1997

How to prevent cheating in Pinch's scheme

G. Hossein

J. Pieprzyk

G. Chaundhry

Jennifer Seberry
University of Wollongong, jennie@uow.edu.au

How to prevent cheating in Pinch's scheme

Abstract

A modified protocol is proposed which prevents cheating in the Online multiple secret sharing scheme proposed by Pinch.

Disciplines

Physical Sciences and Mathematics

Publication Details

Hossein G, Pieprzyk J, Chaudhry G and Seberry J, How to prevent cheating in Pinch's scheme, Electronics letters, 33, 17 (1997) pp. 1453-54.

How to prevent cheating in Pinch's scheme

H. Ghodosi, J. Pieprzyk, G.R. Chaudhry and J. Seberry

Indexing term: Security of data

A modified protocol is proposed which prevents cheating in the Online multiple secret sharing scheme proposed by Pinch.

Introduction: The goal of a secret sharing scheme is to distribute a secret among a group of participants in such a way that the secret can be reconstructed by designated subsets of participants. An important issue in a secret sharing scheme is that the reconstruction procedure must provide the valid secret to all participants from an authorised set. That is, a dishonest participant must not be able to fool the others so that they obtain an invalid secret while the deceiver is able to get the valid secret. This problem has been discussed by several authors (see, for example, [1 – 3]).

Cachin [4] proposed a computationally secure scheme for online secret sharing with general access structures, where all the shares are as short as the secret. The scheme provides the ability to share multiple secrets and allows us to add participants dynamically, without having to redistribute new shares. These abilities are realised by storing additional authentic information at a publicly accessible location.

Pinch [5] points out that Cachin's scheme does not allow shares to be reused after the secret has been reconstructed without a further distributed computation protocol such as that of Goldreich *et al.* [6]. Pinch presents a modified protocol for computationally secure online secret sharing, based on the intractability of the Diffie-Hellman problem, where shares can be reused.

In this Letter, we show that Pinch's scheme is vulnerable to cheating. We then modify it to prevent cheating.

Pinch's scheme: M is a cyclic group of order q (written multiplicatively) in which the Diffie-Hellman problem is intractable (that is, given elements g, g^x and g^y in M, it is computationally infeasible to obtain g^{xy}) and f: $M \to G$ is a one-way function. The group operations in G and M are addition and multiplication modulo a large prime p.

The set P of participants is denoted by $P_1, ..., P_n$. Certain subsets $X \in 2^P$ are authorised to recover the secret K. The family of minimal authorised sets of participants is denoted by Γ (an authorised set X_1 is minimal if $X_1 \subseteq X_2$, and $X_2 \in \Gamma$ implies that $X_1 = X_2$).

Pinch's protocol works as follows : The dealer D, who knows the secret K, randomly chooses shares s_i (integers prime to q) for each participant $P_i \in P$ and transmits s_i over a secure channel to P_i. For each minimal authorised set $X \in \Gamma$, $|X| = t$, the dealer randomly chooses g_X to be a generator of M and computes

$$T_X = K - f\left(g_X^{\prod_{x \in X} s_x}\right)$$

and posts the pair (g_X, T_X) on the notice board. To recover the secret K, a minimal authorised set $X = \{P_1, ..., P_t\}$ of participants comes together and performs the following steps:

(i) member P_1 reads g_X from the notice board, forms $g_X^{s_1}$ and passes the result to P_2;

(ii) each subsequent member P_i, for $1 < i < t$, receives $g_X^{s_1 \ldots s_{i-1}}$ and raises this value to the power si to form $g_X^{s_1 \ldots s_i}$ which is passed to P_{i+1};

(iii) the final participant P_t receives $g_X^{s_1 \ldots s_{t-1}}$ and raises this value to the power s_t to form

$$V_X = g_X^{s_1 \ldots s_t} = g_X^{\prod_{x \in X} s_x}$$

(iv) on behalf of the access set X, member P_t reads T_X from the notice board and reconstructs K as $K = T_X + f(V_X)$.

If there are multiple secrets K_i to share, then it is possible to use the same one-way function f, provided that each entry on the notice board has a fresh value of g_X attached.

Pinch also has a variant proposal which, according to him, avoids the necessity for the first participant P_1 to reveal $g_X^{s_1}$ at step (i). P_1 takes r modulo q at random and forms $g_X^{rs_1}$ and passes the result to P_2, and so on. At the end of protocol, P_t returns the computed value $g_X^{rs_1 \ldots s_t}$ to P_1 which computes

$$V_X = (g_X^{rs_1 \ldots s_t})^{r^{-1}} \pmod p$$

where r^{-1} is the inverse of r, that is $r \times r^{-1} = 1 \bmod q$ (the other parts of the protocol are the same as the original protocol).

How to cheat: Pinch's scheme has a major disadvantage in that it is vulnerable to cheating. In this scheme, a dishonest participant $P_i \in X$ may contribute his fake share $s'_i = \alpha s_i$, where α is a random integer modulo q. Since every participant of an authorised set X ($|X| = t$) has access to the final result $g_X^{s_1 \ldots s'_i \ldots s_t}$, the participant P_i can calculate the value

$$\left(g_X^{s_1 \ldots s'_i \ldots s_t}\right)^{\alpha^{-1}} = g_X^{s_1 \ldots s_i \ldots s_t} = g_X^{\prod_{x \in X} s_x} = V_X$$

and hence the correct secret as in Pinch's scheme, while the other participants calculate an invalid secret.

