing, pp. 185–192 (2008)
4. Hwang, R.J., Su, F.F.: A new efficient authentication protocol for Mobile networks. Computer Standards & Interfaces 28(2), 241–252 (2005)
5. Chien, H., Jan, J.: A hybrid authentication protocol for large mobile network. Journal of System Software 67(2), 123–130 (2003)
6. Lee, C.C., Yang, C.C., Hwang, M.S.: A new privacy and authentication protocol for end-to-end mobile users. International Journal of Communication Systems 16(9), 799–808 (2003)

7. Chang, C., Chen, K., Hwang, M.: End-to-End Security Protocol for Mobile Communications with End-User Identification/Authentication. Wireless Personal Communications: An International Journal 28(2), 95–106 (2004)
8. Park, M., Okazaki, N., Baba, Y.: A New User Authentication Protocol for Mobile Terminals in Wireless Network. In: 7th International Conference on Mobile Data Management (MDM 2006), p. 94 (2006)
9. Yi, X., Lam, K.Y.: Hash function based on block cipher. IEE Electronics Letters 33(23), 1938–1940 (1997)
10. Diffie, W., Hellman, M.E.: New Directions in Cryptography. IEEE Transactions on Information Theory, IT 22(6), 644–654 (1976)
11. Chang, C.C., Lee, J.S.: Improvement on an Optimized Protocol for Mobile Network Authentication and Security. In: Hao, Y., Liu, J., Wang, Y.-P., Cheung, Y.-m., Yin, H., Jiao, L., Ma, J., Jiao, Y.-C. (eds.) CIS 2005. LNCS (LNAI), vol. 3802, pp. 538–541. Springer, Heidelberg (2005)
12. Elgamal, T.: A public key cryptosystem and a signature scheme based on discrete logarithms. IEEE Transactions on Information Theory 31(4), 469–472 (1985)