An improvement on secure communication in personal communication system

Nawal A. El-Fishawy and Albert M. Tadros
Electronics and Electrical Communications Eng. Dept., Faculty of Electronic Eng., Menouf, Egypt

We present two attacks to the authentication protocol that has been proposed in the paper entitled "an approach to secure communication in PCS " in the May 2001 issue of the IEEE Vehicular Technology Conference. We show that the attacks are feasible and propose correction that makes the protocol more robust and resistant against the presented attacks. The corrected protocol yields higher security compared to both the original protocol and GSM authentication protocol. Moreover, the authentication procedures for the different handoff protocols have been presented and discussed. The most appropriate handoff protocol that matches the authentication protocol discussed here is recommended. We also discuss different encryption algorithms and suggest the most suitable one that yields higher throughput and lower delay time to be used in the corrected authentication protocol.

في الجزء الأول من هذا البحث نوضح العيوب الموجودة في بروتوكول توثيق يوجد في بحث بعنوان"مدخل إلى تأمين الاتصال في أنظمة الاتصال الشخصي " وتم نشر هذا البحث في مايو ٢٠٠١ في مؤتمر IEEE vehicular technology conference المنعقد في جزيرة رودس باليونان وأهم هذه العيوب هو افتراض أن الاتصال آمن بين مسجل المشتركين الأساسيين HLR و مسجل المشتركين الزائرين VLR مما يؤدى إلى تبادل المعلومات بدون تشفير بينهما و هذا الافتراض ليس دائما صحيحا و بالأخص في الجيل القادم من أنظمة الاتصال المتحركة وكنتيجة لهذا العيب تم توضيح اختراقين للبروتوكول السابق ذكره . الاختراق الأول يلعب فيه المخترق دور VLR وفى الاختراق الثاني يستخدم المخترق معلومات تم الحصول عليها من الاختراق الأول و بواسطتها يستطيع المخترق أن يستخدم النظام مدعيا انه أحد المستخدمين الموثق لهم استخدام النظام. في الجزء الثاني من البحث تم تصحيح العيب الموجود في البرتوكول السابق عن طريق تشفير المعلومات المتبادلة بين HLR,VLR وتم تقييم البروتوكول المصحح من حيث الوقت المطلوب لاتمام البروتوكول. في الجزء الثالث من البحث ناقشنا البروتوكولات المختلفة لعملية نقل المكالمة وهى التحكم في نقل المكالمة عن طريق الشبكة و نقل المكالمة بمساعدة الراديو المتحرك و التحكم في نقل المكالمة عن طريق الراديو المتحرك وتم توضيح أداء بروتوكول التوثيق المصحح مع كل منهم وتم اختيار بروتوكول التحكم في نقل المكالمة عن طريق الراديو المتحرك ليكون هو بروتوكول نقل المكالمة مع بروتوكول التوثيق المصحح وذلك نتيجة لان الوقت الذي يستغرقه هذا البروتوكول قليل بالمقارنة مع الأنواع الأخرى كما انه يقدم لا مركزية في عملية نقل المكالمة. في الجزء الرابع تم عمل مقارنة بين استخدام أنواع مختلفة من خوارزميات التشفير وهى DES,TDES,RC5 وعدم استخدام التشفير من حيث الوقت المطلوب لنقل المعلومات وتم التوصل إلى أن عدم استخدام التشفير يعطى اقل وقت يليه استخدام خوارزم RC5 حيث أن RC5 يعطى وقت اقل من DES, TDES بمقدار ٥٠،٦ مللي ثانية و ٩٩٢،٩ مللي ثانية على الترتيب حيث تمت المقارنة باستخدام معالج من النوع AMD وسرعته ٥٠٠ميجا هيرتز. و بناءاً على المقارنة السابقة تم اختيار RC5 ليستخدم مع بروتوكول التوثيق المصحح ليعطى احسن أداء.

