Cryptanalysis of Park's Authentication Protocol in Wireless Mobile Communication Systems

Alberto Peinado Dominguez
Deptartment of Ingenieria de Comunicaciones
E. T. S. I. Telecomunicacion, Universidad de Malaga
Campus de Teatinos, 29071 Malaga, Spain (Email: apeinado@ic.uma.es)

(Received Oct. 16, 2005; revised and accepted Nov. 25, 2005)

Abstract

In 2004, C. Park proposed an authentication protocol to provide user anonymity and untraceability in wireless mobile communication systems. The real user identities are hidden and randomized by means of error-correcting codes. In this work, it is shown that Park's protocol does not provide anonymity and untraceability. More precisely, the users real identities can be obtained easily by an eavesdropper. Furthermore, the protocol is not secure since the session key established in the authentication phase can also be obtained, breaking the confidentiality of the radio link.

Keywords: Anonymity, authentication, mobile network, network security, untraceability

1 Introduction

Several security protocols have been proposed to provide anonymity in mobile communication systems [1, 2, 5, 6]. Most of them are based on public-key cryptosystems, which are not the most efficient from the implementation point of view in a mobile environment. Note that mobile terminals present low computational power and low memory size and that the mobile communications are characterized by a low bandwidth and higher channel error rate.

Taking these facts in mind, Park proposed in [4] an authentication protocol providing anonymity and untraceability using a combination of symmetric-key cryptography and error-correcting codes with high correction capability and easy decoding process. This protocol also minimizes the number of messages interchanged between the user and the network.

In the next section, some notation is introduced and the two versions the of Park's protocol are described. Section 3 deals with the cryptanalysis of each version, concluding that none of them are secure. It is also shown that anonymity and untraceability are not provided. Section 4 discusses on several performance aspects.

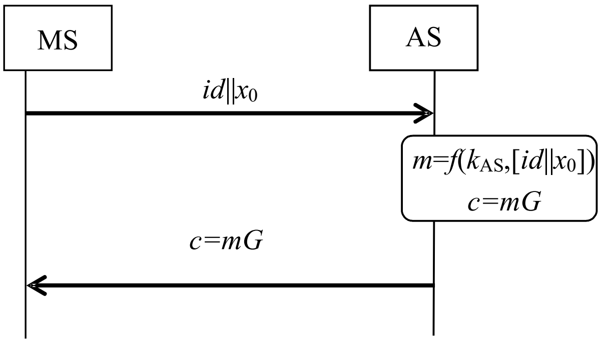

Figure 1: Park's protocol set-up phase

2 Park's Authentication Protocol

Following the same notation introduced in [4], the identity of a mobile subscriber or mobile user (MS) is denoted by *id*, and a symmetric-key encryption function is denoted by $f()$. The encryption of a message m with the secret key k using the function $f()$ is denoted by $f(k,m)$. Finally, the system is represented by the authentication server (AS). In this sense, it is important to note that the MS does not communicate directly with the AS. This communication takes place through certain entities such as the visiting location register (VLR) in the case of GSM system. The implication of this simplification is discussed later in Section 4.

Let h be a pseudorandom generator where the output is $2n$ bits length for an input of n bits length. The output $h(x)$ for any n-bit input x, is divided in two halves. The right half is denoted by $h_0(x)$, while the left half is denoted by $h_1(x)$.

The protocol can be divided in two phases: the set-up phase and the authentication phase, both described below.

Set-up phase:
This phase is performed at the subscription to the service (see Figure 1). Then, the MS computes the following

tokens x_i and session keys k_i:

$$\begin{aligned} x_{i-1} &= h_0(x_i), \quad \text{for } i = s, s-1, \cdots, 1 \\ k_i &= h_1(x_i), \quad \text{for } i = s, s-1, \cdots, 1 \end{aligned}$$

The MS sends the root authentication token x_0 to the AS. The AS generates a symmetric-key certificate m of the MS identity enciphering the identity id and the authentication token x_0, using $f(\cdot)$ with the secret key k_{AS}, that is

$$m = f(k_{AS}, [id||x_0]), \quad \text{for } i = s, s-1, \cdots, 1$$

where || means "the concatenation of". The secret key k_{AS} is only known by AS, hence it is not shared with the users.

