AUTHENTICATION

G.M.J. Pluimakers and J. van Leeuwen

RUU-CS-85-9a
March 1985

Rijksuniversiteit Utrecht

Vakgroep informatica

Budapestlaan 6 3584 CD Utrecht
Corr. adres: Postbus 80.012 3508 TA Utrecht
Telefoon 030-53 1454
The Netherlands

AUTHENTICATION

G.M.J. Pluimakers and J. van Leeuwen

Technical Report RUU-CS-85-9a
March 1985

Department of Computer Science
University of Utrecht
P.O. Box 80.012
3508 TA Utrecht
the Netherlands

This paper is the summary of an invited talk to be presented at the conference "Data: Control and Management" of the NGI-Section on EDP-auditing, Noordwijkerhout, the Netherlands, May 6-8, 1985.

AUTHENTICATION*

G.M.J. Pluimakers and J. van Leeuwen

Department of Computer Science, University of Utrecht,
P.O.Box 80.012, 3508 TA Utrecht, the Netherlands.

Abstract. We describe the problem of authenticating messages in an environment on which many senders can communicate with many receivers. New techniques, e.g. from the area of public-key cryptography, have been devised to determine that messages indeed originate at the claimed source. We give an impression of the current (theoretical) developments.

1. Introduction. In applications that involve sending messages (data etc.) by computer over public media, advanced encryption methods are required to prevent unauthorized parties from reading the information that is transmitted. Traditional techniques (including DES) are based on the use of session keys for scrambling messages at the source and unscrambling them at the destination. The novel techniques of public key cryptography (Diffie & Hellman [3], cf Denning [2]) rely on the assumed computational intractability of certain mathematical problems, to obtain methods in which everyone can encrypt but only those who know how to solve the mathematical problem efficiently can successfully decrypt. The mathematical problems in use for this purpose are: factorization of large integers, quadratic residuosity for a composite modulus, the index with respect to a certain primitive root modulo a prime (the discrete logarithm problem), and several versions of the knapsack problem.

New problems of greater complexity arise in the design of methods in which a receiver B can determine that messages indeed originate at a claimed source A and were meant to be send by A at the present time.

* Extended abstract (march 1985)

Clearly the method should be such that A cannot deny having send the message M to B if B can present M and the proof that the "method" determined A as the sender of the current instance. (Usually time-stamps are incorporated to validate the timeliness of messages.) The same strategy must be used by A to check acknowledgements from B. We refer to this domain of design questions as the "authentication problem". Authentication is closely related to "authorization"; in this case B must authenticate A and verify, grant, and monitor certain rights (e.g. access rights) that A claims.

The difficulties in authenticating messages can be appreciated in the following paradigm. A holds a key, B holds locks for all parties it communicates with, B authenticates a party as being "A" if it presents a key that fits the lock that B holds for A, and A uses a trusted carrier C (e.g. a messenger or a datacom link) to communicate with B. A could present a counterfeit or stolen key, B could hold counterfeit or stolen locks, and C could alter messages (in collaboration with another party or, perhaps, with B itself). Authentication is a particularly pressing problem in EFT systems, electronic ordering and transaction systems, access restriction systems and, in a different vein, national defense systems. In general the following versions of the authentication problem are encountered:

(i) user authentication,
(ii) system authentication,
(iii) message authentication,
(iv) object authentication.

In this paper we give a brief account of recent developments concerning the authentication problem.

2. <u>User and system authentication</u>. User authentication is required when a user (A) wishes to gain access to a system over a direct, i.e. trusted, carrier. It occurs when A presents himself at a POS terminal, at the entrance of a restricted access building, or when A wants to log-in on a particular computer. A is authenticated by one of the following methods, or a combination there-of:

(i) a personal characteristic of A (e.g. fingerprint, voiceprint, Note that step 2 authenticates A (by the assumed difficulty of factoring). A and B now exchange the secret keys k_{AB} and k_{BA} (both products of suitable large primes) and engage in another round of authenticated message transfers to enable B to factor k_{AB} and A to factor k_{BA}. One can show that this enables A to send a row of quadratic residues mod k_{AB} to B in a form which B can decipher (with a similar action for B). The row is used as the seed of a secure random bit generator by A and B, which gives a one-time pad for encrypting and decrypting messages to be send from A to B (similar in the other direction). Goldwasser, Micali, & Tong [10] claim that the probability that any user C≠A,B can decipher a single bit or forge a single message, given a polynomial number of observed encrypted messages, tends to zero for sufficiently long seeds.