Keywords: Mobile communication, Authentication, Encryption, Secret key, Public key

1. Introduction

Wireless communications, as an open medium, are susceptible to a lot of problems concering security. Many algorithms [1-5], have been proposed to overcome the drawbacks of the authentication process proposed by the Global System for Mobile communication (GSM)

In [1] an approach based on Data Encryption Standard (DES) has been proposed for handling the security issues in PCS. The mentioned approach yields a higher security compared to GSM authentication protocol at the expense of using more computational time at call setup, which is still negligible. The main idea behind that approach was:

1. To provide strong subscriber confidentiality.
2. DES is used in the cellular phones, which is easy to implement in hardware, gives fast encryption /decryption time and minimizes any overhead in the ciphertext.

3. The session key is generated at the mobile user (MU) site.
4. Not only the network challenges the MU, but also the MU challenges the network.

The authors showed that the mobility of the PCS presents new problems related to authentication ensuring that service is not obtained fraudulently and privacy of information about the PCS user's location. They assumed that the mobile user (MU) is out of reach of Home Location Register (HLR), and is visiting a new location area. Therefore, Visitor Location Register (VLR) will be the one "talking" to the mobile user. So, the mobile user should proceed through that protocol as shown in fig. 1.

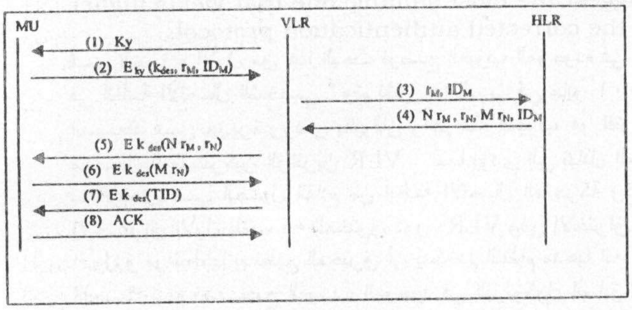
MU
VLR
HLR
(1) Ky
(2) $E_{ky}(k_{des}, r_M, ID_M)$
(3) r_M, ID_M
(4) $N r_M$, r_N, $M r_N$, ID_M
(5) $E k_{des}(N r_M, r_N)$
(6) $E k_{des}(M r_N)$
(7) $E k_{des}(TID)$
(8) ACK

Fig. 1. Authentication procedures for the old algorithm.

The protocol is described in more detail as follows:
1. In flow 1, the VLR broadcasts its public key Ky every minute.
2. In flow 2, the MU generates a random number r_M and a session key k_{des} and sends both together with its identity ID_M encrypted with Ky to VLR.
3. When the VLR receives the message in flow 2 it decrypts it and forwards r_M, ID_M to the HLR.
4. When the HLR receives the message in flow 3; it searches its database for the user ID_M and obtains its secret key k_{MH} and computes the response N r_M to the mobile challenge r_M. Then it generates the network challenge r_N to MU and the response M r_N.
5. VLR sends to MU both of N r_M and r_N encrypted with k_{des}.
6. When the MU receives the message in flow 5, it decrypts it using k_{des} and checks whether N r_M is the correct one, and if yes, it computes the response M r_N to the network challenge r_N using his secret key k_{MH} and sends it encrypted with k_{des} to the VLR.
7. When the VLR receives the message in flow 6, it decrypts by using k_{des} and compares M r_N received from the MU with that received from the HLR. If a match occurs in this step the VLR generates a temporary identity TID for the MU and sends it encrypted with k_{des} to MU.
8. MU responds with ACK.

Unfortunately, the protocol has serious flaws that permit various attacks. In the following section, we present two attacks against the protocol and we propose corrections to avoid their occurrence. Section 3 will discuss the authentication procedures for the different handoff protocols. Section 4 will evaluate different encryption algorithms. The paper will be concluded in section 5.

2. Attacks

The main flow of the protocol arises when the authors assumed a complete trust between HLR and VLR, which led to the transmission of data in clear between them. Such assumption is not always true especially in the next generation of mobile communications, which provides a personal communication user with global roaming service [2] and so, the radio link between the HLR and VLR will be more susceptible to eavesdropping.

