

A Report on Authentication and Security protocols in Wireless Communication

Richa Kumar (2501329)

Submitted as Class Project Report for *Cryptology and Number Theory*

Department of Computer Science

University of Minnesota, Minneapolis MN 55455

December 11, 2001

1 Introduction

With the advent of wireless technology, Personal Communication Systems (PCS) are seeing an increased use of wireless devices. While such devices make it possible to communicate with other systems even while being on the move, there are a number of challenges that need to be met, before wireless communication can be considered as a secure form of communication. This is even more relevant in today's world where the "Internet" is an omni-present entity and commerce and other business related transactions can be carried out on the wire (which ofcourse does not seem to include the case in question, but lets say that by "wire" we mean all forms of network communication). Whether designed for cellular phones, cordless phones, or other user devices, security over the wireless link is still the major concern of the PCSs. Authentication is one of the key issues for security in wireless systems. Presented here is a study of some of the existing authentication protocols that are employed in wireless systems and in some cases weaknesses that may have been identified therein. This is essentially a literary survey.

The report is organized as follows. Section 2 outlines the security requirements in any distributed system (not necessarily wireless). Section 3 describes why security is a bigger challenge in wireless communication. Section 4 is a summary of some of the existing authentication protocols used in wireless systems, with weaknesses highlighted in some cases. The protocols discussed include GSM, DECT, USDC, MSR, Park's protocol as well as the ASPeCT protocol. Finally Sections 5 and 6 talk about the future directions for wireless computing and the conclusions for the report.

2 Security Requirements in a Distributed System

A distributed system consists of local and remote machines connected over a network. Processes running on these machines generally communicate through some form of message passing. This form of communication is an invitation for intruders to attempt to break into the system, since the communication mechanism is asynchronous and is essentially a "listen for a message and then take the appropriate action" system. There are various types of attacks that are possible in a distributed system [8]. These can be broadly classified as follows:

- *Host Compromise*: A host on the network is taken over by a malicious attacker. The degree of compromise ranges from making use of system resources, corrupting system information, to taking complete control of the system.

- *Communication Threats*: Such attacks could fall into one of three categories. Eavesdropping is one form of a network attack, where the attacker simply taps onto the network and gains access to possibly confidential information. However this is considered as a static attack since it does not cause any change in the system. The other two forms of attack are active, as they introduce a change in the system either in the form of adding, deleting or inserting arbitrary messages or replaying old messages.
- *Denial of Service*: This type of attack denies legitimate users of the system access to the system resources by hogging all the available resources.

Based on these potential attack scenarios there are a set of security requirements that are essential in a distributed system. These include *integrity* and *secrecy* of the data as well as *authentication* of all parties that participate in any transaction within the system. Since the focus of this report is on authentication protocols, following is a classification of the types of authentication that are required.

- *message content verification*: verifying the integrity of the contents of the message
- *message origin verification*: verifying that the sender is the same one recorded in the header of the sender
- *identity verification*: verifying that an entity is really who it claims to be

3 Security challenges in a Wireless System

As outlined in the earlier section, security is key in distributed systems. However, it is even more so in wireless communication, as mobile applications have special requirements and vulnerabilities [3].

- *Heterogeneous communication path*: The radio link portion of the communications channel is especially vulnerable to attacks
- *Location privacy*: Information about the location of the mobile station may need to be confidential
- *Computational constraints*: Mobile stations are generally computationally limited as compared to typical communications devices.

In view of the above vulnerabilities that are applicable specifically to wireless and other mobile communications, [2] outlines the *Mobile Security Requirements* with the focus being on authentication protocols.

- *Mutual Authentication of user and network*: Most existing protocols do not address network side authentication.
- *Agreement between user and network on a secret session key with mutual implicit key authentication*: User and network must both agree upon the session key and must be aware of which other entities may know the session key.
- *Mutual key confirmation*: To ensure that the other party does possess the same session key. (This requirement has been pointed out to be debatable by the authors themselves)

- Mutual Assurance of key freshness(mutual key control): This requirement is essential to prevent replay attacks. The mutual key control requirement refers to the inability of a single party to choose the session key. This requirement may not be essential in all protocols.
- Non-repudiation of origin for relevant user data: This is generally provided by digital signatures and is essential when certain information from one party is treated as a commitment and which cannot be reversed.
- Confidentiality of relevant data: Users may want confidentiality of their location and movements, as well as their identities.