The AS chooses a binary linear block code (N, K, D), where N is the bit-length of the codewords, K is the bit-length of the messages to be encoded, and D is the minimum distance. This code must provide an efficient decoding algorithm for a maximum number t of errors, where $t = (D-1)/2$. Hence, if a codeword $c = (c_1, c_2, \cdots, c_N)$ is sent to the channel, and the word $r = c + e$ is received where $e = (e_1, e_2, \cdots, e_N)$ is the error vector, the original codeword c can be recovered if the Hamming weight of e is less than or equal to t.

Thus, the AS generates the generation matrix G of the code to encode the symmetric-key certificate m, as $c = mG$. Next, the encoded certificate c is sent to the MS.

Although it is not clear in [4], it has to be assumed that every step in the set-up phase is performed in a secure way, by means of any encryption mechanism. Otherwise, the encoded certificate c is made public.

Authentication phase:
When the MS wants to access to the system at session i, the following steps must be performed, including only one message between MS and AS.

Step 1. MS⟶ AS: $c + e^{(i)}$

The MS generates the error vector $e^{(i)}$ with Hamming weight t, using a public algorithm (see Algorithm 1 in [4]) to transform $[i||x_i]$ into a vector of weight t. This is the way this protocol provides untraceability. Note that at session j, the error vector will be different.

Step 2. The AS decodes the received word $c + e^{(i)}$ using the corresponding decoding algorithm, obtaining the error vector and the symmetric-key certificate m of the user. AS deciphers m to obtain the identity id and x_0 to verify the token x_i in the error vector. In order to get x_i from the error vector another public algorithm (see Algorithm 2 in [4]) has to be applied.

2.1 Improved Park's Protocol

Several potential weaknesses described by Park, also in [4], motivated him to propose the following improvement affecting only to the authentication phase.

The modification consists mainly to use a cyclic error-correcting code, instead of a generic linear block code. Applying the cyclic property of this code, the MS rotates cyclically his encoded certificate a number r_i of times. The resulting word is also a codeword denoted by $c(r_i)$. Then, the error vector is obtain by Algorithm 1 in [4] from the concatenation string $id||x_i||r_i$.

Hence, the MS sends $c(r_i) + e^{(i)}$ to the AS. The AS decodes the received word, obtaining the error vector $e^{(i)}$, and by means of Algorithm 2 in [4] obtaining $id||x_i||r_i$. Then, AS inverts the r_i times rotation to obtain c. Hence, AS proceeds as in the original version.

3 Cryptanalysis

In this section, cryptanalysis is applied to the two versions of Park's protocol. The first version corresponds to the original protocol proposed in [4], while the second version of the protocol corresponds to the improvement also mentioned in [4] by Park himself to overcome some potential weaknesses.

3.1 Cryptanalysis of Park's Protocol

In [4], Park describes some potential weaknesses of his protocol, in such a way that he also proposes an improvement to avoid the attacks [4], Section 4.1. In this section, the potential weaknesses pointed in [4] are considered describing situations in which the term "potential" turns out to be "real" (weaknesses).

Traceability of users:
Park points that if an eavesdropper knows some parameter of the code used in the protocol, such as the minimum distance$D$ or the maximum number t of errors to be corrected, then he can attempt to trace a particular user in the following way.

The eavesdropper observes $(c + e)$ at some session i, and $(c' + e')$ at session j. If these observed values belong to the same user, then $c = c'$ and the Hamming weight of $(c + e) + (c' + e')$ is less than $2t + 1$, since the Hamming weight of the error vectors is t.

This potential weakness becomes a real weakness when a legal user of the system wants to trace another user. In this case, the eavesdropper is inside the mobile network and, of course, he knows t. Hence, untraceability of users cannot be provided.

Applying the same argument, any entity of the system may trace the users, since it knows D and t.

Known encoded certificate c:
Parks also points in [4] that if an encoded certificate c is exposed to an eavesdropper, the user owner of c can always be traced since the Hamming weight of $(c' + e') + c$ is less than or equal to t, when $c = c'$.