2.1. Attack 1

The protocol is subjected to the so called man-in- the middle attack as shown in fig. 2, where the attacker plays the role of VLR as follows:
1. The attacker impersonates that it is VLR and propagates its public key Ky.
2. When the MU receives the public key Ky, which belongs to the attacker, he will think that this public key is received from a new VLR. So, the MU will generate a session key k_{des} and a random number r_M and sends them encrypted with the public key of the attacker.
3. When the attacker receives the message in flow 2, it decrypts it by using its secret key and forwards r_M and ID_M to HLR. At the end of this step the attacker will have the session key k_{des} which is generated by the MU.

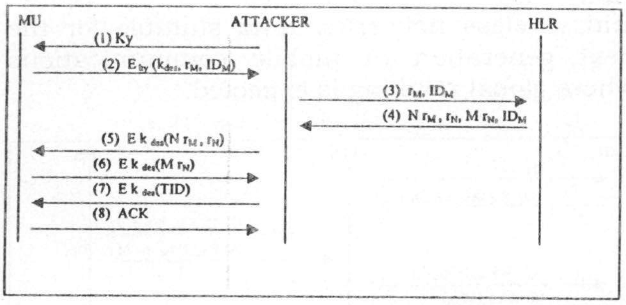
MU
ATTACKER
HLR
(1) Ky
(2) E_{ky} (k_{des}, r_M, ID_M)
(3) r_M, ID_M
(4) N r_M , r_N, M r_N, ID_M
(5) E k $_{des}$(N r_M , r_N)
(6) E k $_{des}$(M r_N)
(7) E k $_{des}$(TID)
(8) ACK

Fig. 2. Authentication procedures with the attacker impersonating VLR.

4. HLR proceeds as in step 4 in the original protocol.
5. When the attacker receives the message in flow 4, it sends N r_M and r_N encrypted with k_{des}, which is obtained from step 2, to the MU.
6. When the MU receives the message in flow 5, it decrypts the message and checks if N r_M is correct or not (sure it will be the correct one since HLR is the one which generates N r_M and not the attacker) if yes he sends E k $_{des}$(M r_N) to the attacker.
7. When the attacker receives the message in flow 6, it generates a temporary identity TID and sends it to the MU.
8. The MU sends an ACK to the attacker.

At the end of this attack, the attacker will now have k $_{des}$, r_N and M r_N.

After the authentication process ends, the user will deal with the attacker instead of the VLR of the visited network in his subsequent requests for service. So, the attacker will be able to eavesdrop all the calls of the user MU with the identity ID_M..

In this attack, the intruder is able, with regard to the network technology, to play the role of the visited network and to make the roaming user sends messages to it instead of the visited network. Practically, impersonating the VLR is not impossible. There are commercially available devices called "IMSI catchers" its functionality is very similar to that needed by the intruder in this attack. 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 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 rely communication between the mobile phone and the base station and stay unnoticed. [3]

2.2. Attack 2

To perform attack 2, attack 1 must be done first from which attacker 1obtains real information for k $_{des}$ and ID_M^* which concern authorized user. Attacker 2 may be attacker 1 himself, or a fraudent user who received these information form attacker 1.

Now, the attacker begins an authentication process impersonating that he is the user MU as follows (see fig. 3):
1. In flow 2 the attacker sends the identity ID_M^* and k_{des}^* of the user MU obtained from attack 1 instead of his identity and impersonating that k_{des}^* is the session key he generates.
2. In flow 3 the VLR will forward r_M and ID_M^* to HLR.
3. HLR will think that the attacker is the user with identity ID_M. So, HLR will sign r_M with the secret key of the user with identity ID_M and generates a challenge (random number r_N) and the response (M r_N) to that challenge.
4. In flow 4, the attacker will replace this challenge and its response with that one obtained from the first attack (r_N^*, M r_N^*). VLR will receive the message in flow 4, it will think that r_N^* and M r_N^* were sent from HLR. So, it will challenge the attacker with r_N^*
5. Since the attacker has r_N^* and M r_N^* he will respond to the VLR challenge by sending M r_N^* encrypted with k $_{des}$ (the attacker already knows k $_{des}$ from attack 1).
6. In flow 6 VLR will receive the response to its challenge and it compares it with that one received from HLR in flow 4 (sure there is a matching in this step because in reality the attacker is the one who sends the challenge to VLR and the response to VLR).
7. VLR will generate a temporary identity (TID) and send it to the attacker.

