IJISET - International Journal of Innovative Science, Engineering & Technology, Vol. 3 Issue 6, June 2016
ISSN (Online) 2348 – 7968 | Impact Factor (2015) - 4.332
www.ijiset.com

Secure Signature Scheme for Network Key Compression System

[1]K.Ravikumar,[1] T.Tamilselvi
[1]Asst.professor, Dept.of.Computer science, Tamil University, Thanjavur-613010.
[2]Research Scholar, Dept.of.Computer Science, Tamil University, Thanjavur-613010.

Abstract:

A group signature scheme allows a group member to sign messages anonymously on behalf of the group. In the event of a dispute, a designated trusted entity can reveal the identity of the signer. Group signatures are claimed to have many useful applications such as voting and electronic cash. A number of group signature schemes have been proposed to-date. Though very easy to implement, these mechanisms are usually based on ad hoc techniques that lack a sound security analysis. In this way we classify notions even though polynomials reducible to each other as stronger or weaker in terms of concrete security. Next we provide concrete security analyses of methods to encrypt using a block cipher, including the most popular encryption method, This paradigm yields protocols much more enceinte than standard ones while retaining many of the advantages of provable security. We illustrate these gains for problems including encryption, signatures, and zero-knowledge proofs.

Keywords: MAC, RSA, SECURITY

I. INTRODUCTION

Group signature schemes are a relatively recent cryptographic concept introduced by Chum and van. In contrast to ordinary signatures they provide anonymity to the signer, a verifier can only tell that a member of some group signed. A group signature is publicly verifiable: it can be validated by anyone in possession of a group public key. However, group signatures are anonymous in that no one, with the exception of a designated group manager, can determine the identity of the signer.

Furthermore, group signatures are unlink able which makes it computationally hard to establish whether or not multiple signatures are produced by the same group member. Verifying the integrity and authenticity of information is a prime necessity in computer systems and networks. In particular, two parties communicating over an insecure channel require a method by which information sent by one party can be validated as authentic by the other. Most commonly such a mechanism is based on a secret key shared between the parties and takes the form of a Message Authentication Code. An encryption scheme enables Alice to send a message to Bob in such a way that an adversary Eve does not gain signi_cant information about the message content. This is the classical problem of cryptography. It is usually considered in one of two settings. In the symmetric private-key one, encryption and decryption are performed under a key shared by the sender and receiver. In the

asymmetric public-key setting the sender has some public information and the receiver holds some corresponding secret information.

II. A SECURE GROUP SIGNATURE SCHEME MUST SATISFY THE FOLLOWING PROPERTIES:

Correctness: Signatures produced by a group member using SIGN must be accepted by VERIFY.

Unforgeability: Only group members are able to sign messages on behalf of the group.

Anonymity: Given a valid signature of some message, identifying the actual signer is computationally hard for everyone but the group manager.

Unlinkability: Deciding whether two different valid signatures were computed by the same group member is computationally hard.

Excludability: Neither a group member nor the group manager can sign on behalf of other group members.1 A closely related property is that of no framing; it captures the notion of a group member not being made Responsible for a signature she did not produce.

Traceability: The group manager is always able to open a valid signature and identify the actual signer.

Motivation As observed in, to be truly useful, a group signature scheme must support dynamic group membership. Current state-of-the-art group signature schemes, support growing membership: new members can join without precipitating changes in the group public key or re-issuing group membership certificates for existing members. However, *shrinking* group membership has not been given the same attention.

2.1. Hash Functions

Recall our goal is to build secure message authentication functions from cryptographic hash functions in particular, from iterated hash functions. Arts clear obstacle is that while secret keys are an essential ingredient in a message authentication function, most cryptographic hash functions, and functions like, do not use keys at all. Therefore, a way to use cryptographic hash functions in conjunction with a key. Notions of Encryption For all complexity measures some probabilistic RAM model. We adopt the convention that \time" refers to the actual running time plus the size of the code relative to some programming language. Oracle queries are answered in unit time.

2.2. Identity Escrow Schemes

This section describes our new group signature scheme and tells how an identity escrow scheme can be derived. As mentioned in Section 2, many recent group signature schemes involve applying two types of non-group signature schemes: one for issuing certificates and one for actual group-signatures, respectively. The security of the former, in particular, is of immediate relevance because it assures, among other things, the coalition-resistance property of a group signature scheme. The reasoning for this assertion is fairly intuitive: Each group

member obtains a unique certificate from the group manager as part of JOIN where each certificate is actually a signature over a secret random message chosen by each member. As a coalition, all group members can be collectively thought of as a single adversary mounting an adaptive chosen message attack consisting of polynomials many instances of JOIN.

The above particular values of open and pad were chosen to have a very simple representation to simplify the function's speciation and minimize the potential of implementation errors, and to provide a high Hamming distance between the pads. The latter is intended to exploit the mixing properties attributed to the compression function underlying the hash schemes in use. These properties are important in order to provide computational independence between the two derived keys.

Correctness: Signatures produced by a group member using SIGN must be accepted by VERIFY.

Unforgeability: Only group members are able to sign messages on behalf of the group.

Anonymity: Given a valid signature of some message, identifying the actual signer is computationally hard for everyone but the group manager.

Unlink ability: Deciding whether two different valid signatures were computed by the same group member is computationally hard.

Excludability: Neither a group member nor the group manager can sign on behalf of other group members.

