UNCONDITIONALLY SECURE AUTHENTICATION SCHEMES AND PRACTICAL AND THEORETICAL CONSEQUENCES

Yvo Desmedt [1]

Dept. of Computer Science[2], University of New Mexico
Albuquerque, New Mexico, U.S.A.

current address:
Katholieke Universiteit Leuven, ESAT01
Kardinaal Mercierlaan, 94, B-3030 Heverlee, Belgium

Abstract

The Vernam scheme protects the privacy unconditionally, but is completely insecure to protect the authenticity of a message. Schemes will be discussed in this paper that protect the authenticity unconditionally. The definition of unconditional security is defined. Stream cipher authentication schemes are proposed. The consequences on information protection using RSA and DES are discussed.

1. Introduction

We will start here by looking how some authors discuss the protection of authenticity in a conventional cryptosystem. The definitions given for unconditional security will be overviewed. We will conclude that both subject matters are mostly presented oversimplified. This will be explained by checking their definitions using the Vernam scheme (see Section 2). We will conclude that unconditionally authentication protection is not discussed. Hereto we define unconditional security from a point of view of authenticity and we also redefine the old definition of an unconditionally secure cryptosystem (see Section 3). We will then build up an unconditionally secure authentication system (see Section 4). The practical and theoretical consequences will be presented (see Section 5).

Some authors, e.g., Denning [4], (pp. 10) pretend that "*in symmetric* (conventional) *cryptosystems ... secrecy cannot be separated from authenticity*", and that "*if users cannot access E_A and D_A, then both the secrecy and authenticity of A's data is assured*". However, today it is well-known that one can authenticate the message (and the sender) without protecting the privacy (of the whole message (see the previous last paragraph of Section 4.3 and Section 6.1)). The NBS authentication method [11] (pp. 24) is an example of this. It is also known that some modes as e.g., the E.C.B. mode in DES, are insecure to protect the authenticity of a *long* message. As Diffie and Hellman [8] (pp. 646) said: "*A cryptographic system intended to guarantee privacy will not, in general, prevent this latter*

[1] NFWO aangesteld navorser, is currently sponsored by the National Science Foundation of Belgium.

[2] This research was done while the author was Visiting Assistant Professor at the University of New Mexico.

H.C. Williams (Ed.): Advances in Cryptology - CRYPTO '85, LNCS 218, pp. 42-55, 1986.

form of mischief". One can conclude that Denning's ideas, earlier cited, are at least oversimplified. One can wonder if *each cryptosystem which protects the privacy can also authenticate long messages if one uses modes* (e.g. CFB or CBC).

The term unconditionally secure is misleading. One can have the impression that it covers more. When one says that the one-time pad cryptosystem is unconditionally secure, one can think that one can never attack the privacy or authenticity. The definition or use of unconditional security only deals today with privacy protection (see e.g. [4], [16], [17], [22]). As Simmons remarked in [22], Shannon's models [21] were only concerned with secrecy. One can wonder if *a scheme which protects the privacy unconditionally does the same for the authenticity.*

We will now answer both questions by discussing the Vernam (or one-time pad) cryptosystem [21], [24].

Important remark

If in the following sections we say that an intruder can inject a fraudulent message with a probability p_1 or that an active eavesdropper can modify a message with a probability p_2, we mean the following: in one on $1/p_1$ respectively $1/p_2$ cases the system used by the receiver will not automatically detect the injection respectively the modification. Automatically here means that the system uses a different way to detect modifications in the message than using the redundancy in the language. If the reader does not agree with this restriction we remark that the worst case is a message without redundancy. In order to be able to deal with such messages previous restriction is evident.

2. The Vernam scheme and authentication

As known, the Vernam scheme protects the privacy unconditionally [21]. Let us shortly explain how it works. Let $M = (m_1, m_2, \ldots, m_n)$ be the plaintext message, where m_1 is the first bit of the message, m_2 the second, and so on. Then the ciphertext $C = (c_1, c_2, \ldots, c_n)$ is the bitwise exor of M and the key K, or $c_i = m_i \oplus k_i$. The key K is really random and only used once. The decryption operation is similar: $m_i = c_i \oplus k_i$.

It is now easy to understand that the probability to *inject* a fraudulent bit is $1/2$ (see important remark in Section 1). If the active eavesdropper wants that the receiver receives a bit 1, he injects a bit (it does not matter if it is a zero or a one that he injects). Because the key bit is in one on two cases (in average) a 0 (and otherwise a 1), the receiver receives a 1 in one on two cases. One can remark that the effect of this attack is not important, because if one wants that the receiver accepts a concrete fraudulent message of 100 bits, the probability to succeed by injecting a message is only $1/2^{100}$. However, in some cases the damage caused by the injecting of one bit may be important. That one bit may tell you to delete or not to delete a file, to transfer the money or not.