At the end of attack 2, the attacker will have k $_{des}$ and TID of the user with identity ID_M and he begins to use the system impersonating that he is a legal user with identity ID_M.

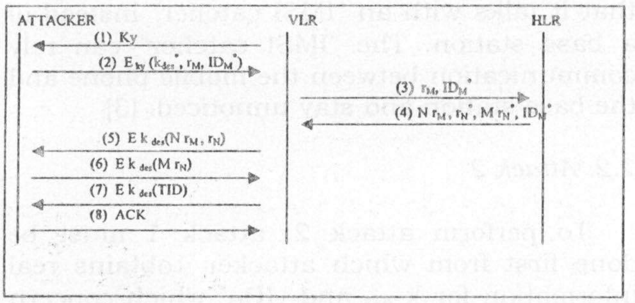
ATTACKER
VLR
HLR
(1) Ky
(2) $E_{ky}(k_{des}^{*}, r_M, ID_M^{*})$
(3) r_M, ID_M^{*}
(4) $N r_M, r_N^{*}, M r_N^{*}, ID_N^{*}$
(5) $E k_{des}(N r_M, r_N)$
(6) $E k_{des}(M r_N)$
(7) $E k_{des}(TID)$
(8) ACK

Fig. 3. Authentication procedures with the attacker impersonating MU.

2.3. Corrections

The main drawback of the suggested protocol is that the authors assumed that the link between HLR and VLR is secure so, the data are sent in clear between them. To correct this problem, we assume that:

- There is a secret key K_n shared only between the visited network and the home network, and this key will be used to cipher all the data exchanged between the HLR and VLR.
- To prevent impersonating VLR, the identity of the visited network ID_V must be sent to the home network, so that HLR be sure of the validity and reality of the VLR.

The corrected protocol will be as follows: Flow 1 and flow 2 will behave as they are. In flow3 (see fig. 4), the VLR will send (r_M, ID_M) encrypted with K_n, accompanied by ID_V. On receiving these information, HLR will authenticate the VLR by making sure of ID_V. From ID_V, HLR searches its database to find the key K_n to decrypt $E_{kn}(r_M, ID_M)$. in flow 4, HLR will encrypt ($N r_M$, r_N, $M r_N$, ID_M) with K_n ,where any fraudulent VLR could not be able to decrypt it. The rest of the protocol will be as it is.

On executing these correction, we avoid attack 1 because the attacker will not be able to obtain k_n and thus he can not exchange messages with HLR and thus he can not perform attack 1. On inhibiting the occurrence of attack 1, attack 2 is avoided.

With the proposed corrections we gain a more secure and robust protocol. The corrected protocol could be used both in wired and wireless networks. It is suitable for the next generation of mobile communications where global roaming is expected.

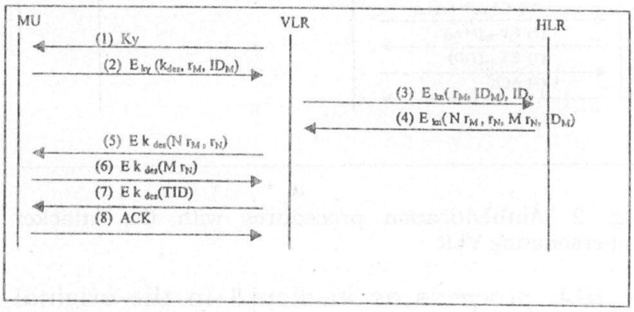
MU
VLR
HLR
(1) Ky
(2) $E_{ky}(k_{des}, r_M, ID_M)$
(3) $E_{kn}(r_M, ID_M), ID_v$
(4) $E_{kn}(N r_M, r_N, M r_N, ID_M)$
(5) $E k_{des}(N r_M, r_N)$
(6) $E k_{des}(M r_N)$
(7) $E k_{des}(TID)$
(8) ACK

Fig. 4. The authentication procedures for the corrections of attack 1 and 2.