4. <u>Message authentication: public-key cryptosystems</u>. Besides the cryptosystems that employ session keys (like DES), there are several public-key systems available nowadays:

(i) the RSA scheme ([26]),
(ii) the Rabin scheme ([25]),
(iii) the Williams scheme ([29]),
(iv) the Pohlig-Hellman scheme ([21]),
(v) the goldwasser-Micali scheme ([8]),
(vi) the Merkle-Hellman ("knapsack") scheme ([18]),
(vii) the Graham-Shamir scheme ([2]).

For a discussion see e.g. Denning [2]. The schemes all depend on the assumed difficulty of solving a particular mathematical problem, which is essential for "breaking" (and decoding) it. For example, in the Goldwasser-Micali scheme A publicizes a number N_A (product of two secret large primes) and a quadratic non-residu y_A mod N_A with $(y_A|N_A)=1$. To send a message $M=m_1...m_k$ (in bits) to A, B sends a message $e_1\#...\#e_k$ with random e_i (integers mod N_A) such that e_i is a quadratic residu mod N_A if $m_i=0$ and e_i is "y_A times a quadratic residu" mod N_A if $m_i=1$ (in which case e_i is a quadratic non-residu with $(e_i|N_A)=1$). By the quadratic residuosity assumption this will only be intelligible to A, who knows the factors of N_A. (B need not know the factors in order to encrypt, as it is sufficient for him to just generate random squares mod N_A.) Several schemes are vulnerable to attacks or exhaustive search. Shamir [28] has shown that the original Merkle-Hellman scheme can usually be broken, by devising a polynomial time algorithm that solves the underlying knapsack equations with reasonable probability.

Public-key cryptosystems provide an elegant way of authenticating messages. A sends $E_B(D_A(M))$ rather than $E_B(M)$ to B, B computes $D_B(E_B(D_A(M)))=D_A(M)$ and $E_A(D_A(M))=M$ using the public E_A. Assuming that E_A gives no reasonable output unless the input is of the form D_A(message), only A could have send M because only A knows D_A. $D_A(M)$ is an example of a digital signature for M (by A). The cryptosystem must be commutative in order that this signature method works. Aside from the fact that not all cryptosystems are commutative, there still is the danger that certain $E_B(x)$ values (using additional information about x perhaps) will reveal the x that is encoded by some clever polynomial time algorithm. Thus one-way trapdoor functions like E_B are not necessarily sufficiently safe in all cases.

Goldwasser & Micali [8] have developed a theory in which the one-way trapdoor functions are replaced by so-called unapproximable trapdoor predicates B, which have the property that everyone can choose an x with $B(x)=0$ or with $B(x)=1$ but no one (without having the trapdoor information) can actually compute B(x) for given x. Deterministic encryption is replaced by probabilistic encryption, as exemplified in the Goldwasser-Micali scheme given above. It is claimed that in the limit no polynomial time algorithms can succeed in breaking even a single encrypted instance unless the conditions for unapproximable trapdoor predicates are violated, e.g. unless the quadratic residuosity assumption is broken in the given example.