An even more serious attack is to *modify* the ciphertext. It is easy to understand that an active eavesdropper can modify a bit of the plaintext with a probability 1. Hereto he has only to complement the ciphertext bit. In the case the active eavesdropper does not know the plaintext, the effect of his action will probably be a not understandable message (sometimes called "garbage"). However, for terrorists that does not matter, it is enough to sabotage. If the active eavesdropper *knows the plaintext, he can easily modify it as he wants* (if the fraudulent message is not longer than the original one)!

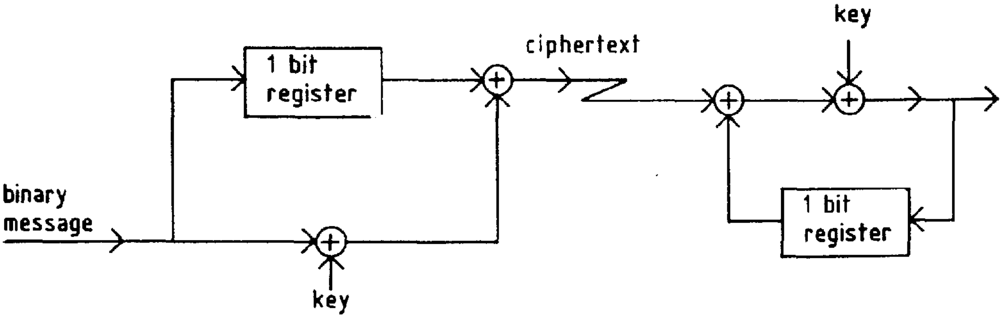

Figure 1: The Vernam scheme used in a CBC mode

We can conclude that the *Vernam scheme can not protect the authenticity. A similar remark was made by Feistel [9] (pp. 19 - 20), without coming up with an unconditionally secure authentication scheme. However, one could remark that, to protect the authenticity of a message, one has to use a mode and some redundancy at the end of the message* (e.g. 64 zeros). In order to show that this does not help, we first suppose that the sender uses the CBC mode in feedforward and the receiver uses it in feedback (in order to have a large error propagation). The plaintext is followed by 64 zeros as authenticator. Using the Vernam scheme for the encryption and decryption devices we obtain Figure 1. The active eavesdropper can modify each bit of the message as he wants, without affecting the authenticator! Suppose he wants to modify m_i. His attack is successful, if he complements bits c_i and c_{i+1} (where c_i is the i^{th} transmitted ciphertext bit). If the active eavesdropper wants to modify more bits, he has to superimpose previous attack. If the sender and receiver do not want to protect the privacy and use hereto a similar scheme as the NBS one [11], the attack is similar, because the Vernam scheme encrypts bit after bit (its block length is one bit). The reader can easily extend the attack for other modes as CFB and OFB (see [11]).

We have shown that the Vernam scheme can not protect the authenticity. Schemes which protect the privacy do not necessarily protect the authenticity, even if some modes are used. This result can be extended to schemes which are similar (e.g. the Vignere scheme).

We will now try to come up with cryptosystems which protect the authenticity unconditionally. Evidently, we have first to define what means an unconditionally secure authentication scheme.

3. New definitions for unconditional security

Definition 1: *A cryptosystem protects the* privacy unconditionally *if, no matter how much ciphertext is intercepted, there is not enough information in the ciphertext to come* *up with a unique solution (plaintext), but many exist. Ideal and perfect cryptosystems [21] fall under this definition.*

Remark that:

- the fact of unconditional security can not be affected by using more (e.g. an infinite amount of) computertime.
- if the appropriate security rules are satisfied (e.g. secrecy of the key) the cryptosystem can *never* be broken!

We now want to come up with a similar definition for the unconditional protection of the authenticity (of message and sender). Let us start from the last remark. So we would say that a cryptosystem protects unconditionally the authenticity if it can never be broken. However, then no cryptosystem at all would satisfy this definition. The process of authentication is probabilistic. If an active eavesdropper tries long enough (e.g. some centuries) he will be able to inject a fraudulent message or modify one. This follows from the fact that messages have finite lengths. So a better definition, based on the first remark, will now be given.

Definition 2: *A cryptosystem protects the* authenticity unconditionally with a security level P *if, the probabilities that an intruder can inject a fraudulent message or that an active eavesdropper can modify a message are less or equal than* $1/P$*, independently how much computertime is used. If one of these probabilities is equal to one we say that the system is insecure (to protect the authenticity).*

As a consequence of this definition the Vernam scheme is insecure related to authenticity. A scheme is considered to be *insecure for practical purposes if the security level is "too" small.* We will not discuss wat means "too" small, the reader is refered to [5] and [6].

The effect of birthday attacks [1], [14] (pp. 127) is not discussed in this paper.

Let us now build cryptosystems which satisfy last definition.

4. Building unconditionally secure authentication schemes

Based on the analysis of the Vernam scheme we will first try to come up with cryptosystems which protect the authenticity unconditionally. Because the schemes which will be proposed in Section 4.1, do not satisfy partly or totally the definition, we will in Section 4.2 come up with a secure one. In Section 4.3 a more practical version will be discussed. Finally by combining other schemes with the ones discussed here, one can still improve the practical aspects (see Section 4.4).

