EFFICIENT ZERO-KNOWLEDGE IDENTIFICATION SCHEME FOR SMART CARDS

Thomas Beth
Universität Karlsruhe
Fakultät für Informatik
Institut für Algorithmen und Kognitive Systeme
Haid-und-Neu-Str. 7
Technologie-Fabrik
D-7500 Karlsruhe

ABSTRACT:
In this paper we present a Fiat-Shamir like authentication protocol for the El-Gamal Scheme.

1. Introduction

The invention of the El-Gamal Scheme [1] has provided another Public-Key-Cryptosystem besides the renowned RSA-System, for which in addition to the Key-Exchange feature both Public-Key-Encryption and Signature Schemes are available. The availability of fast exponentiation hardware for the fields $GF(2^n)$, cf [2], [3] makes this algorithm very attractive for implementation in high-speed-communications. The recent invention of the Fiat-Shamir Authentication Protocol [4] has again attracted wide attention to the RSA-Scheme.

The purpose of this note is to show that a similar type of authentication protocol is available for the El-Gamal-Scheme based on the Diffie-Hellman One-Way-Function, with complexity, and/or error-probability considerably reduced as compared to the Fiat-Shamir-Scheme.

C.G. Guenther (Ed.): Advances in Cryptology - EUROCRYPT '88, LNCS 330, pp. 77-84, 1988.

2. The Basic Protocol

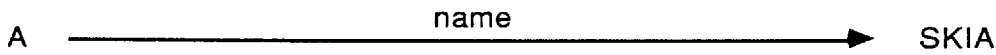
Suppose Alice (A) wants to authenticate herself to Bob (B). For this purpose A has visited a trusted authority, which for obvious reasons we shall call the Secure Key Issuing Authority (SKIA).

Initiation Phase

The SKIA possesses **secret** logarithms $x_1,...,x_m$, whose exponentiated values $y_j = \alpha^{x_j}$ are **public**. Here α is a primitive element of GF(q) known publicly. The SKIA also publishes the one-way-hashing function f.

Setting-up Phase

A goes to the SKIA, identifying herself by <name>.



Then the SKIA produces m identification numbers $ID_1,\ldots, ID_m$ for A by using the public (random) one-way-function f.

$$ID_j \leftarrow f(\text{name}, j)$$

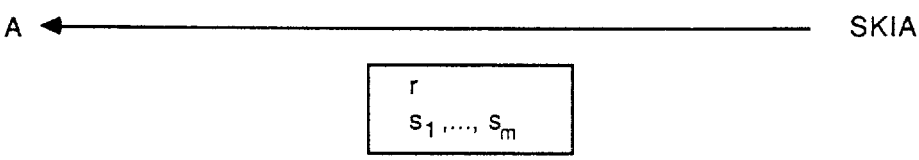
The SKIA chooses a (secret) random logarithm $k=k_A$ and forms

$$r \leftarrow \alpha^k.$$

The SKIA also determines m signatures s_j as solutions of

$$(ID) \quad x_j r + k s_j \equiv ID_j \bmod(q-1) \text{ for } j \in [1:m].$$

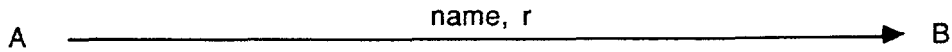
Eventually the SKIA issues a card (with secure memory)* to A.



*) *see sect. 5*

Authentication Phase (Protocol Auth)

A now approaches B identifying herself by her name and the parameter r.



B computes A´s identification numbers and the values $\rho_1, \ldots, \rho_m$

$$\text{For } j \in [1:m] : ID_j \leftarrow f(\text{name}, j)$$

$$\text{and } \rho_j \leftarrow y_j^r$$

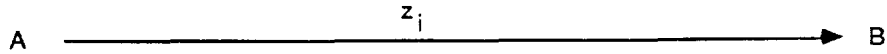
The following procedure is iterated for i=1 to h:

DO

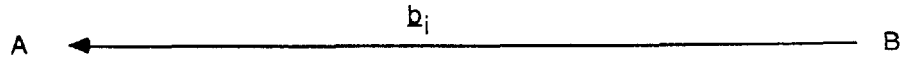
A chooses a random element $t_i \in \mathbb{Z}_{q-1}$, forms

