PUBLIC-KEY SYSTEMS BASED ON THE DIFFICULTY OF TAMPERING
(Is there a difference between DES and RSA?)

Yvo Desmedt * and Jean-Jacques Quisquater **

(*) Katholieke Universiteit Leuven, ESAT, Belgium [1].
Current address: Université de Montréal, Dépt. IRO, Case postale 6128, succursale A, Montréal (Québec), H3C 3J7 Canada.
(**) Philips Research Laboratory Brussels, Avenue Van Becelaere, 2, B-1170 Brussels, Belgium.

Abstract

This paper proposes several public key systems which security is based on the tamperfreeness of a device instead of the computational complexity of a trapdoor one-way function. The first identity-based cryptosystem to protect privacy is presented.

EXTENDED ABSTRACT

1 Introduction

We first give three main motives for this paper and overview the presented ideas.

Since the invention of public-key systems by Diffie and Hellman almost all public-key systems proposed were based on some computational hard problems (e.g. factoring). It was however shown that it is *not* easy to design a secure public-key system based on computational hard problems. Examples of failures are the Lu-Lee system, the Merkle-Hellman knapsack scheme (and others) and the Matsumoto-Imai scheme. If we remark that the McEliece scheme is not enough analysed to be used, there do not exist fast public-key systems (the speed of RSA is today less than 64 kbit/sec.). This is one of the main reasons to come up with other public-key systems.

Bennett and Brassard remarked that it is not necessary to use computational complexity to design a public-key system. As an example they started from the uncertainty principle, which claims that some physical problems are very hard to solve (impossible to measure). Bennett and Brassard mentioned that their system would remain secure if NP=P and if factoring would be easy. However the cryptosystems they proposed are today impractical. One can conclude that a second reason for this paper is to design cryptosystems which are not based on the assumption that trapdoor one-way functions exist.

The authenticity of the public key is a major problem in the set-up of a secure cryptosystem, certainly in the case of a large network. A nice solution was proposed by Shamir in 1984 called "*identity-based cryptosystem*". Instead of using the public key of the receiver (to encrypt in order to protect the privacy of a message), the name of the receiver is used as public key. The secret key of each user was calculated by an authority at the start-up of the system. (It is not excluded that the authority destroys itself after the start-up of the system.) Public-key systems, identity-based cryptosystems and their key generation are systematically explained in Fig. 1.

[1] This research was done while the author was aangesteld navorser NFWO at the Katholieke Universiteit Leuven

A.M. Odlyzko (Ed.): Advances in Cryptology - CRYPTO '86, LNCS 263, pp. 111-117, 1987.