Traceability: The group manager is always able to open a valid signature and identify the actual signer. Therefore, any colluding subset of group members cannot generate a valid signature that the group manager cannot link to one of the colluding group members.

III. CORRECTNESS

The correctness condition pertains to signatures generated by honest group members, and asks the following: the signature should be valid; the opening algorithm, given the message and signature, should correctly identify the signer; the proof returned by the opening algorithm should be accepted by the judge. Formalizing these conditions in the dynamic group setting is more involved than formalizing them in a static setting in that these conditions must hold for all honest users under any schedule" under which these users join the group.

Shorter CRL: in our method a CRL is proportional to the number of revoked members. An ideal scheme would have a fixed-size or, at least, a shorter CRL logarithmic in the number of revoked members.

More efficient VERIFY: The cost of VERIFY is linear in the number of revoked members. It remains to be seen whether a constant- or sublinear-cost VERIFY can be devised.

Double discrete log: proofs using double discrete logarithms are inefficient, requiring

many exponentiations. For revocation to become truly practical, we need to devise either more efficient double discrete log proofs or different revocation structures that avoid double discrete log proofs altogether.

IV. GROUP SIGNATURE SCHEME

In this section we provide an overview of the ACJT scheme. Readers familiar with ACJT may skip this section with no loss of continuity. In its interactive, identity escrow form, the ACJT scheme is proven secure and coalition-resistant under the Strong RSA and DDH assumptions. The security of the non-interactive group signature scheme relies additionally on the Fiat-Shamir heuristic also known as the random oracle model.

Our new group signature scheme improves on the state-of-the-art exemplified by the scheme of Camenisch and Michels which is the only known scheme whose coalition-resistance is provable under a standard cryptographic assumption. In particular, our scheme's registration protocol (JOIN) for new members is an order of magnitude more efficient. Moreover, our registration protocol is statistically zero-knowledge with respect to the group member's secrets. In contrast, in the group member is required to send the group manager the product of her secret, a prime of special form, and a random prime; such products are in principle susceptible to an attack due to Coppersmith. Moreover, our scheme is provably coalition-resistance against an adaptive adversary, whereas for the scheme by Camenisch and Michels this holds only for a static adversary.

One can ask how much our assumptions on the underlying hash function can be further weakened and still have a simple construction of a secure MAC. Although we cannot answer this question in a formal way secure MAC functions can be built from the weaker assumption that the compression function is a one-way function, but the known constructions to achieve that are totally impractical, we can point out to two facts. First, by just assuming that the compression function is a MAC one cannot guarantee that the iterated function is a MAC.

That is clearly shown by the extension attacks discussed. In particular, this shows that one cannot just omit the outer application of NMAC and still get a secure MAC. As for basing the construction in collision-resistance only, we stress that this property also is insouciant to make the function a secure MAC. Indeed, one can construct examples of strong collision-resistant functions that are easily forgeable as MAC. Moreover, one can show this to hold for proposals of MAC functions based on hash schemes.

The main challenge in designing a practical group signature scheme is in finding a signature scheme for the certification of membership that allows the second signature scheme which is used to produce actual group signatures to remain efficient. Typically, the second scheme is derived using the Fiat-Shamir heuristic from a proof of knowledge of a membership certificate. Hence, the certification signature scheme must be such that the latter proof can be realized efficiently.

V. CONCLUSIONS

A very efficient and provably secure group signature scheme and a companion identity escrow scheme that are based on the strong RSA assumption. Their performance and security appear to significantly surpass those of prior art. Extending the scheme to a blind group-signature scheme or to split the group manager into a membership manager and a revocation manager is straight-forward. In addition, it uses the same key for prep ending and appending. The best analysis known for this type of functions is given in which show that when using divergent and independent keys for pretend and append the security of the function can be based on the pseudorandom properties of the underlying compression function.

VI. REFERENCES

[1] R. Atkinson, \Security Architecture for the Internet Protocol", IETF Network Working Group, RFC 1825, August 1995.

[2] R. Atkinson, \IP Authentication Header", IETF Network Working Group, RFC 1826, August 1995.

[3] J. Camenisch and M. Stadler. Efficient group signature schemes for large groups. In *Advances in Cryptology — CRYPTO'97*, vol. 1296 of *LNCS*, pp. 410–424, Springer-Verlag, 1997.

[4] M. Stadler. Publicly Verifiable Secret Sharing, In *Advances in Cryptology EUROCRYPT '96*, Springer-Verlag, 1996.

[5] J. Camenisch. Group signature schemes and payment systems based on the discrete logarithm problem. PhD thesis, vol. 2 of *ETH Series in Information Security an Cryptography*,

[6] Hartung-Gorre Verlag, Konstanz, 1998. ISBN 3-89649-286-1.

[7] D. Chaum and E. van Heyst. Group signatures. In *Advances in Cryptology EUROCRYPT '91*, vol. 547 of *LNCS*, pp. 257–265, Springer-Verlag, 1991.

[8] N. Bari´c and B. Pfitzmann. Collision-free accumulators and fail-stop signature schemes without trees. In Advances in Cryptology EUROCRYPT'97, vol. 1233 of LNCS, pp. 480–494, Springer-Verlag, 1997.

[9] M. Bellare and P. Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In 1st ACM Conference on Computer and Communication Security, pp. 62–73, ACM Press, 1993.