Secret Sharing with Reusable Polynomials

Liqun Chen*, Dieter Gollmann, Chris J. Mitchell and Peter Wild

Information Security Group,
Royal Holloway, University of London,
Egham, Surrey TW20 0EX, UK
Email: {liqun, dieter, cjm}@dcs.rhbnc.ac.uk;
P.Wild@alpha1.rhbnc.ac.uk

Abstract. We present a threshold secret sharing scheme based on polynomial interpolation and the Diffie-Hellman problem. In this scheme shares can be used for the reconstruction of multiple secrets, shareholders can dynamically join or leave without distributing new shares to the existing shareholders, and shares can be individually verified during both share distribution and secret recovery.

1 Introduction

Since Blakley [2] and Shamir [13] introduced the concept of secret sharing in 1979, a number of secret sharing schemes have been proposed with varying properties meeting diverse application requirements. The basic idea of secret sharing is that a *dealer* distributes partial information (a *share*) about a *secret* to each of a set of *shareholders* such that only authorised subsets of the shareholders can reconstruct the secret. For the purposes of this paper, the main properties of interest here are the following.

- *Perfect security or computational security.* A secret sharing scheme is perfectly secure if an unauthorised subset of shareholders can obtain no information about the secret [14], and it is computationally secure if it is computationally infeasible to determine the secret from such a subset [8, 16].
- *Verifiable shares.* We consider two kinds of share verification. The first is that during share distribution each shareholder can verify his received share to detect a dishonest or failed dealer [3, 10, 11]. The second is that during secret reconstruction a forged share contributed by a cheating shareholder can be detected by the other shareholders [7, 15].
- *'Online' shareholders.* Shareholders can dynamically join or leave the sharing group without having to redistribute new shares secretly to the existing shareholders [1, 4, 12, 13].
- *Reusable shares.* Shares can be reused after the shared secret has been reconstructed, although there may be a modification allowing a predetermined number of multiple secrets to be reconstructed in a specified order [7, 9, 12, 15, 16].

* The work of this author has been funded by the European Commission under ACTS project AC095 (ASPeCT).

This paper presents two variants of a new threshold secret sharing scheme holding the above properties. The mechanism is based on several previously proposed schemes, namely those of Shamir [13], Pedersen [10], Cachin [4] and Pinch [12]. We now briefly summarise the basic features of these schemes.

Shamir [13] proposed a polynomial interpolation based (t, m)-threshold secret sharing scheme, in which no group of fewer than t from a set of m shareholders can obtain any information about a secret, and any group of at least t shareholders can compute the secret. While keeping t fixed, a share can be dynamically added or deleted without affecting the other shares as long as the total number of shareholders remaining is at least t. Deleting a share in this context means that it is made completely inaccessible, even to the owning shareholder. This scheme does not allow the shares to be reused after the secret has been reconstructed.

Pedersen [10] adds a distributed prover to the Shamir scheme, so that the m shares (one for each shareholder) can be verified by the m shareholders.

Cachin [4] presents a protocol with 'online shareholders', in which each of a set of shares is chosen randomly. Adding or deleting a share does not affect the other shares, provided that additional authentic, but not secret, information is posted in a publicly accessible central location. This proposal also does not allow multiple use of the shares.

Pinch [12] gives a modification of Cachin's scheme which allows for an arbitrary number of secrets to be reconstructed without having to redistribute new shares.

Both the Cachin and Pinch protocols are designed for general access structures. The number of authentic messages in a publicly accessible noticeboard is proportional to the number of minimal shareholder sets trusted to recover the secret. If this trusted set number is large, a large noticeboard will be needed. In particular, a (t, m)-threshold scheme is an important special case of a general access structure, where the trusted shareholder sets are all combinations of the m shareholders taken t at a time, of which there are $\binom{m}{t}$, which is of order m^t for t small relative to m. Thus, when using the Cachin or the Pinch scheme to implement (t, m)-threshold secret sharing, a potentially large public noticeboard of size $\binom{m}{t}$ must be maintained.

In this paper we modify the Pinch scheme to obtain a threshold secret sharing scheme by using a reusable polynomial function. The main advantage of the new scheme is that a large noticeboard is not required. In addition, this mechanism makes use of publicly accessible provers to give the property of share verification: i.e., each share is verifiable during both share distribution and secret reconstruction.