2.4. Evaluation of the corrected protocol

The authors in [1] evaluate the time needed to perform their proposed protocol according to some assumptions, which are stated below:

1. The propagation time between MU and VLR is 0.25 msec. This corresponds to a 50 km distance, which takes into consideration the distance from MU to the nearest Base Station (BS) and the distance from the BS to VLR.
2. The propagation time between VLR and HLR is 2.5 msec that corresponds to a 500 km distance.
3. Transmission rate is 13.3 kbps and is the same at all the links.
4. Encryption / decryption time for 114 bits for GSM is 4.6 msec [4], while for DES it is 1.17 msec, assuming 13 Mhz chip at the MU end [5].
5. ACK size is 1 bit.
6. All the other variables, encrypted or not are 64 bits, except Ky which is 192 bits.
7. Assume that the computational time at HLR and VLR is negligible compared to the computational time spent at the MU site.

According to the previous assumptions, the time needed for setting up a connection by GSM protocol is 49.26 msec while it increases to 86.125 msec for the protocol in [1], which means an increase of approximately 36.865 msec is needed to perform their protocol. This time is still within the acceptable range of setting up a connection as stated in [1].

The time needed by the corrected protocol is 90.905 msec. The reason for this increase is due to additional term ID_v in flow number 3. The encryption between the HLR and VLR adds no time to the whole protocol because the computational time at HLR and VLR is negligible as stated in assumption (7). The time difference between the corrected protocol and GSM protocol is 41.645 msec which is still within the acceptable range of setting up a connection.

3. Handoff

Handoff is the process of changing the radio channel with a new one either because of moving the mobile user from one cell to another cell (intercell handoff) or because of deteriorating the channel quality below certain level within the same cell (intracell handoff). Form the execution phase of intercell handoff process there are three basic types of handoff protocols [6]:

1. Network-Controlled Handoff (NCHO): in the NCHO protocol, the network makes a handoff decision based on measurements of the Received Signal Strenghts (RSSs)of the MS at a number of BSs. As in [6] the overall delay of this protocol can be of the order of 5-10 sec. This type of handoff is not suitable for a rapidly changing environment and a high density of users due to the associated delay.
2. Mobile-Assisted Handoff (MAHO): an MAHO protocol distributes the handoff decision process. The MS makes measurement, and the MSC makes decisions. The overall delay of this protocol can be of the order of 1 sec [6].
3. Mobile-Controlled Handoff (MCHO): in MCHO the MS is completely in control of the handoff process. This type of handoff has a short reaction time (on the order of 0.1 sec) and is suitable for microcellular systems. MCHO is the highest degree of handoff decentralization. Some of the advantages of handoff decentralization are that handoff decision can be made fast, and the MSC does not have to make handoff decisions for every mobile which is a very difficult task for the MSC for high-capacity microcellular systems.

In the following sections we will discuss the authentication procedures of the handoff process in view of the three addressed protocols.

3.1. Authentication procedures for NCHO and MAHO protocols

Consider that the MU is making a call in cell 1, which is served by base station 1(BS1). During the call the MU is moving toward another cell (assume it is cell 2) that is served by another base station (assume it is BS2). At this point a handoff process must be intiated to continue the call successfully. Based on the measurements of the RSSs, the Mobile Switching Center (MSC) will determine that the MU needs a handoff, so it sends a handoff request to BS1 telling it that the user will be served by BS2. According to the request of the MSC, BS1 will send k_{des} and TID encrypted with the public key of BS2 (Ky_2) to the MSC. Since k_{des} and TID are encrypted with the public key of BS2 (ky_2), the only one who can decrypt this message is BS2 itself with its secret key. The MSC will forward E_{ky2} (k_{des}, TID) to BS2. Now BS2 will decrypt this message using its secret key and locates a new channel for the MU. Next BS2 will encrypt the new channel information in cell 2 with k_{des} so, the only one who decrypt these information is the user MU itself. The user will now begin to be served by BS2, see fig. 5.