4.1. Trials

We have seen that the Vernam scheme is bit oriented. In the first proposal we use an authenticator for each bit. Each bit of the message is followed by a *fixed pattern* (e.g. 64 zeros). All these bits are then encrypted using the Vernam scheme. We now give a formal way to describe this scheme. If the plaintext $M = (m_1, m_2, \ldots, m_n)$ then the input for the Vernam scheme is $A = (a_1^1, a_1^2, \ldots, a_1^q, a_2^1, a_2^2, \ldots, a_2^q, \ldots, a_n^1, a_n^2, \ldots, a_n^q)$, where a_i^j are bits such that for all i $(1 \leq i \leq n)$ we have $(a_i^1, a_i^2, \ldots, a_i^q) = (m_i, 0, \ldots, 0)$ in the case the fixed pattern is $(0, 0, \ldots, 0)$ and $q - 1 =$ length of the fixed pattern. The ciphertext is then $C = (c_1^1, c_1^2, \dots, c_1^q, c_2^1, c_2^2, \dots, c_2^q, \dots, c_n^1, c_n^2, \dots, c_n^q)$ where $c_i^j = a_i^j \oplus k_i^j$ and the key is $K = (k_1^1, k_1^2, \dots, k_1^q, k_2^1, k_2^2, \dots, k_2^q, \dots, k_n^1, k_n^2, \dots, k_n^q)$, where k_i^j is a bit (for all i and j, where $1 \leq i \leq n$ and $1 \leq j \leq q$). Remark that we have an expansion of the ciphertext and the key with a factor q (e.g. × 65). This scheme is however insecure (form the point of view of authenticity) because an active eavesdropper can complement a ciphertext bit, corresponding with an information bit, without modifying the ciphertext bits corresponding with that authenticator (in other words complementing c_i^1, without modifying any c_i^j where j satisfies $2 \leq j \leq q$). This implies that the probability that an active eavesdropper can modify a message is evidently one.

In another proposal the information bit is placed at random in the authenticator. The mathematical description of this scheme is similar, except that $(a_i^1, a_i^2, \dots, a_i^q) = (0, 0, \dots, 0, m_i, 0, \dots, 0, 0)$, where the bit m_i appears on a random location. In order that the receiver could verify that location he has to know the random value (otherwise an active eavesdropper can easily modify a message bit 0 into a message bit 1). This random value is a part of the *authentication key. We define the authentication key as the key which is used to protect the authenticity, and the privacy key as the key which is used to protect the privacy.* The length of the authentication key is $n \cdot \lceil \log_2 q \rceil$. The length of the *privacy key* is $q \cdot n$ and the expansion of the ciphertext is q. The probability that an active eavesdropper can modify a message bit is however high and is $1/q$. Evidently, if the active eavesdropper knows q and modifies one bit of the q bits randomly, the probability to modify the bit corresponding with a certain m_i is $1/q$. In order to obtain a *practical acceptable security level* (e.g. 2^{64}) *the expansion of the ciphertext has to be enormous,* because the security level only increases linearly with increasing expansion of the ciphertext.

The last scheme will be modified to come up with an unconditionally secure authentication scheme, for which the security level increases exponentially with linearly increasing ciphertext and keyexpansion.

4.2. A secure scheme

By studying previous proposal we see that the security level is q, and there exist q different $(a_i^1, a_i^2, \dots, a_i^q)$ for each $m_i = 1$. In general, we can have $2^q - 1$ different $(a_i^1, a_i^2, \dots, a_i^q) \neq (0, 0, \dots, 0)$. Hereto we use an authentication key $H = (h_1^1, h_1^2, \dots, h_1^q, h_2^1, h_2^2, \dots, h_2^q, \dots, h_n^1, h_n^2, \dots, h_n^q)$ random such that for all $i : (h_i^1, h_i^2, \dots, h_i^q) \neq (0, 0, \dots, 0)$ and the key H is secret and used only once by sender and receiver (similar as in the Vernam scheme). The scheme differs only from previous one in the fact that for each bit $m_i : (a_i^1, a_i^2, \dots, a_i^q) = m_i \cdot (h_i^1, h_i^2, \dots, h_i^q)$. In other words if $m_i = 0$ then $(a_i^1, a_i^2, \dots, a_i^q) = (0, 0, \dots, 0)$, otherwise $(a_i^1, a_i^2, \dots, a_i^q) = (h_i^1, h_i^2, \dots, h_i^q)$.

Remark that in the discussed scheme the ciphertext expansion is still q. The length of the authentication key is $q \cdot n$, and the same for the privacy key. So the complete key used in this scheme is $2q$ times longer than in the Vernam scheme.