$z_i \leftarrow r^{-t_i}$

and sends it to B



B chooses a random string $\underline{b}_i = (b_{ij}) \in R^m$ and sends it to A, where $R \subset \mathbb{Z}_{q-1}$ is a suitably chosen subset



A computes

$$u_i \leftarrow t_i + \sum_j b_{ij}\, s_j \mod (q-1)$$

and sends it to B

B computes

$$v_i \leftarrow \sum_j b_{ij} \cdot ID_j$$

$$\gamma_i \leftarrow \left(\prod_j \rho_j^{b_{ij}}\right) \cdot r^{u_i} \cdot z_i \cdot \alpha^{v_i}$$

OD

B accepts the authenticity of A if for all $i \in [1 : h]$ $\gamma_i = 0$. In this case we say that Protocol **Auth** ends successfully.

3. Analysis of the Protocol

3.1. Observation (Verification)

If A and B are genuine, the Protocol **Auth** ends successfully.

Proof: For all $i \in [1:h]$

$$\left(\prod_j \rho_j^{b_{ij}}\right)$$

can be computed by B based on B's knowledge.
From the definition we have

$$\left(\prod_j \rho_j^{b_{ij}}\right) = \prod_j y_j^{rb_{ij}} = \alpha^{r\sum_j x_j b_{ij}}$$

B can also compute r^{u_i} from u_i . Having received z_i, by definition

$$r^{u_i} \cdot z_i = r^{t_i + \sum_j b_{ij} s_j} \cdot r^{-t_i} . \quad \text{Thus } \alpha^{\sum_j b_{ij}(rx_j + ks_j)} = \alpha^{v_i} .$$

3.2. Observation (Correctness)

Assume, that A cannot compute El-Gamal signatures in polynomial time: If A is false, i.e. does not possess the signatures s_j, then the protocol **Auth** ends successfully with a false-right probability

$$p_1 \leq \frac{1}{|R|^{m \cdot h}}$$

Proof: As long as $|R|$ is small enough as compared to q^m (see remarks), A would have to guess the challenge vectors $\underline{b}_i$ in advance, analogously to the method described in the proof of lemma 2 by Fiat and Shamir [4].

3.3. Remark

The cheating method discussed in the proof of lemma 3.2 is only interesting if the size of choice space R^m for the $\underline{b}_i$ is small compared to the complexity of forging El-Gamal signatures, which itself is at most as hard as taking discrete logs, cf. sect. 4.

3.4. Lemma (Security)

For arbitrary q and h, with fixed m and $|R| \in 0((\log q)^w)$ for given $w \in \mathbb{N}$ the Protocol **Auth** is a Zero-Knowledge Protocol.

Proof: Following the papers by Berger/Kannan/Peralta [9] and Chaum, Evertse, van de Graaf [5] it can be seen that the size of the choice space R^m is the decisive parameter for the construction of a poly-time-simulator S for a cheating B: to guarantee a probability for S to "guess" the challenge $\underline{b}_i$ correctly in poly-time, we have to provide $|R|^m$ such that

$$\text{prob ("badluck")} = (1 - \frac{1}{|R|^m})^{\text{poly}(\log(q))} < c^{-\log q}$$

for some constant $c > 1$.

4. Practical Security Considerations

The system (ID) gives m linear equations for (m+1) unknowns (w.r.t. the assumption that the discrete log problem is unfeasible). As consequences we note:

(i) Not even A can forge new signatures.
(ii) The requirement of storing the signatures s_j in the secure memory of the card is only needed as protection against copying the card.
(iii) This requirement could be dropped if the one-way-hash function f (when stored on the card) could be employed by the card as a means of testing the user's identity before the card is authenticated. For this test several user features can be challenged, in each case requiring an interface between user and card, however!
(iv) To bring the security of the signatures closer to the level of the discrete log-problem it may be feasible to make the computation of the ID_j additionally dependent on the public random number r.

5. Implementation Aspects

In view of the demand for low cost designs of security processors for chip cards we suggest considering the following case for practical implementation :

$$q = 2^n,$$

where n should be suitably chosen, roughly in the interval $[2^9 : 2^{11}]$ depending on the required security. For these cases fast VLSI exponentiators have been suggested (Beth/Cook/Gollmann [3], Vanstone/ Mullin [2], Massey/Omura/Wang [7]).