Remark: Cachin's scheme is secure against this form of cheating, because in his scheme participants have no access to $V_X = \Sigma_{x \in X} s_x$. Thus, if a participant contributes a fake share, he cannot modify the result to obtain the valid secret (the function f is assumed to be one-way).

How to detect cheating: Suppose in the initialisation phase of the Pinch scheme, the dealer publishes $g_X^{V_X}$ (corresponding to every authorised set X). Let the reconstruction protocol be the same as in the original Pinch scheme and let V'_X be the final result. Every participant $x \in X$, can verify whether

$$g_X^{V_X} \stackrel{?}{=} g_X^{V'_X}$$

If the verification fails, then cheating has occurred in the protocol and thus the computed secret is not valid. This protocol detects cheating but does not detect the cheat(s) and also cannot prevent cheating. That is, the cheater(s) obtain the secret while the others gain nothing.

How to prevent cheating: Let $C = \Sigma_{x \in X} g_X^{s_x}$ correspond to an authorised set X. We assume that in the initialisation phase of the Pinch scheme the dealer also publishes $C_X = g_X^C$. Note that this extra public information gives no useful information about the secret or about participants' shares. Otherwise, we could solve the discrete logarithm in M and easily solve the Diffie-Hellman problem.

Let X be an authorised set of participants. At the reconstruction phase, every participant $P_i \in X$ computes $g_X^{s_i}$ and broadcasts it to all participants in the set X. Thus, every participant $P_i \in X$ receives $t - 1$ values $g_X^{s_j}$ corresponding to all $P_j \in X$, $P_j \neq P_i$. Each participant computes C and verifies $C_X \stackrel{?}{=} g_X{}^C$. If the verification fails, then the protocol stops. Let participants agree to perform computation in the cycle $P_1, ..., P_t$. If the check $C_X \stackrel{?}{=} g_X{}^C$ is successful, then each participant P_i ($i = 1, ..., t$) knows the true value $G_X^{s_{i-1}}$ of its predecessor (P_t is the predecessor of P_1). So participant P_i (i = 1, ..., t) initiates the protocol by computing $(g_X^{s_{i-1}})^{s_i}$ and passing it to P_{i+1}. The protocol proceeds as in the Pinch scheme and ends at P_{i-2}. In this way, the participant P_{i-1} cannot directly contribute to the computation which started by P_i.

Let there exist only one cheat, P_i ($1 \leq i \leq t$) in the system. If P_i cheats, the computation initiated by P_{i+1} must be correct (the correctness can be verified as $g_X{}^{V_X} \stackrel{?}{=} g_X^{V'_X}$, where V'_X is the result obtained by P_{i-1}). That is, although cheating has occurred, the honest set of participants can recover the secret.

If there exists a group of collaborating cheats, then in the above protocol each participant must play (simultaneously) the role of P_1 for every other t_1 participants in the set X. Although the number of computations increases rapidly, before completing the protocol any possible cheating will be detected and the protocol will be stopped. At stage j ($1 < j < t$), for every set of j (out of t) participants (without loss of generality, for $P_1, ..., P_j$) there will be $j!$ different computations of $g_X^{s_1 \ldots s_j}$. Hence, inequality of these values indicates cheating in the system. Moreover, assuming the majority of participants are honest (this is a reasonable assumption for any robust secret sharing scheme) the minority of participants who obtain values different from the common value in stage j, are the cheats.

Remark: A group of m cheats can cheat the system at the first stage, that is, they can contribute with fake shares such that the resulting C is equal to the original one. However, the above protocol detects their cheating in next stages (there are at least $2m + 1$ stages for such a set of collaborating participants).

25 June 1997

Electronics Letters Online no: 19970980

H. Ghodosi, J. Pieprzyk, G.R. Chaudhry and J. Seberry (*Center for Computer Security Research, Department of Computer Science, University of Wollongong, NSW 2522, Australia*)

References

1 TOMPA, M., and WOLL, H.: 'How to share a secret with cheaters', *J. Cryptol.*, 1988, **1**, (2), pp. 133–138

2 BRICKELL, E., STINSON, D., and GOLDWASSER, S.: 'The detection of cheaters in threshold schemes' *in* GOLDWASSER, S. (Ed.): Advances in Cryptol. - Proc. CRYPTO '88, Vol. 403 of Lecture Notes in Comput. Sci., (Springer-Verlag, 1990), pp. 564–577

3 LIN, H., HARN, L., IMAI, R.R.H., and MATSUMOTO, T.: 'A generalised secret sharing scheme with cheater detection' *in* IMAI, R.R.H., and MATSUMPTO, T. (Eds.): Advances in Cryptol. - Proc. ASIACRYPT '91, Vol. 739 of Lecture Notes in Comput. Sci., (Springer-Verlag, 1993), pp. 149–158

4 CACHIN, C., and BOYD, C.: 'On-line secret sharing', *in* BOYD, C. (Ed.): Cryptography and Coding: 5th IMA Conf., 1995, (Institute for Theoretical Computer Science, ETH Zürich), Vol. 1025 of Lecture Notes in Comput. Sci., (Springer-Verlag, 1995), pp. 190–198

5 PINCH, R.: 'Online multiple secret sharing', *Electron. Lett.*, 1996, **32**, pp. 1087–1088

6 GOLDREICH, O., MICALI, S., and WIGDERSON, A.: 'How to play any mental game'. Proc. Nineteenth ACM Symp. Theory of Comput., STOC, 25–27 May 1987, pp. 218–229