An Impersonation-Proof Identity Verification Scheme*

Gustavus J. Simmons
Sandia National Laboratories
Albuquerque, NM 87185

Most schemes for the verification of personal identity are logically flawed in that they require an individual to exhibit a piece of private, i.e., secret, information such as a computer access password, a telephone credit card number, a personal identification number (PIN), etc., to prove his identity. The logical problem is that this information, once exhibited, is potentially compromised and could be used by anyone to undetectably impersonate the legitimate owner. What is needed is a protocol that will allow an individual to "prove" that he knows the secret piece of information, whose possession is equated with his identity, without revealing anything about the information itself which could aid a would-be cheater to impersonate him. Several investigators have proposed identification schemes to accomplish this [1,2,3,4] that depend on interactive-proof schemes, often referred to as zero-knowledge proofs or ping-pong protocols, in which the individual responds to a series of queries in a way that the legitimate user could, but which an impostor (probably) could not. We describe a simpler identity verification scheme which uses a public authentication channel to validate a private authentication channel belonging to the individual who wishes to prove his identity. The public and the private channels can be completely independent and can even be based on different authentication algorithms, or they can both be of the same type. This scheme also provides certified receipts for transactions whose legitimacy can later be verified by impartial arbiters who were not involved in the transaction itself.

The identity verification scheme described here presupposes the existence of a trusted issuer of validated (signed) identification credentials. This could be a government agency, a credit card center or financial institution, a military command center, a centralized computer facility, etc. The issuer first establishes a public authentication channel to which he retains the (secret) authenticating function. For simplicity, we will use the well known authentication channel based on the RSA cryptoalgorithm for both the public (issuer) and the private (user) channels, although, as mentioned earlier, authentication channels based on any other algorithm would work equally well. The issuer chooses a pair of primes p and q by the same standards used to compute a good RSA modulus, i.e., so that it is computationally infeasible for anyone to factor n, and then calculates a pair of encryption/decryption exponents, e and d such that;

* This work performed at Sandia National Laboratories supported by the U. S. Department of Energy under contract no. DE-AC04-76DP00789.

C. Pomerance (Ed.): Advances in Cryptology - CRYPTO '87, LNCS 293, pp. 211-215, 1988.

$$ed \equiv 1 \pmod{\varphi(n)} \quad .$$

n and d are made public. The issuer keeps e (and equivalently the factors p and q) secret; in fact, the security of the system against fraudulent claims of validated identity is no better than the quality of protection given to e by the issuer. The issuer also chooses a polyrandom function f that maps arbitrary strings of symbols to the range [0,n). By polyrandom we mean that f cannot be distinguished from a truly random function by any polynomially bounded computation. Many strong, single-key, cryptographic functions, such as the DES, appear to adequately approximate this condition. f is also made public by the issuer.

User i's identity is associated with an identifier, I_i, consisting of such information as his social security number, his bank account or credit card number, his military ID, etc., and which could also include physical descriptors such as digitized fingerprints, voice prints, retinal eye prints, etc., or any other useful descriptive information, as well as any limitations on the authorization conveyed in the signed identifier, such as credit limits, expiration date, levels of access, etc. Most importantly, I_i must include the public part of the user's personal authentication channel consisting of an RSA modulus n_i, $n_i > n$, and an associated decryption exponent d_i, plus, redundant information, such as message format, fixed fields of symbols common to all identifiers, I_i, etc. The issuer calculates

$$m_i = f(I_i)$$

and encrypts m_i using his secret key, e, to form the signature, s_i, for the identifier, I_i,

$$s_i \equiv m_i^e \pmod{n}$$

The issuer gives the credential (I_i, s_i) to user i. No part of the credential need be kept secret. However, the user must keep secret his private encryption exponent, e_i, corresponding to d_i. His security against impersonation is dependent on his protecting e_i, since his proof of identity in the scheme is equated to knowing e_i.

The public information is the issuer's modulus n and decryption exponent d, the polyrandom function f and a knowledge of the redundant information present in all of the I_i, which must be sufficient to prevent a forward search cryptanalytic attack [5] on the polyrandom function f. In other words, someone wishing to fraudulently validate an identity could calculate $s_j^d = m_j$ for randomly chosen signatures s_j in the hopes of obtaining a hit with f(I) for some usable I -- this is the forward search attack. By making I contain sufficient redundant information, the probability of success of this sort of attack can be made as small as desired.