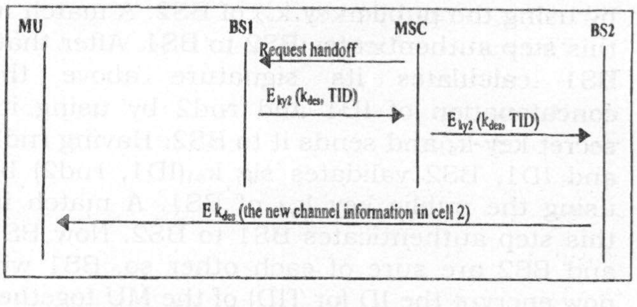
MU
BS1
MSC
BS2
Request handoff
E_{ky2} (k_{des}, TID)
E_{ky2} (k_{des}, TID)
E k_{des} (the new channel information in cell 2)

Fig. 5. Authentication procedures for NCHO and MAHO protocols.

The above behavior is suitable for two types of handoff protocols: NCHO and MAHO but it is not suitable for MCHO.

3.2. Authentication procedures for MCHO protocol

Assume that the MU initiated a call in cell 1, which is served by BS1, and he is now moving towards cell 2, which is served by BS2. As the MU moves toward cell 2, the signal strength received from BS1 decreases, whereas the signal strength received from BS2 increases. There is a time where the signal strength received from BS2 is higher than that received from BS1. At this time, a handoff is initiated. The MU sends a message to BS1 requesting a handoff to BS2. So, an authentication process starts between BS1 and BS2 as shown in fig. 6.

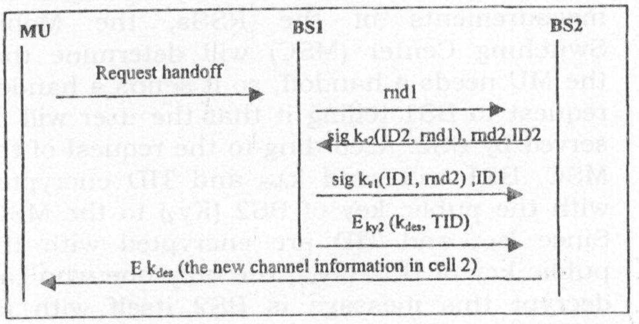
MU
BS1
BS2
Request handoff
rnd1
sig k_{s2}(ID2, rnd1), rnd2,ID2
sig k_{s1}(ID1, rnd2) ,ID1
E_{ky2} (k_{des}, TID)
E k_{des} (the new channel information in cell 2)

Fig. 6. Authentication procedures for MCHO.

BS1 sends a random number rnd1 to BS2 to challenge it. BS2 calculates its signature above the concatenation of ID2 and rnd1 by using its secret key $k_{s2}$and sends it together with random number rnd2 to BS1. Having rnd1 and ID2, BS1 validates sig k_{s2} (ID2, rnd1) by using the public key k_{y2} of BS2. A match in this step authenticates BS2 to BS1. After that, BS1 calculates its signature above the concatenation of ID1 and rnd2 by using its secret key $k_{s1}$and sends it to BS2. Having rnd2 and ID1, BS2 validates sig k_{s1}(ID1, rnd2) by using the public key k_{y1} of BS1. A match in this step authenticates BS1 to BS2. Now BS1 and BS2 are sure of each other so, BS1 will now encrypt the ID (or TID) of the MU together with the session key k_{des}, by using the public key k_{y2} of BS2 and sends them to BS2. If BS2 successfully locates a channel for the MU, it will send the channel information encrypted by using k_{des} to MU; otherwise the call is dropped. Only the MU using k_{des} will be able to decrypt the information and switch to the new channel.