For $q = 2^n$ the Discrete Log Problem can be solved in

$$0(\exp\ (c \cdot n^{1/3} \cdot (\log\ n)^{2/3})$$

steps [6]. Therefore a suitable amount of security can be guaranteed.

5.1. Tuning the Protocol

Using the fact that squaring is a field automorphism in $GF(2^n)$ we suggest to use the following refinements of the protocol in order to save on computational effort and required storage area as well as on the length h of the protocol:

*) Choose the random string only from binary words of weight less than w, i.e. choose b_{ij} equally distributed in

$$R = \{b \in Z_{q-1} \mid \text{wgt (binary (b))} \leq w\}$$

5.1.1.Corollary

With these additional restrictions if A is false the protocol ends successfully with probability

$$p_1 \leq \frac{1}{2^{t \cdot m \cdot h}}$$

where

$$t = \log \left(\sum_{i=0}^{w} \binom{n}{i} \right) .$$

**) Choosing the further simplification m = 1, the number of computational steps especially in computing

$$\gamma_i \leftarrow \rho^{\sum_{j=1}^{m} b_{ij}} \cdot r^{u_i} \cdot z_i^{v_i} - \alpha$$

is reduced considerably.

***) Combining (**) and (*) for w = 1 the exponent of ρ being a power of two requires a fast squaring operation only.

5.1.2. Technical Observation

With $m = 1$ and $h = 3$ and $\log q > 2^9$ the *Protocol* **Auth** allows an authentication procedure at a residual false-right error probability smaller than

10^{-8} for $w = 1$
10^{-15} for $w = 2$
10^{-22} for $w = 3$.

5.2. Conclusion

With **one** signature (m = 1) and a small number of iterations ($l \geq 3$) this protocol provides a security level appropriate to many smart card applications. In comparison to Fiat-Shamir`s protocol [4] the memory consumption on the smart card is considerably reduced for the proposed protocol, as the signature s_j and the number r only require approximately 64 Bytes each, and the representations of $GF(2^n)$-arithmetics can be compressed to considerably less bits. If the application requires only to authenticate the card through a trusted terminal, the public keys y need not to be stored on the card. Otherwise, the same protocol of course would be used by A to challenge B.

Note that an additional advantage to this protocol is provided by the fact, that based on purpose-made-algorithms the $GF(2^n)$-arithmetics can be carried out at a higher speed than modular arithmetic required for the Fiat-Shamir-Scheme.

In summary, the present scheme provides a user-friendly zero-knowledge authentication and signature protocol that offers itself as a small, fast and low cost verification tool for the use in token technology as it is presented by smart cards, intelligent tokens and other identification mechanisms.

Acknowledgement

The author is grateful to Dr. Ivan Damgård for his helpful critical remarks.

6. References

[1] El-Gamal: A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms, IEEE-IT-31, 469-472, 1985

[2] Vanstone, Mullin: Communication 1986, Cryptech, Waterloo, Ontario, Canada

[3] Beth, Cook, Gollmann: Architectures for Exponentiation in GF(2^n), Proceedings of Crypto 86, Santa Barbara, Springer LNCS 263, 302-310, 1987

[4] Fiat, Shamir: How to prove yourself: Practical solutions to identification and signature problems, Proceedings of Crypto 86, Santa Barbara, Springer LNCS 263, 186-194, 1987

[5] Chaum, Evertse, van de Graaf: An unproved Protocol for Demonstrating Possession of Discrete Logarithms and some Generalizations, Proceedings EUROCRYPT´87, Springer LNCS 304, 127-141, 1988

[6] Coppersmith: Fast Evaluation of Logarithms in Fields of Characteristic Two, IEEE-IT-30, 587-594, 1984

[7] Wang: Exponentiation in Finite Fields, Ph.D. dissertation , University of California, Los Angeles, 1985

[8] Goldwasser, S.: Micali, S.; Rackoff, C.: The Knowlege Complexity of Interactive Proof Systems, Proc. 17th ACM Symp. on Theory of Computing, 1985

[9] Berger, Kannan, Peralta: A Framework for the Study of Cryptographic Protocols, Proc. CRYPTO'85, Springer LNCS 218, 87-103