The discussed scheme protects the authenticity unconditionally with a security level 2^{q-1}. An intruder can inject a bit 0 with a probability $1/2^q$, because in order to inject a 0 (after that the legitimate $(i-1)^{\text{th}}$ bit was sent) he has to guess the correct $(k_i^1, k_i^2, \dots, k_i^q)$ and because these bits are really random he has only a probability $1/2^q$ to succeed. A similar reasoning is true for the case he wants to inject a 1 (remark that $(h_i^1 \oplus k_i^1, h_i^2 \oplus k_i^2, \dots, h_i^q \oplus k_i^q) \neq (k_i^1, k_i^2, \dots, k_i^q)$ for all i). So he can inject a bit with a probability $1/2^{q-1}$. An active eavesdropper can modify a bit with a probability $1/(2^q - 1)$, because he has to guess correctly $(h_i^1, h_i^2, \ldots, h_i^q)$. Remark that it is "hard" for an active eavesdropper to mix the bits of the plaintext, or to retransmit them, because the key is really random and only used once. This follows easily from previous discussion. Remark that previous discussions remain valid if we consider known plaintext attacks, as long as the privacy and authentication keys are secret.

In order to better understand the discussed scheme let us wonder what happens if we do not protect the privacy (or if $(k_1^1, k_1^2, \ldots, k_1^q, k_2^1, k_2^2, \ldots, k_2^q, \ldots, k_n^1, k_n^2, \ldots, k_n^q) = (0, 0, \ldots, 0)$). The reader can easily verify that the injection of a bit 0 or the modification of a plaintext bit 1 into a 0 is easy. However, it is "hard" to inject a bit 1 or to modify a plaintext bit 0 into a 1. We can conclude that the protection of the privacy is necessary in order to protect, with this scheme, the authenticity. However, one can easily imagine situations in which the protection of a bit 1 is more crucial than the protection of a bit 0 [5], [6]. In E.F.T. for example, the plaintext can be a bit 1 if the transaction is authorized, a 0 in the other cases. Following our definition of unconditional security we do not consider the scheme secure under these circumstances. One can wonder if the key H has to be secret. The answer is evidently yes, otherwise an active eavesdropper can easily modify the message.

Without discussing if this scheme is practical (see Section 5.1) we can remark that such a large text expansion is impractical. It slows down the communication and makes it much more expensive! For these reasons a more practical scheme will now be presented.

4.3. A more practical scheme

An unconditionally secure authentication system which is based on the one discussed in previous section, will be presented. For previous scheme, remark that if an intruder wants to inject two bits the probability to succeed is $1/2^{2q-2}$. In general it is for m bits $1/2^{mq-m}$, because each bit has its own authenticator. A similar reasoning is true for the modification of m bits. *In this section we will only use a authenticator for the whole message.* That idea will also solve the speed and cost problem of previous scheme. Nevertheless the new scheme is also unconditionally secure.

In this scheme we send the message M enciphered with the Vernam scheme, followed by an authenticator of q bits. So the ciphertext is $C = (c_1, c_2, \ldots, c_n, c_{n+1}^1, c_{n+1}^2, \ldots, c_{n+1}^q)$ such that for $i < n+1$ we have $c_i = m_i \oplus k_i$, where c_i, m_i and k_i are bits. For $i = n+1$ we have $c_{n+1}^j = r^j \oplus k_{n+1}^j$ for each j such that $1 \leq j \leq q$, where c_{n+1}^j, r^j and k_{n+1}^j are bits. $R = (r^1, r^2, \ldots, r^q)$ is the authenticator and $K' = (k_1, k_2, \ldots, k_n, k_{n+1}^1, k_{n+1}^2, \ldots, k_{n+1}^q)$ is the privacy key. Remark that c_i^j has no sense here if $i < n+1$, similar for the key and for the message. The register R is build up iteratively when each message bit m_i is sent, using the authenticator key $H = (h_1^1, h_1^2, \ldots, h_1^q, h_2^1, h_2^2, \ldots, h_2^q, \ldots, h_n^1, h_n^2, \ldots, h_n^q)$. The contents of R in the begin is 0, then $(r^1, r^2, \ldots, r^q) := (r^1 \oplus m_i h_i^1, r^2 \oplus m_i h_i^2, \ldots, r^q \oplus m_i h_i^q)$ for each m_i, where $1 \leq i \leq n$. In other words at the end

$$r^j = \bigoplus_{i=1}^{n} m_i h_i^j \qquad \text{for each } j \quad (1 \leq j \leq q). \tag{1}$$

This scheme is unconditionally secure with a security level $(2^q - 1)$. An intruder can inject a message that will be accepted with a probability $1/2^q$, because only one on 2^q messages give that authenticator R. An active eavesdropper can only modify one bit of the message with a probability $1/(2^q - 1)$ to succeed, because he has to guess the correct $(h_i^1, h_i^2, \ldots, h_i^q)$ if he wants to modify m_i. If he wants to modify more bits, he has to guess the correct modification (see Eqn. 1), the probability to succeed is only about $1/2^q$.