The corrected protocol in this article can follow any of the mentioned protocols for excuting the handoff process. The point of delay and the handoff decentralization are tow important factors which recommend MCHO to be more appropriate for the next generation of mobile communications where the users will be allowed to have global access to the network.

4. Evaluation of encryption algorithms

In this section, we evaluate different encryption algorithms to decide the most suitable one to be applied on the corrected protocol. These algorithms are data encryption standard (DES), triplet (TDES) and RC5. We could not include the encryption algorithm for GSM because of lack of more details concerning A3, A5 and A8. The evaluation points were restricted on estimating the delay time and throughput.

4.1. Delay time estimation

On calculating the delay time taken to transfer data, we consider that channel conditions are good (i.e.; the probability of error is zero). The delay time without applying any encryption algorithm is given by [1]:

$$D_{wot} = \frac{8(D + \frac{D}{1368}H)}{R_b}, \qquad (1)$$

while the delay time to transfer data where security is applied to protect the communication is :

$$D_{wel} = \frac{8(D + \frac{D}{1368}H)}{R_b} + \frac{8(D + \frac{D}{1368}H)T}{n}, \qquad (2)$$

where D denotes the data size in bits, R_b is the channel transmission rate in bps, H denotes the overhead, T is the time in seconds taken to encrypt n bits. We assume a packet size of 1422B, where 1368 are data. The following table 1 summarizes the results obtained for the delay time without applying any encryption algorithm (D_{wot}) and with applying different encryption algorithms (D_{wel}), at different values of the channel rate and data size of 136.8 KB.

Table 1
The delay time

Transmission rate in bps	D_{wot} in sec	D_{wel} for DES in sec	D_{wel} for riple-DES(TDES) in sec	D_{wel} for RC5 in sec
5.00E+03	227.52	227.7153828	228.6576	227.6647596
1.00E+04	113.76	113.9553828	114.8976	113.9047596
1.50E+04	75.840	76.0353828	76.9776	75.9847596
2.00E+04	56.880	57.0753828	58.0176	57.0247596
2.50E+04	45.504	45.6993828	46.6416	45.6487596
3.00E+04	37.920	38.1153828	39.0576	38.0647596
3.50E+04	32.503	32.69823994	33.64045714	32.64761674
4.00E+04	28.440	28.6353828	29.5776	28.5847596
4.50E+04	25.280	25.4753828	26.4176	25.4247596
5.00E+04	22.752	22.9473828	23.8896	22.8967596

The values of the table is obtained on a processor AMD of 500 MHZ which is much faster than the processor in the mobile handset, so the difference between the obtained values is small but we consider these values as a guide for comparison. It is clear from the table that the delay time without applying any encryption algorithms is the smallest values. At any value of the transmission rate, the delay time with RC5 is smaller than that with DES and TDES by values of 50.6 ms and 992.9 ms respectively. This means that the channel transmission rate has no effect on the delay time and the encryption decryption rate has the major effect on increasing or decreasing the delay. In [1], the author proved that DES is better than GSM, and now we have got to the conclusion that RC5 is better than TDES, DES and GSM as well, where it offers the least encryption /decryption delay time.

4.2. Throughput

For calculating the throughput, we assume that the channel conditions are not good and the BER has a certain value. This means that the packet received in error needs to be retransmitted several times until received correctly. The average number of times a packet needs to be retransmitted given the packet length is N and the channel BER is P is given by [3]:

$$AV_{nr} = \frac{1}{(1-p)^N + Np(1-p)^{N-1}} . \tag{3}$$

The values of throughput without applying encryption algorithm (Th_{wot}) and with applying encryption algorithm (Th_{wel}) are given by:

$$Th_{wot} = \frac{8D}{D_{wot}R_b} , \tag{4}$$

$$Th_{wel} = \frac{8D}{D_{wel}R_b} , \tag{5}$$

where D_{wot} is as found in eq. (1) except that it is multiplied by eq. (3) and D_{wel} is simply D_{wot} found above plus the second term in eq. (2).

Table 2 illustrates the different values of throughput with and without applying encryption algorithms at different values of BER, data size of 14.22 KB and channel rate of 13.3 Kbps.