5. <u>Message authentication: digital signatures</u>. The basic protocol of digital signatures (given above) applies to many cryptosystems, and can be used to authenticate both messages and users. The protocol is vulnerable is A claims he "lost" his D_A and denies responsibility for

a signature. Merkle [17] has suggested that secret keys be time-stamped and kept by a central authority. A key is considered valid until reported (and time-stamped) as stolen. Messages signed by A must be time-stamped at the central authorithy, in order that B can verify (and later: defend) that A's signature is valid at the time of receipt. Clearly the scheme does not preclude forging by a third party. In addition to the schemes derived from DES and public-key cryptosystems, the following signatures schemes have been proposed:

(viii) the Diffie-Hellman signature scheme ([3]),

(ix) the Shamir ("knapsack") signature scheme ([27]),

(x) the ElGamal signature scheme ([5]),

(xi) the Ong-Schnorr-Shamir signature scheme ([20]).

In the ElGamal scheme A publicizes a large prime p, a primitive root g modulo p, and an integer y modulo p of which the index e is known to A but kept secret. A signs M by r#s with $r \equiv g^k$ mod p and $s \equiv (M-er)k^{-1}$ mod p-1, for some k with (k, p-1)=1. Signatures can be verified by checking that $g^M \equiv y^r . r^s$ mod p, but cannot be forged by the assumed difficulty of computing indices (discrete logarithms, cf. [19]).

Many signature schemes are vulnerable to some form of chosen message attack. For example, in the Rabin scheme (A signs M by a square root of M modulo N_A=pq, provided M is a quadratic residu) an enemy C could ask A to sign a message $M \equiv r^2$ mod N_A with r known to C. With probability 1/2 A signs with the second essential root s of $x^2 \equiv M$ mod N_A, and C breaks the secret code of A because (r+s, N_A) is a nontrivial factor of N_A. More subtle attacks may enable forging of signatures without necessarily breaking the entire scheme. Goldwasser, Micali, & Yao [11] have devised two signature schemes (called "strong signature schemes") for which forging under a known message attack is provably equivalent to e.g. factoring or inverting RSA functions. "Strong" schemes may still collapse under different forms of attack. For example, if forging under known message attack is equivalent to factoring N_A, C might "run" the proof of factoring by forging and actually factor by asking A to sign any message C needs (interactive attack). It works if indeed the equivalence to forging is not further concealed. Goldwasser, Micali, & Rivest [9] have devised an ingenious strong signature scheme for which forging under interactive attacks is still as intractable as e.g. factoring.

6. Object authentication. Object authentication is required in large (distributed) operating systems when an object manager A must reinstantiate an object M that it held under control at some earlier moment. M may have migrated through the system (e.g. to background or off-line storage) while A occupied itself with other objects, and may have been "changed" by an enemy without A knowing about it. In some cases A might keep random test data in protected storage, to later validate an M as current and unaltered. More precisely, A determines an external representation R (a bitstring) of the object and its state and stores $D_A(R)$ with M before it relinquishes control over M. Tampering with M presumably changes R, but it is assumed that no one can forge a new signature. To authenticate M upon reinstantiation, A computes R and checks that the signature is consistent. Instead of D_A any secret encryption algorithm (like DES with a secret key) may be used.

Lindsay & Gligor [16] have proposed two refinements of the given "migration scheme". In one scheme the signature is computed as $D_A(R\#S)$ where S is a sequence of extra bits, e.g. checksum bits of the binary code of M. In another scheme R is stored with M as well but the signature is computed from a (secret) encryption of R.

References.

[1] DeMillo, R.A., N.A. Lynch, and M.J. Merritt, Cryptographic protocols, Proc. 14th Ann. ACM Symp. Theory of Computing, 1982, pp. 383-400.

[2] Denning, D.E., Cryptography and data security, Addison-Wesley Publ. Comp., Reading, Mass., 1982.

[3] Diffie, W., and M.E. Hellman, New directions in cryptography, IEEE Trans. Inform. Theory, IT-22(1976) 644-654.

[4] Diffie, W., and M.E. Hellman, Privacy and authentication: an introduction to cryptography, Proc. IEEE 67(1979) 397-427.

[5] ElGamal, T. A public-key cryptosystem and signature scheme based on discrete logarithms, IEEE Trans. Inform. Theory (to

appear).

[6] Evans, A., W. Kantrowitz, and E. Weiss, A user authentication scheme not requiring secrecy in the computer, C. ACM 17(1974) 437-442.