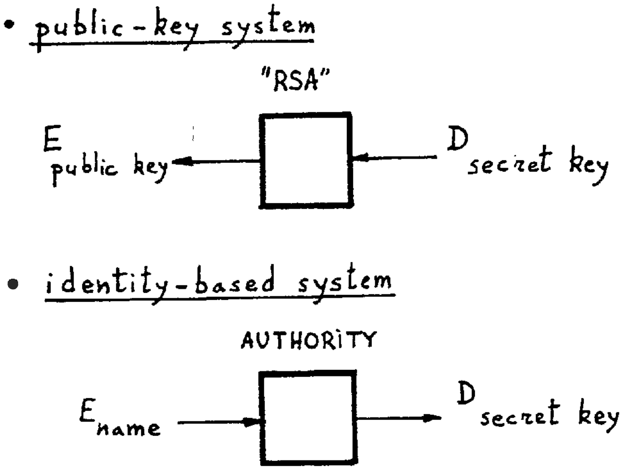

Figure 1: Key generation for public-key and identity-based systems

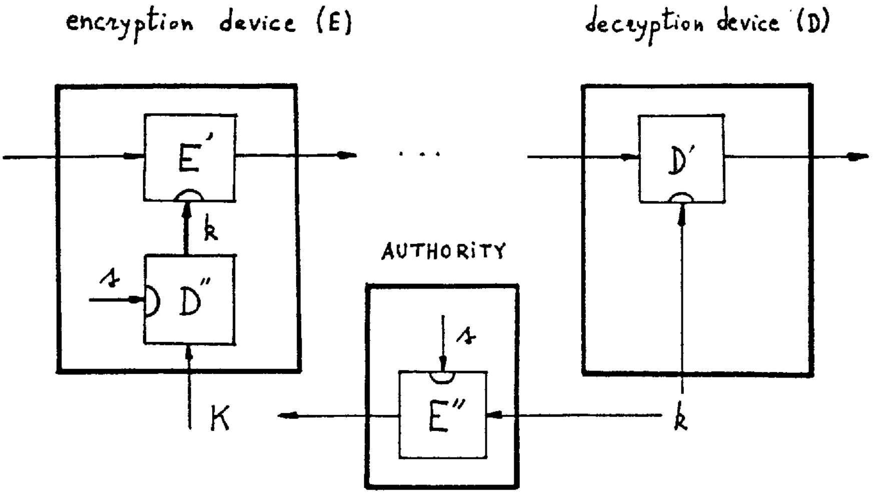

Figure 2: A first implementation of a public-key system

In our paper we start from the assumptions that hard conventional systems exist and that it is possible to make tamperfree devices. Remark that the first assumption is based on the complexity of algorithms, but seems acceptable, certainly if one takes into consideration that it is much harder to build trapdoor one-way public-key systems than conventional ones. Without the second assumption a lot of modern uses of cryptography would become unsecure. Indeed a secure system must be tamperfree otherwise an opponent can simply steal the secret key used in the system. Several practical systems start from this second assumption. *E.g.*, a software copyright protection system proposed by NPL becomes completely insecure if tamperfree devices can not be build. Remark too that each identification method is at least partially based on some tamperfree system or card (see also Section 5).

Given two conventional cryptosystems and the existence of tamperfree implementations we propose in our full paper several public-key systems, and the first identity-based cryptosystem to protect privacy.

2 Public keys

2.1 The basic idea

Let us give an example of such a system. From now on we call E', D', E" and D" the encryption and decryption of respectively the first and second conventional cryptosystems. Special cases use the algorithm DES in encryption mode for E' and E" or decryption mode for D' and D". To obtain a public-key system three devices are used: an encryption device (corresponding to the operation E), a decryption device (corresponding to the operation D) and a system which generates the public key starting from the secret key (corresponding to the operation G). Each user of the system generates a secret key k. He obtains his corresponding public key K by applying G on k, or $K = G(k)$. The device G is nothing but E" with a supersecret key s (which in the best case nobody knows). The device G is tamperfree so that it is hard to find the key s. In this example the supersecret key s is used in all devices G.

2.2 Two implementations of such a public-key system

We now discuss two implementations to obtain such a public-key system (see also Fig. 2 and Fig. 3).

In the first example (see also Fig. 2) the decryption device (D) uses the secret key. In fact here D is equal to D'. The encryption device (E) uses as a black box the public key K. The system E is build up using E' and D". The box E is tamperfree. In the box E first D" is used to find k, or $k = D''(K)$ using the supersecret key s. This last calculation is done inside E, and no trace of this calculation and its result can leak out to the outside world. In other words because the device E is tamperfree it is hard to find k. The encryption of messages is done by E' using the key k.

The described scheme can be used to protect, as a public-key system, the privacy and authenticity of messages as well to sign. To protect privacy the sender uses E with the public key of the receiver (although the receiver uses D with his secret key). Remark again that nevertheless the sender uses in fact the secret key of the sender, he cannot access it. To sign the sender uses D with his secret key (evidently redundancy is introduced in the