The result of table 2 is in agreement with that of table 1, where RC5 achieves higher throughput with relative to TDES and DES, when the channel conditions are good (i.e;BER has small values). The value of TH_{wot} is still

Table 2
Throughput

BER	Th_{wot}	Th_{wel} for DES	Th_{wel} for TDES	Th_{wel} for RC5
1.00E-05	9.6193E-01	9.5974E-01	9.4930E-01	9.6030E-01
1.00E-04	9.5318E-01	9.5103E-01	9.4078E-01	9.5158E-01
1.00E-03	5.6200E-01	5.6126E-01	5.5767E-01	5.6145E-01
1.00E-02	9.1818E-06	9.1818E-06	9.1818E-06	9.1818E-06

gaining the highest throughput. As the channel conditions deteriorate, the difference in throughputs for all the approaches becomes narrower. The reason is simple, the time to transmit the packet correctly is much larger than the time taken for encryption, decryption. From the results of table 1 and 2, we recommend RC5 to be the most suitable encryption algorithm to be applied on the corrected protocol. In addition to the proved values of low delay and high throughput of RC5, it has other advantages such as:

1. Suitable for hardware and software.
2. Fast because the basic operations work on full words of data at a time.
3. Adaptable to processors of different word lengths.
4. Variable number of rounds.
5. Variable length key.
6. Simple and is easy to implement.
7. Low memory requirement that makes RC5 suitable for smart cards and other devices with restricted memory.
8. High security Data-independent rotations: RC5 incorporates rotations whose amount is data dependent. This appears to strengthen the algorithm against cryptanalysis.

5. Conclusions

In this paper, we presented two attacks to a published protocol for secure communication in PCS [1]. The first attack represents VLR (man in the middle), and the second attack is done by a fraudent user who makes use of the confidential information obtained by the first attacker. Commercial existence of the first attack is feasible by the "IMSI catchers" whose functionality is similar to that of the VLR. Inserting a secret key shared only between the visited network and the home network to cipher the data will overcome the drawbacks of this protocol. Evaluating the corrected protocol in terms of the time needed to perform it has been done. The authentication procedures for the different handoff protocols (NCHO, MAHO and MCHO) are presented and discussed. The more appropriate handoff algorithm which suits the authentication protocol is recommended. Different encryption algorithms are evaluated concerning the delay time and the throughput and the results proved that RC5 achieves the highest throughput and the lowest delay.

References

[1] M. Xu and S. Upadhyaya, "An approach to secure communication in PCS," IEEE vehicular technology conference, May (2001).

[2] S. Suzuki and K. Nakada, "An authentication technique based on distributed security management for the global mobility network," IEEE journal on selected areas in communications, Vol. 15 (8), pp. 1608-1617 (1997).

[3] L. Buttayn, C. Gbaguidi, S. Staamann and U. Wilhelm, "Extensions to an authentication technique proposed for the global mobility network," IEEE transactions on communications, Vol. 48 (3), pp. 373-376 (2000).

[4] ETSI/TS, Recommendation for GSM 03.20, Security related functions, version 3.3.2., January (1991).

[5] http://www.samsung.com/

[6] N. Tripathi, N. Reed and H. Vanlandingham, "Handoff in cellular systems, " IEEE personal communications, December, pp. 26-37 (1998).

[7] National Bureau of Standards, Data Encryption Standard, Federal Information Processing Standards Pub., Vol.46 (1977).

[8] W. Diffie and M.E. Hellman, "New directions in cryptography, "IEEE transactions information theory, Vol. IT-22, pp. 644-654 (1976).

[9] M. Beller, L. Yung, and Y. Yacobi, "Privacy and authentication on a portable communication system," IEEE on selected areas in communications, Vol. 11 (6), pp. 821-829 (1993).

[10] W. Stallings, "Cryptography and network security, " Prentice Hall (1999).

[11] W. Stallings, "Network security and essentials, " Prentice Hall (2000).

Received June 24, 2002
Accepted November 23, 2002