2 Notation and assumptions

Let p and q be large primes such that q divides $p-1$. Let G be the additive group of integers modulo p, and let M be a subgroup of order q of the multiplicative group of non-zero integers modulo p (and hence, since q is prime, M will be cyclic). Let g be a generator of M, and $h : M \rightarrow G$ be a one-way function

such that it is computationally infeasible to recover x from $y = h(x)$. These parameters will be used throughout this paper.

Suppose that in M the Diffie-Hellman problem [6] is intractable: that is, given elements g, g^x and g^y in M and the modulus p, it is computationally infeasible to obtain g^{xy}. This implies in particular the intractability of the corresponding discrete logarithm problem: i.e. given g and g^x in M and p it is computationally infeasible to recover the exponent x.

All protocols in this paper are carried out between a dealer D and a set of shareholders $P = \{P_1, ..., P_m\}$. We suppose that the communication channels between D and the m shareholders provide origin authentication and data integrity for the retrieval of noticeboard information. The above parameters, p, q, g, h, G and M, are publicly known to D and P.

3 Outline of the Pinch scheme

In this protocol, certain subsets $X \subseteq P$ are trusted to recover the secret K. The family of minimal trusted sets is denoted as Γ. The protocol makes use of a noticeboard where messages can be written by D and read by all shareholders.

D initiates the protocol by randomly choosing secret shares s_i $(1 \leq i \leq m)$, integers satisfying $1 < s_i < q$, for each shareholder P_i and then transmits s_i secretly to P_i. Alternatively, D and P_i engage in a key-exchange protocol such as Diffie-Hellman [6] to exchange a suitable s_i.

For each minimal trusted set $X \in \Gamma$, D randomly chooses g_X, a generator of M, then computes

$$T_X = K - h(g_X^{\prod_{x \in X} s_x}) \pmod{p},$$

and posts the pair (g_X, T_X) on the noticeboard.

To recover the secret K, a minimal trusted set X of shareholders comes together (without loss of generality suppose $X = \{P_1, ..., P_t\}$). They form

$$U_X = g_X^{\prod_{i=1}^{t} s_i} = g_X^{\prod_{x \in X} s_x} \pmod{p}$$

via a chain from P_1 to P_t: i.e., each shareholder P_i contributes his secret share s_i as a power. Finally P_t obtains U_X and then reconstructs K as

$$K = T_X + h(U_X) \pmod{p}.$$

Note that in this scheme shares are not revealed during secret reconstruction, and $h(U_X)$ is a fresh value given a fresh g_X. However, the one-way function h and the shares can be reused for an arbitrary number of secrets reconstructed by different trusted sets, provided that D posts a fresh pair (g_X, T_X) on the noticeboard for each secret and for each particular trusted minimal set.

As mentioned earlier, the disadvantage of this scheme is that it needs a large noticeboard when Γ is a big family. In particular, if this protocol is used to implement (t, m)-threshold secret sharing, then the number of elements in Γ is $\binom{m}{t}$, which grows very quickly and which is of the order of m^t for m large

relative to t. In the next section we present a modification of this protocol, based on a reusable polynomial, to provide a threshold scheme which does not need a large noticeboard.

4 A new secret sharing scheme

We present two versions of a new scheme with different requirements for the handling of invalidated shares. The first version works on the assumption either that the shareholders are trusted not to use a share that has been invalidated, or that invalidated shares are completely deleted. The second version does not rely on this assumption.

4.1 Version 1

Let s (an element of $\mathbb{Z}_q$) be a 'long term secret', which is shared by the m shareholders. Let K (an element of $\mathbb{Z}_q$) be a 'short term secret', which will be reconstructed by any group of at least t ($t \leq m$) shareholders. The shares of s can be used for the reconstruction of multiple 'short term secrets'.

D initially distributes partial information (a share) about s to each of the m shareholders, in a way based on the Shamir scheme [13] and the Pedersen scheme [10].

D chooses a degree $t-1$ polynomial

$$f(x) = a_0 + a_1 x + ... + a_{t-1} x^{t-1}$$

over $\mathbb{Z}_q$ satisfying $a_0 = s$, and computes each share $s_i = f(x_i)$ $(1 \leq i \leq m)$. Here $x_i \in \mathbb{Z}_q - \{0\}$ ($x_i \neq x_j$, for $i \neq j$) is public information about P_i. D then sends the share s_i secretly to P_i and broadcasts a verification sequence

$$V = (g^{a_0}, g^{a_1}, ..., g^{a_{t-1}}),$$

each value computed modulo p, to all m shareholders.