When user i wishes to prove his identity to a party A, say to gain access to a restricted facility or to log on to a computer or to withdraw money from an ATM, etc., he initiates the exchange by identifying himself to A using his identification credential;

$$i \xrightarrow{(I_i, s_i):u_i} A$$

u_i is a string of symbols that describes or identifies the transaction i is requesting; u_i could be the date, the amount of the withdrawal, etc. A, who need not be a subscriber himself, i.e., he may not have an identification credential issued by the trusted issuer, replies with a string of symbols, u_A, that describe the transaction from his standpoint; terminal ID, transaction number, confirmation of withdrawal amount, etc.

$$i \xleftarrow{u_A} A$$

Both user i and A form the concatenation of u_i and u_A, $u = u_i, u_A$, and calculate the polyrandom function f(u) of the resulting string;

$$z = f(u) \quad .$$

In addition, A calculates $f(I_i) = m_i$ and s_i^d which will in all probability equal m_i modulo n if and only if the issuing authority signed I_i with s_i. Hence A accepts the credential (I_i, s_i) as valid if and only if

$$f(I_i) = s_i^d \pmod{n}.$$

At this point in the protocol, A is confident that the user identified in I_i can authenticate messages using the private authentication channel described in I_i, in other words, that user i knows e_i. In particular, user i can calculate

$$t_i = z^{e_i} \pmod{d_i}$$

using his private exponent e_i, which he communicates to A;

$$i \xrightarrow{t_i} A \quad .$$

Note that z is being used effectively as a one-time key, indeterminate to both i and A because of the polyrandom nature of f, to permit i to give A an encrypted function of z in a form that will permit A to satisfy himself that i had to know e_i without providing any information whatsoever about e_i. A knows the identity claimed by i from I_i, which he accepts as valid if and only if the following identity is satisfied:

$$(1) \qquad t_i^{d_i} \equiv z \pmod{n_i}$$

If the person seeking to be recognized as user i really is who he claims to be, i.e., if he knows e_i, then (1) will be satisfied. However, if he is not user i, so that he doesn't know e_i, then in order for him to be able to impersonate i, i.e., to cause (1) to be satisfied, he must be able to find a number x such that

$$(2) \qquad x^{d_i} \equiv z \pmod{n_i} \quad .$$

n_i or d_i are values signed by the issuer in I_i with only the authorized user knowing e_i or equivalently the factorization of n_i. z is a pseudorandom number jointly determined by user i and by A. Solving (2) without knowing e_i is equivalent to breaking the RSA cryptoalgorithm from ciphertext alone.

A keeps the 4-tuple $\left[(I_i,s_i):u,t_i\right]$ as his certified receipt for the transaction. Anyone, using only publicly available information, i.e., n, d and f, can verify that the 4-tuple satisfies (1) which validates the transaction description and verifies that it was signed, i.e., endorsed, by user i. If both communicants require a certified receipt the one-way protocol described above can be easily modified to be a two-way protocol between two parties, i and j, both of whom must possess identification credentials validated by the issuer. All actions are symmetric in this case. The exchange is of the form

$$\begin{array}{ccc} i & \xrightarrow{\quad (I_i,s_i):u_i \quad} & j \\ & \xleftarrow{\quad (I_j,s_j):u_j \quad} & \\ & \xrightarrow{\qquad t_i \qquad} & \\ & \xleftarrow{\qquad t_j \qquad} & \end{array}$$

where user i would keep the 4-tuple $\left[(I_j,s_j):u,t_j\right]$ as his certified receipt, etc.

References

1. A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identification and signature problems," Proceedings of Crypto-86, Santa Barbara, August 1986, pp. 1-13.

2. A. Shamir, "Identity-based cryptosystems and signature schemes," Proceedings of Crypto'84, Santa Barbara, August 1984, pp. 47-53.

3. O. Goldreich, S. Micali and A. Wigderson, "Proofs that yield nothing but the validity of the assertion and the methodology of cryptographic protocol design," Submitted to 27th Symposium on Foundations of Computer Science, November 1986.

4. E. Okamoto, "Proposal for identity-based key distribution systems," Electronics Letters, Vol. 22, No. 24 (Nov. 20, 1986), pp. 1283-1284.

5. G. J. Simmons and D. B. Holdridge, "Forward search as a cryptanalytic tool against a public key privacy channel," Proceedings of the IEEE Computer Society 1982 Symposium on Security and Privacy, Oakland, CA, April 26-28, 1982, pp. 117-128.