[7] Feistel, H., W.A. Notz, and J.L. Smith, Some cryptographic techniques for machine to machine data communications, Proc. IEEE 63(1975) 1545-1554.

[8] Goldwasser, S., and S. Micali, Probabilistic encryption, J. Comp. Syst. Sci. 28(1984) 270-299.

[9] Goldwasser, S., S. Micali, and R. Rivest, A "paradoxical" solution to the signature problem, Conf. Rec. 25th Ann. IEEE Symp. Foundations of Computer Science, 1984, pp. 441-448.

[10] Goldwasser, S., S. Micali, and P. Tong, Why and how to establish private codes on a public network, Conf. Rec. 23rd Ann. IEEE Symp. Foundations of Computer Science, 1982, pp. 134-144.

[11] Goldwasser, S., S. Micali, and A. Yao, Strong signature schemes, Proc. 15th Ann. ACM Symp. Theory of Computing, 1983, pp. 431-439.

[12] Ingemarsson, I., and C.K. Wong, A user authentication scheme for shared data based on a trapdoor one-way function, Inf. Proc. Lett. 12(1981) 63-67.

[13] Konheim, A.G., A one-way sequence for transaction verification, Report RC9147(#40034), IBM T.J. Watson Res Cntr, Yorktown Heights, NY, 1981.

[14] Konheim, A.G., Cryptography: a primer, Wiley & Sons, New York, NY, 1981.

[15] Lamport, L., Password authentication with insecure communication, C.ACM 24(1981) 770-772.

[16] Lindsay, B., and V. Gligor, Migration and authentication of protected objects, Report RJ2298(#31040), IBM Res. Lab., San Jose, CA., 1978.

[17] Merkle, R.C., Protocols for public-key cryptosystems, Proc. IEEE Symp. Security and Privacy, 1980, pp. 122-133.

[18] Merkle, R.C., and M.E. Hellman, Hiding information and signatures in trapdoor knapsacks, IEEE Trans. Inform. Theory, IT-24(1978)

525-530.

[19] Odlyzko, A.M., Discrete logarithms in finite fields and their cryptographic significance, preprint, AT&T Bell Labs, Murray Hill, NJ, 1984.

[20] Ong, H., C.P. Schnorr, and A. Shamir, An efficient signature scheme based on quadratic forms, Proc. 16th Ann. ACM Symp. Theory of Computing, 1984, pp. 208-216.

[21] Pohlig, S.C., and M.E. Hellman, An improved algorithm for computing discrete logarithms over GF(p) and its cryptographic significance, IEEE Trans. Inform. Theory, IT-24(1978) 106-110.

[22] Pollard, J., How to break the OSS signature scheme, unpubl., 1984.

[23] Purdy, G.B., A high-security log-in procedure, C.ACM 17(1974) 442-445.

[24] Rabin, M.O., Digitalized signatures, in: R.A. DeMilo et.al. (eds.), Foundations of secure computation, Acad. Press, New York, NY, 1978, pp. 155-168.

[25] Rabin, M.O., Digitalized signatures and public-key functions as intractable as factorization, Techn. Rep. 212, Lab. for Computer Sci, MIT, Cambridge, Mass., 1979.

[26] Rivest, R., A. Shamir, and L Adleman, A method for obtaining digital signatures and public-key cryptosystems, C.ACM 21(1978) 120-126.

[27] Shamir, A., A fast signature scheme, Techn. Memor. 107, Lab. for Computer Sci, MIT, Cambridge, Mass., 1978.

[28] Shamir, A., A polynomial time algorithm for breaking the basic Merkle-Hellman cryptosystem, Conf. Rec. 23rd Ann. IEEE Symp. Foundations of Computer Science, 1982, pp. 145-152.

[29] Williams, H.C., A modification of the RSA public-key encryption procedure, IEEE Trans. Inf. Theory, IT-26(1980) 726-729.

(References [4], [12], and [22] are not cited in the text.)