Each P_i computes

$$v_i = \prod_{j=0}^{t-1} (g^{a_j})^{(x_i)^j} (\text{mod } p),$$

and verifies whether

$$v_i = g^{s_i} (\text{mod } p).$$

If this does not hold then P_i broadcasts s_i and stops. Otherwise P_i accepts the share.

Note that there are a number of cryptographic techniques to distribute shares, e.g. Sun and Shieh [15] use the Diffie-Hellman scheme [6] to do so.

For each 'short term secret' K, D chooses a random nonce $r \in \mathbb{Z}_q$ (which must be used only once and cannot be predicted in advance or guessed in the

future by any shareholders or intercepting third parties), and then broadcasts g^r, T_r and another verification sequence V_r to all m shareholders, where

$$T_r = K - h(g^{rs}) \pmod{p},$$

and

$$V_r = (g^{g^{rs_1}}, ..., g^{g^{rs_m}}),$$

each value computed modulo p.

To recover the secret K, any t shareholders (without loss of generality suppose they are P_1, ..., P_t) join together. Each shareholder P_i $(1 \leq i \leq t)$ computes and contributes to all other shareholders g^{rs_i}. Based on V_r, each contribution can individually be verified by the other shareholders. After all the contributions have been checked successfully, the t shareholders each compute $g^{rs_ib_i}$, where

$$b_i = \prod_{\substack{j=1 \\ j \neq i}}^{t} \frac{x_j}{x_j - x_i} \pmod{q}.$$

They then form

$$g^{rs} = g^{rs_1b_1 + ... + rs_tb_t} = \prod_{i=1}^{t} g^{rs_ib_i} \pmod{p},$$

and reconstruct the secret K as

$$K = T_r + h(g^{rs}) \pmod{p}.$$

Note that the polynomial f, one-way hash function h, 'long term secret' s and shares s_i $(1 \leq i \leq m)$ can all be reused for recovering multiple 'short term secrets', provided that D broadcasts the fresh values of g^r, T_r and V_r for each new 'short term secret'.

We make the following remarks concerning the above protocol.

- In order to allow shareholders to join and leave dynamically, we must assume that any deleted shares will no longer be acceptable for the reconstruction of any secret. If a shareholder were to leave and his share were to become known to another shareholder (or a third party), then a threshold of only $t-1$ shareholders (possibly with the help of that third party) could reconstruct a 'short term secret' by using the invalidated share with their $t - 1$ valid shares. The above protocol requires either that the shareholders are trusted not to use a share that has been invalidated, that invalidated shares are completely deleted, or that there is some (physical) constraint on the shares to ensure that invalidated shares cannot be accepted for the reconstruction of any secret.

- ◇ The whole point of (t, m)-threshold secret sharing is that any t or more shareholders are collectively trustworthy. In this protocol, it is assumed that as soon as any t or more shareholders are no longer considered trustworthy, D must terminate use of the polynomial. Although this is not an unreasonable assumption, there may be some application environments where D wants to continue to use the polynomial after deleting a group of t or more untrustworthy shareholders.

In the next subsection we will present a modified version, that can avoid the possibility of reduction of the threshold when some shares are invalidated, and that can be used under the condition of at most a predetermined number of shares having been invalidated.

4.2 Version 2

In this version, we make use of a degree $t+u-1$ polynomial instead of a degree $t-1$ one. Let u be an upper bound for the number of the shares that can be deleted by D without compromising the shared secrets.

To distribute shares of s, D chooses a degree $t+u-1$ polynomial

$$f(x) = a_0 + a_1 x + ... + a_{t+u-1} x^{t+u-1}$$

over $\mathbb{Z}_q$ satisfying $a_0 = s$ ('long term secret'), and computes each share $s_i = f(x_i)$ $(1 \leq i \leq m+u)$. Here $x_i \in \mathbb{Z}_q - \{0\}$ $(x_i \neq x_j$, for $i \neq j)$ is public information. The element x_{i+u} is publicly associated with shareholder P_i. D then secretly stores the u shares s_1, ..., s_u and sends the remaining shares s_{u+i} $(1 \leq i \leq m)$ to P_i $(1 \leq i \leq m)$, one for each shareholder. After that, D broadcasts a verification sequence

$$V = (g^{a_0}, g^{a_1}, ..., g^{a_{t+u-1}}),$$

each value computed modulo p, to all m shareholders.