4 Authentication protocols

In this section, the first three protocols that are discussed are based on traditional one key cryptosystems. Following this, three PKI based protocols are discussed.

4.1 Conventional Single Key based systems

Certain terms that will be used in the following protocols are described here. Each mobile service provider contains an HLR which is the Home location register and the VLR which is the Visiting location register. These two maintain information about the current and visiting subscribers respectively.

4.1.1 GSM

This protocol has been described in detail in [1, 7]. Presented here is a summary of the protocol and some possible attacks. Each subscriber obtains a secret key, K_i , as well as a unique identification referred to as IMSI, from the authentication authority. The protocol involves the use of three algorithms: A3 and A8, one-way hash functions, A5, encryption/decryption algorithm. When the protocol starts the subscriber sends its IMSI to the VLR. The VLR then contacts the corresponding HLR. The HLR pulls out information about the subscriber from the authentication center and sends the following information back to the VLR: RAND, a random number, RES, the result of $A3(K_i, RAND)$ and a session key K_c . K_c is nothing but the $A8(RAND, K_i)$. VLR then sends RAND to the subscriber, which returns RES1 computed as $A3(K_i, RAND)$. If VLR finds that $RES1 = RES$, then the subscriber is authenticated. At this point the subscriber can calculate K_c using RAND and its key K_i . VLR then generates a temporary identifier TMSI and sends it to the subscriber encrypted by A5 using key K_c . In order to start secure communication between the subscriber and the VLR, both must have the same K_c . Since the subscriber at this point cannot still be sure that the VLR is authentic, there is another round of messages before they start communicating using K_c as the session key for encryption/decryption.

The protocol can be represented as follows:

```
S-->VLR: IMSI
VLR-->HLR: IMSI
HLR: get  $K_i$  based on IMSI
    RES =  $A3(K_i, RAND)$ 
     $K_c$  =  $A8(K_i, RAND)$ 
HLR-->VLR: IMSI,  $K_c$ , RAND, RES
```

```

VLR-->S: RAND
S: RES1 = A3(Ki, RAND)
S-->VLR: RES1
VLR: if RES1 = RES, IMSI authenticated
VLR-->S: A5(Kc, TMSI)
S-->VLR:ACK

```

For each call thereafter, the TMSI is used for identification, in place of IMSI, and a new TMSI is generated every time. Since the IMSI is not transmitted anymore for authentication, this provides better protection of identity information.

The protocol is simple, however, there are some vulnerabilities in this protocol, namely, the TMSIs are generated based on the previous TSMI, therefore a missed synchronization in the TMSIs may require the IMSI to be used to set it up again, wherein the IMSI is sent in plain-text to the VLR, exposing its true identity. Also, there is no mechanism to prevent replay attacks. Once the session key Kc is compromised, by playing back the RAND, and the RES, an intruder can impersonate the VLR since the protocol does not support network authentication.

4.1.2 DECT

In the case of DECT[6, 7], the subscriber and the home station share a secret key, K. The following is the authentication sequence when a subscriber contacts the home station. The home network applies algorithm A11 to obtain KS from the K and a random number RS. Then using KS and another random number RAND_F, it obtains RES. The home station then sends RS and RAND_F to the subscriber, who performs similar calculations (using A11 and A12) to arrive at RES1 which it returns to the home network. If RES = RES1, the subscriber is authenticated. Data can be (optionally) encrypted using a key DCK, which is also derived from KS and RAND_F by both parties. The protocol can be represented as follows:

```

S-->HLR: ID
HLR: KS = A11(K,RS) (where K depends upon ID)
      (DCK, RES) = A12(KS, RAND_F)
HLR-->S: RS, RAND_F
S : KS = A11(K,RS) (where K depends upon ID)
      (DCK, RES1) = A12(KS, RAND_F)
S-->HLR: RES1
HLR: if RES = RES1, ID authenticated

```

In the case of a visiting base network, there are a few options in the way the authentication might be done.

```

OPTION1
HLR-->VLR: K, rest proceeds as above
OPTION2
HLR-->VLR: RS, RAND_F, RES, (DCK), this is similar to DSM
OPTION3
HLR-->VLR: RS, KS, the VLR does the challenge response

```

The process for authenticating the subscriber can also be carried out the other way to authenticate the network. Or the network can be authenticated implicitly by using the optional key (DCK) to encrypt the data, since only the legitimate VLR/HLR can have the DCK.