In order to better understand this scheme, let us wonder if we need to protect the privacy. If we do not protect the privacy (or if K' is equal to zero), then the previous reasoning remains valid, except that it is easy to inject the message $(0, 0, \ldots, 0)$ or to modify a message into that zero message. Indeed, for a zero message the authenticator R is zero. However a very simple protocol can overcome the transmission of a zero message. One could for example agree that if (e.g.) the first bit of the message is one, the real message is zero. If the first bit of the message is zero then the message is not zero. With such a protocol the pattern $(0, 0, \ldots, 0)$ will never be send and as a consequence the authenticity of the message can be protected without protecting the privacy. Remark that the authentication key H has to be secret in all circumstances otherwise modification is easy.

The discussed scheme can be used to protect the authenticity of a message without protecting the privacy. However this system is not acceptable in countries or in circumstances that "others" want to be able to verify that the communication is not used for spying. Such situations can occur as a restriction of local laws, or to be used to verify military actions, e.g., a ban of the testing of nuclear weapons [22]. The reason, why the described algorithm is unacceptable is that one can understand the message M, but one is never sure that the sender will not transmit a secret message instead of the authenticator R.

The length of the key in these schemes is $(q+1)n$ bits respectively $qn + n + q$ bits, depending if we only protect the authenticity, or privacy and authenticity. The keyexpansion is only the half compared with the scheme discussed in Section 4.2. The ciphertext expansion here is $(n+q)/n$ or not significant. Now, a scheme will be presented in which the length of the key is only about the double of the length of the message (about $2n$ bits). Remark that in a practical secure scheme q is normally 64, such that the expansion of the key in the scheme we just discussed, is still large.

4.4. Other unconditionally secure authentication schemes

Some other authors discussed unconditionally secure authentication schemes before, but did not use this name. Simmons [23] and Brickell [2] discussed several bounds related to the security level, the keylength, etc. *They called a system perfect (or double perfect) if the key was used optimally, or was not longer than necessary.* Gilbert et. al. [12] discussed implementations of such perfect authentication systems.

It is easy to prove that the schemes discussed in previous sections are not perfect in the sense defined by Simmons [23] or double perfect as defined by Brickell [2]. This means that the key is not used optimally. To obtain such an optimal keylength one could use projective planes, as discussed in [12] on pp. 414 – 415. However for long messages (e.g. Megabits) the calculations in the Gilbert scheme are awful. Now a scheme will be presented which is unconditionally secure, for which the calculations are not too awful, and for which the expansion of the key is only about two.

The idea is that the users first agree on a lowerbound for the security level P. The message is divided up in blocks of length $q = \lceil \log_2 P \rceil$ bits. So the message $M = (M_1, M_2, \ldots, M_\alpha)$ where $\alpha \cdot q \geq n$ and $(\alpha - 1) \cdot q < n$. If n is not a multiple of q then one fills the message up with zeros. The security level will be 2^q. For each q bits a key of length $2q$ bits is used. So the length of the total key is $2\alpha q$ (about $2n$) bits and is really random. The idea of projective planes [12] is used to generate for each M_i a binary vector $(t_i^1, t_i^2, \ldots, t_i^q)$ in $GF(2^q)$ (remark that this binary vector was called c on page 414 in [12]). The scheme continues as the previous one (see Section 4.3) except that:

- instead of H the vectors $(t_i^1, t_i^2, \ldots, t_i^q)$ are used, where $1 \leq i \leq \alpha$
- Eqn. 1 is replaced by:

$$r^j = \bigoplus_{i=1}^{\alpha} t_i^j \qquad \text{for each } j \quad (1 \leq j \leq q). \tag{2}$$

- The scheme is normally used to protect the authenticity, if you also want to protect the privacy you use a different privacy key which length is n bits.

In next section we will discuss the practical and theoretical consequences.

5. Practical and theoretical consequences

All schemes we discussed can be extended if we replace the modulo 2 sum by another modulo sum (e.g. modulo 53). We will wonder if the discussed schemes are useful. Consequences of the discussed schemes on the security of stream ciphers and DES will also be discussed.

5.1. Are previous schemes useful?

If you find the Vernam scheme impractical for your application, you find the discussed schemes also impractical. If however, you are dealing with national security (e.g., military and diplomacy) or you need unconditional security, the discussed schemes are interesting. If you use the Vernam one-time pad, you have to take into consideration that e.g., terrorists can modify your messages. As a consequence of terrorists attacks and of computer networks the problem of authenticity becomes more and more important, also in domains as the military or other governmental organizations. The discussed schemes allow to protect the authenticity unconditionally. The scheme discussed in Section 4.4 is preferable because the ciphertext expansion is about inexistent, while the length of the key is only about twice the length of the message. The security level obtained is less than the one which can be obtained ([2], [12], [22]), but the scheme is much more practical if long or very long messages are sent, while one can still choose the security level one wants. The key is used as in Vernam, so is random and distributed beforehand on a secure way. *Senders and receivers can easily handle messages with variable length.*

5.2. Stream ciphers protecting authenticity

Some authors, e.g. Denning [4] (pp. 144) say that stream ciphers have the disadvantages that the message can easily be modified. The schemes which we discussed here and certainly the one in Section 4.3 allow to modify stream ciphers such that they can be used to protect the authenticity. However their security is *no more unconditionally secure*, because stream ciphers generate pseudorandom, and their security is based on computationally complexity. If one adapts stream ciphers to protect authenticity, we suggest to use a different key for the pseudorandom generator which will be used to protect the privacy and the one which will be used to protect the authenticity.