As in the previous case, the system knows the encoded certificate of every user. Hence, the real identity of the users could only be hidden from the rest of users but not from the system.

Actually, it is not clear in [4] how the encoded certificate c is sent to the user. No encrypted transmission is described. Anyway, taking into account the one-time password approach used in this protocol, previous off-line transaction could be considered to set-up the system. Otherwise, if an on-line process is used to send in plaintext the certificate c to the user, everyone will know the certificate, and user anonymity could not be provided.

Recovering the generator matrix G: Assuming that no one can read the encoded certificate c sent to the user, i.e., the encoded certificate are sent enciphered from the system, and taking into account the parameters of the error-correcting code proposed by Park in [4] ($N = 255$, $K = 131$, $D = 37$, $t = 18$), the generator matrix could be computed from 131 encoded certificates at least, if all of them are linearly independent.

It is probable that some of the 131 certificates collected can be expressed as a combination of the others. Hence, the attacker may attempt to collect more encoded certificates, but also may attempt to construct a partial parity-check matrix H to partially decode the values $(c + e)$.

In a real environment, where the number of users subscribed to the mobile network is very high, it would not be unreasonable to consider the possibility that a hundred users confabulate to obtained the generator or the parity-check matrix.

In the case that the encoded certificate was sent in clear, it is easy to get hundred of them directly from the radio link.

Anyway, it is important to consider the possibility that an eavesdropper can recover the matrix G or H. In such a case, the eavesdropper will have access to the content of error vectors, and thus to the token x_i used to compute the session key.

3.2 Cryptanalysis of Improved Park's Protocol

The previous section shows that the original Park's protocol does not provide anonymity and untraceability. Moreover, in the case that an eavesdropper computes the generator matrix G, the system turns out to be insecure. In this section, the improved version of the protocol is analysed, leading us to define an algorithm to break completely the security.

As it is described in Section 2.1, the improvement of Park's protocol is based on the utilization of a cyclic error-correcting code, instead of a generic linear code. This modification tries to avoid the attacks described in the previous section as *Traceability of users* and *Known encoded certificate c*. The foundation of this protocol resides on the cyclic property of this kind of codes. Hence, instead of masking the certificate c with the error vector e, the encoded certificate is first rotated cyclically r_i times and then masked with the error vector e.

This kind of codes are determined by a generator polynomial $g(x)$ of degree $(N - K)$, in such a way that every codeword $c = (c_1, c_2, \cdots, c_N)$ expressed as polynomial $c(x) = c_1 + c_2x + c_3x^2 \cdots + c_Nx^{N-1}$ is a multiple of $g(x)$, where $g(x)$ divides $(x^N + 1)$ (*cf.* [3]).

Recovery of generator polynomial: Consider that a legal user is the eavesdropper. Hence, he knows his own encoded certificate c, and the parameters t and N of the cyclic code. If a BCH or Reed-Solomon code is used, as it is proposed in [4], then the parameter K is completely determined by t and N [3]. Thus, the eavesdropper may attempt to obtain the generator polynomial $g(x)$ in the following way.

Step 1. The eavesdropper computes the greatest common divisor of $c(x)$ and $(x^N + 1)$, that is

$$g'(x) = gcd(c(x), (x^N + 1))$$

Step 2. If the degree of $g'(x)$ is $(N - K)$, then $g'(x)$ is the generator polynomial.

Step 3. If the degree of $g'(x)$ is greater than $(N - K)$, then the eavesdropper chooses a factor $a(x)$ of degree $(N - K)$ dividing $g'(x)$.

Step 4. The eavesdropper computes the syndrome of $c+e$ for known values of the error vector e, to obtain c. If the operation success, then $a(x)$ is the generator polynomial. Otherwise, the user selects a different factor of $g'(x)$ and applies Step 4 again.

Once the eavesdropper has computed $g(x)$, he can decoded every value $(c + e)$. In this way, the real identity of every user can be obtained, breaking the anonymity and untraceability. Furthermore, the error vector e is also obtained by means of syndrome computing, allowing the attacker to access the parameters i, x_i and r_i, thus breaking the security of the system, since the session key is $k_i = h_1(x_i)$.