Although DECT does not provide ID confidentiality, it does provide protection against replay attacks, since the network can be authenticated.

4.1.3 USDC

In USDC[5, 7], the subscriber obtains a secret key from the home network. From this key, K , a secret data SSD is derived using the following sequence, referred to as the SSD Update protocol. In this protocol, both parties mutually authenticate each other.

```
HLR-->S: R1, random number
HLR: SSD = CAVE(K, R1)
S: XSSD = CAVE(K, R1)
    AUT = CAVE(K, R2)
S-->HLR: XSSD, R2
HLR: if XSSD = SSD, subscriber authenticated
    XAUT = CAVE(K, R2)
HLR-->S: XAUT
S: if XAUT = AUT, network authenticated
```

When a subscriber visits another network, the VLR obtains the SSD from the HLR and the subscriber and VLR communicate using the SSD. The VLR can initiate the SSD Unique Challenge Response Procedure which is as follows:

```
VLR: AUT = CAVE(SSD,RANDU)
VLR-->S: RANDU
S: XAUT = CAVE(SSD, RANDU)
S-->VLR: XAUT
VLR: if AUT = XAUT, subscriber authenticated
```

If this fails the SSD Update protocol is initiated to reestablish SSD. Using the SSD and a pin associated with the mobile phone, the subscriber can authenticate himself.

In this protocol, the authentication(or rather the subscriber's secret) is tied to a specific phone, which makes it less flexible. Also, there is no user ID confidentiality, neither is there any way for the subscriber to initiate an authentication of the network being visited. An intruder can convince the network that an SSD update was successful by sending a positive ACK before being authenticated by the network. This would lead to different SSDs being maintained by the legitimate user and the network (for that user), leading to denial of service. To avoid this the ACK should be encrypted using the updated SSD.

4.2 Public Key based Authentication protocols

In most of the above described protocols, one of the important shortcomings is that of the confidentiality of the subscriber's identity. Also, one of the key issues is whether it should be a key distribution or a key agreement protocol. Public key protocols provide stronger security at the expense of greater computation. Even in the case of wired networks, which can cope with greater computation loads, public key cryptography is used simply for exchanging session keys, since it is computationally expensive. Provided here are three public key based authentication protocols for wireless networks.

4.2.1 MSR

This protocol [7] uses both the one-key as well as the public key technique for authentication. First, by Diffie and Hellman's key distribution technique [4], each of the subscriber and the network chooses a secret key and publishes the corresponding public key such that each subscriber-network pair is bound by a secret key K_s , which can be computed by only these two entities. Each entity also obtains a certificate from a certificate authority which vouches for the authenticity of the public key. During the authentication the base station first sends its ID, Certificate and Public key to the mobile station. The mobile station verifies the certificate, then encrypts a random number RAND as $\text{sqr}(\text{RAND}) \bmod N$ where N is the composition of two large primes known only to the base station. It also encrypts, using function f , its own ID, Certificate and Public key with RAND and sends it along with the encrypted RAND to the base station. Now both the parties compute the session key K_c as $f(K_s, \text{RAND})$. To ensure that both parties are using the same key another round of messages is exchanged.

The protocol can be represented as:

```
HLR-->S: IDh, CERTh, Ph
S-->HLR:  $\text{sqr}(\text{RAND}) \bmod N_h, f(\text{RAND}, \text{IDs}, \text{CERTs}, \text{Ps})$ 
S<->HLR: exchange of known message using  $K_c$ 
```

As is the case with public key protocols, the portable device has to do immense amount of calculation as compared to the protocols described earlier. Another issue with this protocol is that it uses two cryptographic techniques (difficulty in factoring prime numbers and difficulty in finding discrete logarithm). If any of these are compromised, the protocol is compromised, rather the strength is the strength of the weaker of the two. If a base station gets compromised, then potentially all mobile stations that have visited it could be compromised. Revoked certificates need to be announced in some way.