5.3. Hashing and unconditionally secure authentication

One could remark that the final solutions (proposed in Section 4.3 and in Section 4.4) hide the use of hashing, which seems the natural solution. However if hashing is used in these schemes, one looses the unconditional security. Indeed the difficulty to find two different texts which produce the same authenticator, is then based on the computational complexity. The solution of hashing can be used when unconditional security is not necessary, e.g. in the scheme discussed in Section 5.2.

Most of the schemes which we will discuss further on, do not protect the authenticity unconditionally, however some remarks are also valid for them.

5.4. The protection of privacy and authenticity together

In the schemes we have discussed we used a different key to protect the privacy from the one used to protect the authenticity. We suggest that a similar strategy would be used for all cryptosystems. Jueneman et. al. [15] suggested the same in their paper. Another example of the importance to use different keys will be discussed in Section 5.5.

We can also conclude that in a conventional system the protection of privacy and authenticity are *partly* (see the previous last paragraph of Section 4.3 and Section 6.1) separable, and that the use of a mode as e.g. CBC does not necessarily guarantee the protection of the authenticity. So we do not agree with the remark of Denning [4] (pp. 10), cited in the introduction (Section 1).

5.5. The consequences on the use of DES

Today DES [10] is probably the most used commercial crypto algorithm. An authentication scheme was proposed by the NBS [11]. We will show that if you protect both privacy and authenticity with the same key, that a fraudulent message may be easily injected, and that one can easily modify messages. Jueneman et. al. [15] suggested to use different keys to protect the authenticity and the privacy. Several attacks were presented in the case that the same key would be used, even if the NBS authentication method is used. They were able to modify the message without affecting the authenticator, however the received plaintext will (in almost all cases) be "garbage". The attack which will be presented now, allows an active eavesdropper to modify a message in a fraudulent one, he chooses! So in bank applications he is able to transfer money on his account such that the fraud will not be detected by the authentication system.

The attack presented here is an adaptation of an idea originating from Cloetens [3]. In [13] a realistic exhaustive keysearch machine was presented which would break DES in about four weeks, and would cost about $1,000,000. The idea is to use such a machine. Hereto let us make some reasonable assumptions: the key is only modified once each four weeks, the privacy protection uses the same key as the authentication process and the active eavesdropper uses a known plaintext attack. He can then exhaustively determine the key, starting from a block of the ciphertext and a block of the plaintext. This attack is not influenced if the encryption system uses a mode. Once that key is found, the active eavesdropper can inject or modify messages. One could argue that by modifying the key frequently enough, the attack is not more valid. However, it can still be used! Suppose that the sender and receiver modify their key each s seconds. The active eavesdropper can now stop his exhaustive keysearch machine each s seconds and try to find the next key. If the machine does this process enough randomly, it will *not* find *a* key after four weeks with a probability:

$$\left(\frac{x-1}{x}\right)^x \qquad \text{where} \quad x = \frac{3600 \cdot 24 \cdot 7 \cdot 4}{s}.$$

In limit *a* key will be found after four weeks with a probability $1 - e^{-1}$, in eight weeks with a probability $1 - e^{-2}$, and so on. Once a key is found the active eavesdropper modifies the message as he wants.

Remark that the above attack is valid for all modes as long as the key, used to protect the privacy, is the same as the key, used to protect the authenticity, even if that key is modified frequently! Also, for several non-standard implementations of the DES such an attack is possible. Remark that the attack can not be avoided if for each message a different key is used (e.g. the first message is encrypted using key K_1, the second with K_2 and so on). Indeed because the attack is even in limit ($x \to 0$) still valid. To realize the attack, it is enough to add a delay in the transmission and to have a described exhaustive machine which can be easily restarted. *Even in the case the key used to protect the authenticity* is different *from the one used to protect the privacy, care is necessary. Indeed if short messages are sent, it is trivial to prove that a similar attack is still valid. The time needed to break, increases only linearly with the length of the message.* This is a consequence of the linearly increasing time to calculate the authenticator, and as a consequence of the exhaustive attack. Similar as in the above case, it does not help to modify the key frequently. *Such situations of short messages can be forced with chosen text attacks*! Better exhaustive machines (than the one discussed in [13]) can make the discussed attacks cheaper, faster and so on. This discussion is certainly outside the scope of this paper (for more details see [7]).

Each encryption algorithm which is "similar" as DES suffers from this attack. The meaning of "similar" is explained in [18]. One could wonder if it would not be better to use always the so called "triple encryption" in order to avoid such and similar attacks. But even in that case we recommend that the key used to protect the privacy *is different* from the key used to protect the authenticity.