In order to improve the efficiency of this cryptanalysis, a confabulation attack can be considered. That is, if two or more users share their encoded certificates then the generator polynomial can be obtained faster. Note that $g(x)$ divides every codeword.

4 Performance Analysis

Besides the security problem exposed in the previous section and the impossibility to provide anonymity and untraceability, several aspects have to be discussed regarding the performance efficiency.

First, the mobile network model considered in [4] hides important details about the network architecture of this kind of systems. For example, the AS is present in every network, and it is the only entity with the ability to authenticate legal users, but the MS does not communicate directly with AS. Instead of that, the MS uses the radio-link path to communicate with the base station (BS in GSM or node-B in UMTS). The base station through the internal fixed network communicates with the visiting location register (VLR). This entity has the responsibility to request the authentication information to the AS of the home network (HN), which the MS has subscribed to. Hence, in order to preserve the anonymity and confidentiality, some mechanism could be proposed to transfer the sensitive information from AS to VLR. No matter where the VLR is, in the home network or in a visiting network, the VLR always requests this information to the AS.

In this new scenario, two possibilities can be considered. On one hand, the authentication would be performed in the own AS, and hence every time the MS wants to access the system the whole path from MS to AS has to be used. This fact implies a traffic increment, and of course, is less efficient than the current authentication protocols in current mobile networks (GSM, UMTS). Note that in GSM and UMTS, the AS generates authentication information, but the users are authenticated in the VLR.

On the other hand, if the authentication is performed in the VLR, as in GSM and UMTS, then the VLR has to know the secret key k_{AS} in order to verify the identity of users MS. Hence, the protocol does not work because every VLR would have the capability to certify the identity of subscriber. Moreover, the generation matrix G would have to be distributed to every VLR, which is neither efficient nor secure.

5 Conclusions

It has been proved that the Park's protocol does not provide anonymity and untraceability, because any legal user acting as an eavesdropper can easily trace the other users. Furthermore, the improved version of the protocol does not provide anonymity and untraceability and is not secure because any legal user or a confabulation of two or more, can easily compute the generator polynomial of the error-correcting code, allowing them to obtain the identities of each user and the session keys to be employed later in the (un)confidential communications.

6 Acknowledgements

This work has been supported by MEC-Spain under grant SEG2004-02418, "Evaluacion de protocolos y algoritmos de seguridad en sistemas de informacion" ("*Security protocols and algorithms evaluation in information systems*").

References

[1] A. M. Barbancho and A. Peinado, "Cryptanalysis of anonymous channel protocol for large-scale area in wireless communications," *Computer Networks*, vol. 43 , pp. 777-785, 2003.

[2] W. D. Lin and J. K. Jan, "A wireless-based authentication and anonymous channels for large scale area," in *Proceedings Sixth Symposium on Computers and Commnications (ISCC'01)*, pp. 36-41, July 3-5, 2001.

[3] W. C. Huffman and V. Pless, *Fundamentals of Error-Correcting Codes*, Cambridge University Press, 2003.

[4] C. Park, "Authentication protocol providing user anonymity and untraceability in wireless mobile communication systems," *Computer Networks*, vol. 44, pp. 319-333, 2004.

[5] A. Peinado, "Privacy and auhentication protocol providing anonymous channels in GSM," *Computer Communications*, vol. 27, pp. 1709-1715, 2004.

[6] C. Yang, Y. Tang, R. Wang and H. Yang, "A secure and efficient authentication protocol for anonymous channel in wireless communications," *Applied Mathematics and Computation*, In press, 2005.

Alberto Peinado Dominguez was born in Malaga (Spain). He received the Ing. degree in telecommunications engineering from the University of Malaga in 1993, and the PhD degree in computer science from the Polytechnic University of Madrid, Spain, in 1997.

From 1995 to 1998, he was with the National Spanish Council for Scientific Research (CSIC), Madrid, Spain, where his research interests were in cryptography and network security. Since 1998, he has been with the Department of Ingenieria de Comunicaciones at the University of Malaga as an Assistant Professor and then as an Associate Professor. His research interests include cryptography, mobile communications, CDMA codes, smart cards and watermarking.