4.3 Notation

For the following two protocols we use the following notations:

```
A: mobile user
B: network
 $X_a, X_b$ : private key of A and B resp.
 $Y_a, Y_b$ : public keys of A and B resp.
[all computations take place modulo  $p$ , which is a large prime.
 $g$  is chosen so that the discrete logarithm problem with respect
to it is difficult.
 $R_a, R_b$ : random values generated by A and B resp.
 $\{X\}_k$ : message  $X$  encrypted with key  $k$ 
```

4.3.1 Park's Protocol

This protocol [7] is based on an earlier protocol [9] by Yacobi and Shmueli. For this protocol, we have $Y_a = g^{-X_a}$ and $Y_b = g^{-X_b}$ respectively. The protocol messages in [9] are as follows:

```
B-->A:  $X_b + R_b$ 
A-->B:  $X_a + R_a$ 
```

The session Key $K_{ab} = g^{RaRb}$. This is calculated by A as $K_{ab} = (g^{Xb+Rb}Yb)^{Ra}$ and by B as $K_{ab} = (g^{Xa+Ra}Ya)^{Rb}$.

Park's protocol modifies this to:

B \rightarrow A: g^{Xb+Rb}

A \rightarrow B: Xa+Ra

This reduces the computation load on A which is the mobile station as the asymmetry between the messages is made with the limited computation capability of A in mind. The discussion does not speak about two additional fields in the message exchange as they are not relevant here.

Attacks on this protocol:

- If an attacker C obtains an old session key K_{ab} , for which the protocol messages have been recorded. The following attack can be done:

C \rightarrow A: Xb+Rb

A \rightarrow C(A thinks it's B): Xa+Ra'

The session key computed by A is $K'_{ab} = g^{Ra'+Rb}$. This can easily be computed by B as follows, $K'_{ab} = (g^{Xb+Rb}Yb)^{Ra'-Ra}K_{ab}$. With this done, A does not suspect that C is masquerading as B. In the original protocol, the attack can be mounted from both directions due to the symmetry of the messages. Ability to obtain an old session key is a standard assumption in attack analysis.

- From the message g^{Xb+Rb} , an attacker E can easily obtain obtain $g^{Xb+Rb'}$ from Yb^{-1} and $g^{Rb'}$, where Rb' is a random number chosen by E(since g is public). Once it sends this message to A, the normal sequence of events occur, as if E was B. In this case E does not even have to attack as "man-in-the-middle" but can intrude at its own convenience.

4.3.2 ASPeCT

In this protocol[7], the public keys of the mobile station A, and the base station B, are given by $Y_a = g^{X_a}$ and $Y_b = g^{X_b}$ resp. The other parameters are as follows:

h1, h2, h3 \rightarrow three hash functions

$Sig_a(X)$ \rightarrow A's signature on message X

ACert \rightarrow A's Certificate

BCert \rightarrow B's Certificate

chd \rightarrow charging data

pay \rightarrow payment data

T_B \rightarrow timestamp issued by B

CA \rightarrow certification authority

The protocol goes as follows:

A \rightarrow B: g^{Ra} , CA

B \rightarrow A: $Sig_b(Rb, h2(K_{ab}, Rb, B), chd, T_B, BCert)$

A \rightarrow B: $\{Sig_a(h3(g^{Ra}, g^{Rb}, Rb, chd, T_B, pay), ACert, pay)\}K_{ab}$

The session key is calculated by A and B as follows: A calculates it as $K_{ab} = h1(Rb, Y_b^{Ra})$ and B as $K_{ab} = h1(Rb, ((g^{Ra})^{X_b}))$

The main weakness of this protocol is that the identification of A happens as late as message 3. This might be a consequence of one of the requirements which said that the user's identity be preserved. However, an attacker can take advantage of this, cut off the third message from A to B. In this case A thinks it's carrying out a transaction with B, and B has never even heard of A. This is because, even though B receives the first two messages, it does not know who the messages are coming from unless it receives the third message. The consequence of such an attack is unpredictable, the attacker does not have a direct gain except if it's a rival of B (for service provision).