6. Can a public key scheme protects the authenticity without privacy?

6.1. Introduction

It is evident that the RSA scheme [19],[20] can protect (today) the authenticity of short messages (taken into consideration that a secure key is chosen [4]). However not so much research is done to protect the authenticity of long messages with RSA. Indeed, if one divides the message up in blocks and authenticates the blocks separately then an active eavesdropper can mix the blocks up, repeat them, delete some, and so on. To protect the authenticity of long messages, some authors propose the use of hashing functions, or propose to use DES and to distribute the key with RSA, or to use a protocol that

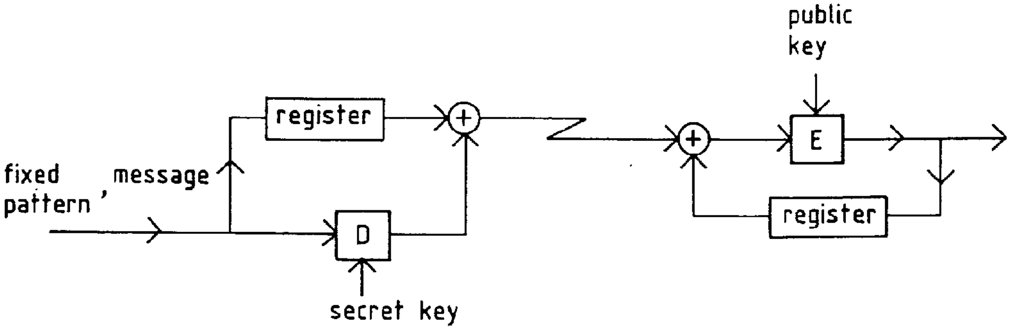

Figure 2: CBC mode with a public key algorithm to protect authenticity

"ping-pongs" the message from sender to receiver and back and so on. However these ideas have several disadvantages:

- hashing functions suffer mostly from "meet-in-the-middle" attacks [1]
- protocols, hashing functions and DES are extra costs
- ping-pong protocols slow down the communication
- hashing functions and DES do not allow the protection of the authenticity *without excluding the possibility to transmit secret information.* In some cases this is not acceptable e.g., as in arm limitation control [22] or if some country does not allow that encrypted messages are sent to foreign countries. Using a hashing function or DES, the authenticator can be replaced (e.g. partly) by secret information.
- random can not be used, because it can be misused for sending secret information.

We wonder if in a public key system *privacy and authenticity are* completely *separable* under the conditions mentioned (we don't use hashing functions, or a conventional cryptosystem, or a ping-pong protocol, or random). We will now come up with a mode to protect the authenticity of long messages, however the presented scheme is not secure.

6.2. An insecure proposal

In the scheme we use a CBC mode (see Figure 2) to protect the authenticity. The sender uses a feedforward and the decryption algorithm with his secret key. The message is followed by a fixed pattern A as authenticator (a variable one could contain secret information, what we do not want). The receiver uses a feedback and the encryption algorithm with the public key of the sender. The initial contents of the registers used in the feedforward and feedback is fixed and publicly known, otherwise we protect partly the privacy. We call this initial contents I. Because a feedback has a large error propagation one could expect that this system is secure. However this scheme is insecure if an active eavesdropper know one block, e.g. I. Because the receiver uses the public key, an active eavesdropper is able to follow exactly what the contents of the register is in the device of the receiver, he is also able to see what the output is and so on. He can now attack the protection by modifying arbitrary all sent blocks, except the last one corresponding with the authenticator. He will also modify the last transmitted block, however he calculates the modification such that the receiver still receives the authenticator A. Because he is able to do all the calculations the receiver does, he knows the previous last received message block M'_n. If the active eavesdropper would not have modified the transmitted blocks the receiver would have received M_n instead of M'_n. The last block transmitted by the sender is $M_n \oplus D(A)$, where $D(\cdot)$ is the decryption operation. If the active eavesdropper exors $M_n \oplus M'_n$ with the last block, the receiver will accept the message.

Using this attack the received message will probably be "garbage", nevertheless it will be accepted in an automatic system. For terrorists it does not matter if the received text is garbage, sabotage is enough. The active eavesdropper knows however the message that the receiver will receive and can try to come up with better "garbage".

The mode here proposed is insecure, and one can wonder if a secure mode exists. As long as no secure mode is found to protect the authenticity of *long* messages without protecting privacy and which satisfies the mentioned conditions, we have to conclude that authenticity can not be *completely* separated from privacy. This conclusion would be strange!

7. Conclusions

7.1. Overview of the presented results

We came up with several unconditionally secure authentication schemes. Nevertheless that they are not perfect in the sense of Simmons definition, the last unconditionally secure scheme proposed in our paper is more practical than the perfect ones.

We came up with stream ciphers which protect the authenticity.

We demonstrated that the ideas of Denning [4] about conventional systems are oversimplified. There exist conventional systems that protect the privacy but not the authenticity (e.g. Vernam one–time pad).

The protection of privacy and the protection of authenticity (and integrity) are partly separable, we wonder if they are completely separable.

7.2. Advices for users

If you need to protect privacy and authenticity use different keys for the different purposes.