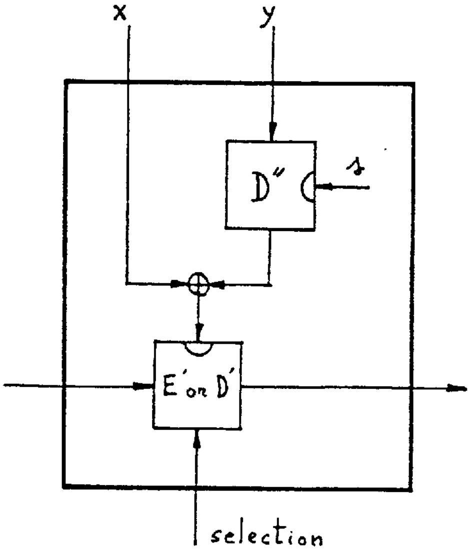

Figure 3: A second implementation of a public-key system

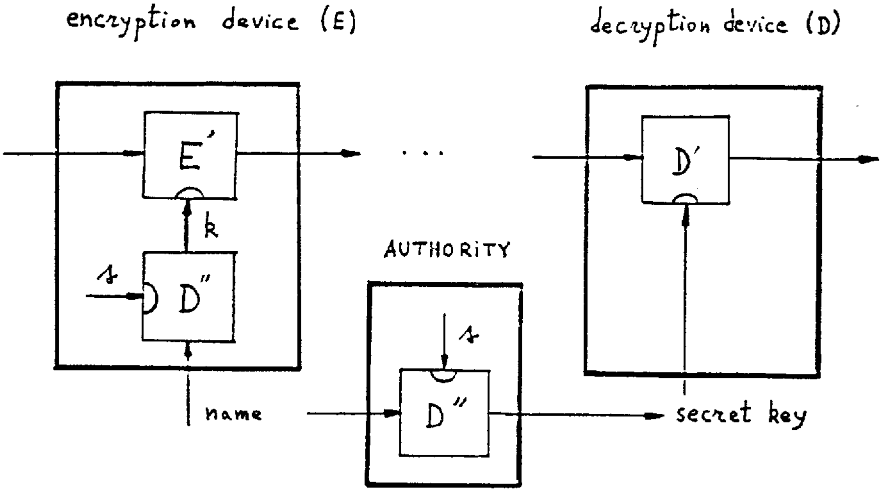

Figure 4: The first identity-based system to protect privacy

message). The receiver can check the signature (using the mentioned redundancy). The sender is the only one who could generate that signature.

The second implementation has the advantage that each user in the system has the same tamperfree device for encryption as well as for decryption. Let us describe such a system in some words. For this paragraph, we refer to Fig. 3. Let (T) be the tamperfree device used in the system. As for the first system, each user i generates a secret key k_i and a corresponding public key K_i. For that he uses the device G as already discussed. The device (T) contains E', D', D'' and the supersecret key s as described in Fig. 3. To send an encrypted message to a user B, a user A uses the device (T) in mode encryption and applies his secret key k_A to the input x and the public key K_B of the user B to the input y. To decrypt this message the user B uses the device (T) in mode decryption and applies his secret key k_B to the input x and the public key K_A of the user A to the input y. In these two phases, the effective key in use is the same but is unknown to the two parties. There are many variants to this scheme with the possibility of a session key, a.s.o. Let us remark that using a symmetric cryptosystem (sometimes called conventional system) together with such a symmetric implementation (the devices are the same for the encryption and the decryption) leads to an asymmetric cryptosystem (sometimes called public–key system).

3 Identity–based cryptosystem

By modifying a little bit previous examples it is no longer necessary to use public keys (or the public key of somebody is equal to his name or identification). The key generation machine G now is modified. The system G now uses D'' (with the supersecret key s) and the input of G is the name (or a sufficient identification of the person to be unique), the output is the secret key of the user (see also Fig. 4). In order to avoid frauds the uses of G are controlled by an authority. Each user can use G only once, and is only allowed to give as input something that corresponds with his identification (birth day, name of his father, name of company, ...). This is a first advantage because it avoids in large networks the authentication of the public key. This technique gives a first solution to a problem open by Adi Shamir, to propose an identity-based cryptosystem to protect privacy.

4 Security

In this section only necessary conditions in order to obtain a secure implementation are discussed. Sufficient conditions are still under research.

The system E'' has to be a secure cryptosystem such that all attacks fail in finding s by cryptanalytic methods. Therefore it is necessary that E'' is secure e.g. against an adaptive chosen text attack. The reader could wonder how an adaptive chosen text attack could be set up, certainly if an authority limits the use of the device G (as in the case of identity-based cryptosystem). The answer is that the adaptive aspect can be obtained if several users (which have e.g. special names) collaborate.

Evidently the cryptosystems E', D' and D'' have also to be secure cryptosystems.

Another necessary condition is that the system may not have (or use) weak keys (a term introduced by Davies related to weak keys in DES) or similar weaknesses. Using a weak key there is no difference between an encryption and a decryption operation.

Indeed an asymmetry is required to obtain public-key systems. If not, this implies that everybody can generate signatures of an opponent using his public key, because E' will in fact internally use the secret key of the opponent and for weak keys this E' operation is the same as the D' operation. In general in order to protect signatures (with the described scheme) it must be hard to generate outputs of D' starting from outputs of E'. So semi-weak keys are also dangerous. The same remark holds for the protection of privacy. Otherwise everybody could decrypt message send to Bob, using Bob's public key for a similar reason.