4.4 Security of the Authentication/Certification Center

In the case of symmetric key encryption protocols, as discussed earlier, the secret key of each entity needs to be stored with an Authentication center. This authentication center needs to be online in order to provide the necessary information to the communicating parties. In the case of public key based protocols, either an offline Certification authority issues certificates verifying the authenticity of an identity's public key or a an online Certificate authority is required if the identity needs to be kept secret. In either case, there is a central authority whose security now becomes a key issue in the whole system. If the authentication/certification center gets compromised, there's no way the system can sustain any form of security what so ever.

One way to mitigate the risk is to break down the system into smaller domains. Each domain has its own authentication/certification center. Some form of cross-domain communication requires to be established in such a case. This limits the damage due to a compromised authentication/certification center to a single domain. Another way to minimize the risk is to distribute the capability among various centers, which will require all the key/certificate centers to be compromised before it affects any entities within the system.

Another issue in the case of Certificates is Revocation lists. Each certification authority maintains a list of certificates that have been revoked. For a party verifying a certificate, it is essential that the revocation list be checked for the absence of the certificate being verified before the certificate is trusted.

5 Future Direction for Wireless Computing

In this report, a number of authentication protocols have been outlined both using symmetric key encryption as well as public key encryption. Both approaches have their advantages and disadvantages as mentioned before. It seems unanimously agreed upon that symmetric session keys will be employed. However it is debatable whether public key encryption should be used for authentication and session key setup. Many wired systems use this technique, yet find it computationally expensive. In a wireless network, where the computation power is limited, this is an even bigger concern. Yet one of the important requirements of a wireless system is that the location and identity of the mobile station be kept confidential which is better achieved using public key systems.

Also, it can be noted that in most cases subscribers roam between multiple base stations within the home area, and infrequently visit foreign domains. In such a case, a secret key could be used between the subscriber and the home area, which is set up using public key encryption, only the first time the subscriber communicates with the network. On subsequent sessions the same shared key could be used to generate the session specific key. This may lead to a not so expensive amortized

cost for the system and may prove to be worth its while to use public key encryption and benefit from some of its features.

6 Conclusions

The purpose of this report was to outline some of the constraints, vulnerabilities and security requirements in a wireless network and to study few selected examples of authentication protocols that are being/have been proposed. From the study conducted, there is one underlying feeling that comes to light, which is that "there's a price for everything" As mentioned in the previous section, different protocols have different weaknesses. As hardware and software technology progresses, applications for data, audio, images or their combination will be incorporated (and have been to a large extent). New applications will be built and the security requirements for each may differ. Its worth tailoring the authentication and security requirements to specific applications. Many new protocols are targeting the end-to-end spectrum for security. One of the key constraints for the existing protocols is the computation power of a wireless device. However it is not unthinkable that this may change with time, giving way to extremely powerful authentication mechanisms.

The protocols described here were chosen because they were most accessible in terms of material available on each of these protocols. There is no implication here that these are the best protocols existing within each of the categories listed. There are various other protocols for authentication in wireless networks like Varadharajan-Mu, Aziz-Diffie, CDPD, which make for interesting reading and comparison among different protocols.

References

- [1] ETSI/TC Recommendation GSM 03.20. "Security Related Network Function, version 3.3.2", January 1991.
- [2] C. Boyd and D. Park. "Public key protocols for wireless communications", 1998.
- [3] Colin Boyd and Anish Mathuria. "Key Establishment Protocols for Secure Mobile Communications: A Selective Survey". In *Australasian Conference on Information Security and Privacy*, pages 344–355, 1998.
- [4] Whitfield Diffie and Martin E. Hellman. "New Directions in Cryptography". *IEEE Transactions on Information Theory*, IT-22(6):644–654, 1976.
- [5] EIA/TIA-IS-54-B.
- [6] ETS 300 175-7 ETSI, October 1992.
- [7] Hung-Yu Lin, Lein Harn, and Vijay Kumar. "Authentication Protocols in Wireless Communications".
- [8] Woo and Lam. "Authentication for Distributed Systems, from Computer, January, 1992". In *William Stallings, Practical Cryptography for Data Internetworks, IEEE Computer Society Press, 1996*. 1996.
- [9] Y. Yacobi and Z. Shmueli. "On Key Distribution Systems". In *Advances in Cryptology - Crypto'89*, pages 344–355, 1989.