Use triple encryption in DES. A standard (e.g. ANSI, ISO) which does not always use triple encryption is unacceptable. This is true as well as for the protection of the authenticity as well as for the protection of privacy (A full discussion would be too long and out of the scope of this paper, see [7] and [13]).

7.3. Acknowledgements

The author wants to thank Ernest Brickell for discussions about perfect authentication systems. These discussions where impossible without the visiting position at the

Univeristy of New Mexico, to who I am very grateful. The author thanks Henri Cloetens for his personal communication about the authentication with DES.

REFERENCES

[1] S. G. Akl, "On the security of compressed encodings," *Advances in Cryptology, Proc. Crypto 83*, Santa Barbara, California, U. S. A, August 21 - 24, 1983, pp. 209 - 230.

[2] E. F. Brickell, "A few results in message authentication," *Congressus Numerantium*, vol. 43, December 1984, pp. 141 - 154.

[3] H. Cloetens, personal communication.

[4] D. E. R. Denning, "*Cryptography and Data Security*", Addison - Wesley, Reading, Mass. , 1982.

[5] Y. Desmedt, J. Vandewalle and R. Govaerts, "The mathematical relation between the economic, cryptographic and information theoretical aspects of authentication," *IEEE Intern. Symp. Inform. Theory*, St. Jovite, Quebec, Canada, September 26 - 30, 1983, Abstract of papers, pp. 93.

[6] Y. Desmedt, "*Analysis of the Security and New Algorithms for Modern Industrial Cryptography*", Doctoral Dissertation, Katholieke Universiteit Leuven, Belgium, October 1984.

[7] Y. Desmedt, F. Hoornaert and J.-J. Quisquater, paper in preparation.

[8] W. Diffie and M. E. Hellman, "New directions in cryptography," *IEEE Trans. Inform. Theory*, vol. IT-22, no. 6, pp. 644 - 654, November 1976.

[9] H. Feistel, "Cryptography and computer privacy," *Scientific American*, vol. 288, no. 5, May 1973, pp. 15 - 23.

[10] FIPS publication 46 "Data Encryption Standard," *Federal Information Processing Standards Publ.*, National Bureau of Standards, January 1977.

[11] FIPS publication 81, "DES modes of operation," *Federal Information Processing Standard*, National Bureau of Standards, U. S. Department of Commerce, Washington D. C. , U. S. A. , 1980.

[12] E. N. Gilbert, F. J. MacWilliams, and N. J. A. Sloane, "Codes which detect deception," *Bell Syst. Tech. Journ.* , vol. 53, no. 3, March 1974, pp. 405 - 424.

[13] F. Hoornaert, J. Goubert, and Y. Desmedt, "Efficient hardware implementations of the DES," *Advances in Cryptology, Proc. Crypto 84*, Santa Barbara, California, U. S. A, August 19 - 22, 1984 (Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1985), pp . 147- 173.

[14] R. R. Jueneman, "Analysis of certain aspects of output feedback mode," *Advances in Cryptology, Proc. Crypto 82*, Santa Barbara, California, U. S. A, August 23 - 25, 1982, pp. 99 - 127.

[15] R. R. Jueneman, S. M. Matyas and C. H. Meyer, "Authentication with manipulation detection code," *Proceedings of the 1983 IEEE Symposium on Security and Privacy*, Oakland, California, April, 1983, pp. 33 - 54.

[16] D. Kahn, "Modern Cryptology," *Scientific American*, July 1966, pp. 38 – 46.

[17] A. Konheim, *"Cryptography: A Primer,"* John Wiley, Toronto, 1981.

[18] J.-J. Quisquater, Y. Desmedt and M. Davio, "The importance of "good" key scheduling schemes (How to make a DES* scheme with $\leq$ 48 bit keys?)", presented at Crypto '85, Santa Barbara, August, 1985, to appear in: *Advances in Cryptology, Proc. Crypto 85*, (Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1986).

[19] R. L. Rivest, A. Shamir, and L. Adleman, "On digital signatures and pulic-key cryptosystems, "*Massachusetts Institute of Technology Technical Report LCS/TN–82*, Cambridge, Massachusetts, April 1977.

[20] R. L. Rivest, A. Shamir and L. Adleman, "A method for obtaining digital signatures and public key cryptosystems," *Commun. ACM*, vol. 21, pp. 294 – 299, April 1978.

[21] C. E. Shannon, "Communication Theory of Secrecy Systems," *Bell Syst. Tech. Journ.*, vol. 28, pp. 656 – 715, Oct. 1949.

[22] G. Simmons, "Symmetric and Asymmetric Encryption," *ACM Computing Surveys*, vol. 11, no. 4, December 1979.

[23] G. Simmons, "Authentication theory/coding theory," *Advances in Cryptology, Proc. Crypto 84*, Santa Barbara, California, U. S. A, August 19 – 22, 1984 (Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1985).

[24] G. S. Vernam, "Cipher Printing Telegraph Systems for Secret Wire and Radio Telegraphic Communications," *Journal American Institute of Electrical Engineers*, v. XLV, pp. 109 – 115, 1926.