5 Advantages, disadvantages and other aspects

A major advantage of the discussed systems is the speed. Using DES (and dropping weak keys) much faster public-key systems can be made. An important disadvantage of the system is that everybody who knows s can attack all users! However in some cases such a property is desired (by the authority), as in the case of communications between persons of a same company (e.g. a bank). In this context we remark that the key distribution problem in some large companies (when a normal conventional system is used), can be hard to solve.

Remark also that in previous discussions one can e.g. replace the supersecret key s, by some secret function. In the discussed example E', D', E'' and D'' are public known conventional algorithms. It is trivial to understand that the same holds if E', D', E'' and D'' are secret. In other words if some organization promotes secret algorithms, key distribution centers can be avoided and one can use the described public-key method. Indeed in order to maintain the secrecy of the used secret algorithms, the devices must be at least tamperfree.

Finally one can question that the described system is really a public-key system. To solve this problem one can use the well known Turing test. Suppose DES and RSA are used (to be mathematically correct n DESes are used with n different keys), is it then possible to find in polynomial time (as function of n) if DES or RSA is used? It is well known that the answer is yes, using the Jacobi symbol in a known plaintext attack. In a secure implementation of RSA and DES it must be hard to make a difference between real random and the ciphertext in polynomial time. As a consequence if DES (in such public-key system) and RSA are used in a secure implementation, no difference can be observed in polynomial time.

Remark that in a part of our paper on the importance of good key scheduling schemes (1985, CRYPTO '85), we did not obtain a real public-key system as we do here, moreover, some of our assumptions there are the opposite of some assumptions here.

It is not too hard to find better schemes which satisfy some desired properties, some of these other schemes are still under research. For instance, in the context of tamperfree devices, it is possible to design claw-free functions with conventional cryptoalgorithms and thus to have very fast algorithms to sign documents (Rivest, Goldwasser, Micali, Goldreich).

Another advantage is that the above idea of identity-based cryptosystem can be used in a protocol in order to protect passports. Let us again start from the assumption that tamperfree devices and that conventional cryptosystems exist, where the decryption operation *can not be obtained by applying polynomially the encryption operation. **Remark** that the assumption of tamperfree devices is also necessary in Shamir's protocol* (presented at the same conference). Indeed if an owner of a passport is able to find his corresponding secret (the square roots in Shamir's protocol), there is no protection against cloning. For very busy businessman or consultants or researchers it can be an important advantage to clone themselves, in order that the cloned one handles the public relation and other aspects, for which the original persons are too busy. If a difference has to be made between the identity of the person and his cloned version, the person himself is not allowed to know the secret corresponding with his secret. So tamperfree devices are necessary.

Our identification protocol is very similar to the one of Shamir, except that a different type of algorithm is used and that the country that is visited generates the random. Again we use the identity-based cryptosystem to protect signatures. Each country (*e.g.* Israel) distributes to other countries the E devices, containing their supersecret s. During use, a visitor (*e.g.* Alice) tells the officials her nationality (*e.g.* Israelian) and her identity. The country which she visits (*e.g.* Belgium) then uses the tamperfree device obtained from Israel and the name (identity) of Alice is used as key by that country (e.g. Belgium). Belgium generates then some random t and gives $E(t)$ to Alice. If Alice knows her secret key (obtained from her country: Israel), she is able to decrypt it and obtain t, which she gives to Belgium. If both t's match Belgium accepts Alice identity. The disadvantage of this system is that 200 different kinds of machines are necessary (each for each country). The advantage is that each country relies on their own technology to avoid false passports made by other countries. A proof for the security of the discussed protocol is still under research.

6 Open Problems

A main open problem is to find an identity-based cryptosystem which protects privacy and which security is not based on the assumption of the existence of tamperfree devices.

Another open problem is to overcome the problem of the supersecret key s, mentioned in Section 5. Does there exist an identity-based cryptosystem to protect privacy which security is based on tamperfree devices and computational complexity and which use different supersecret s for different users. In other words that system would remain secure if the computational problem is solved, but the tamperfreeness is still valid, or if the reverse situation happened.

The authors have the impression that both mentioned open problems are strongly related.

Remarks

Other works, more or less related to this one, were made by M. E. Smid, R. E. Lennon, S. M. Matyas and C. H. Meyer, H. Beker and M. Walker.

Acknowledgement

The authors are grateful to Adi Shamir for the discussions related to Section 6.