Each P_i computes

$$v_i = \prod_{j=0}^{t+u-1} (g^{a_j})^{(x_{u+i})^j} (\mathrm{mod}\ p),$$

and verifies whether

$$v_i = g^{s_{u+i}} (\mathrm{mod}\ p).$$

If this does not hold then P_i broadcasts s_{u+i} and stops. Otherwise P_i accepts the share.

For each 'short term secret' K, D chooses a random nonce $r \in \mathbb{Z}_q$ (as mentioned before, it must be used only once and cannot be predicted in advance or guessed in the future by any shareholders or intercepting third parties). D then distributes some public information in one of two different ways depending on whether any shares are invalidated.

When no share is invalidated, D broadcasts g^r, W_r, T_r and V_r to all m shareholders, where

$$W_r = (g^{rs_1}, ..., g^{rs_u}),$$

each value computed modulo p;

$$T_r = K - h(g^{rs}) \pmod{p};$$

and

$$V_r = (g^{g^{rs_{u+1}}}, ..., g^{g^{rs_{u+m}}}),$$

each value computed modulo p.

The secret shares $s_1, s_2, ..., s_u$ kept by D serve as placeholders for shares that may be invalidated in the future. In case a share, s_{u+i} $(1 \leq i \leq m)$, has to be invalidated, one of the secret shares will be dismissed and replaced by the invalidated share. More precisely, assume that L $(1 \leq L \leq u)$ shares, say s'_1, ..., s'_L in s_{u+1}, ..., s_{u+m}, are invalidated. D replaces s_1, ..., s_L with these invalid shares and uses these values in the computation of W_r. D then broadcasts g^r, W_r, T_r, V_r (without $g^{g^{rs'_1}}$, ..., $g^{g^{rs'_L}}$) and the corresponding variables x'_1, ..., x'_L which have replaced x_1, ..., x_L, i.e. the positions of the invalidated shares.

To recover the secret K, any t shareholders, say P_1, ..., P_t, join together. Each shareholder P_i $(1 \leq i \leq t)$ computes and contributes to all other shareholders $g^{rs_{u+i}}$. Based on V_r, each contribution is then individually verified by the other shareholders. After all the contributions are checked successfully, these t shareholders compute $g^{rs_i b_i} (1 \leq i \leq t+u)$, where

$$b_i = \prod_{\substack{j=1 \\ j \neq i}}^{t+u} \frac{x_j}{x_j - x_i} \pmod{q}.$$

They then form

$$g^{rs} = \prod_{i=1}^{t+u} g^{rs_i b_i} \pmod{p},$$

and reconstruct the secret K as

$$K = T_r + h(g^{rs}) \pmod{p}.$$

We make the following remarks concerning the above protocol.

- The choice of the value u is dependent on the application requirements. We here consider two possible conditions affecting the choice of u. The first requires that the number of the shares currently valid at any time during the working of the scheme will not be less than t. In this case, the upper bound on the value u is $m' - t$, where m' is the number of valid shares at the beginning of the scheme. The second requires that the possible number of invalidated shares must never be larger than u as long as the number of valid shares remaining is at least t. In this case, the lower bound on the value u is $m'' - t$, where m'' is an upper bound for the total number of the shares

issued by D to P. Note that the complexity of computation of a secret in this version will grow quickly when the value u becomes very large. Thus in practice, u would be set at an appropriate level for the application, and the scheme re-initialised should the number of invalidated shares exceed u.

- ◇ It is clear that if an authorised group (t or more) of shareholders do not follow the protocol specifications correctly, the protocol will fail. For example, if they reveal their shares to each other, then each member has access to the secret equivalent to that of the authorised subgroup. In particular, an authorised group of t shareholders, say P_1, P_2, ..., P_t, can pool their secrets and calculate a value $s' = \sum_{i=1}^{t} b_i s_i$, which each one can use to calculate a 'short term secret' (using only public information) when D makes a broadcast.

5 Analysis of the new scheme

The security of the scheme. The proposed scheme has the following security properties.

1. By using precisely the same arguments as used to prove the corresponding statements for the Shamir scheme, we can show that the scheme meets the basic requirements for (t, m)-threshold secret sharing: i.e., during the reconstruction of a 'short term secret' no group of fewer than t shareholders can obtain any information about g^{rs}, and any t shareholders can compute this value.
2. By using precisely the same arguments as used to prove the corresponding statements for the Pinch scheme, we can show that the 'short term secrets' are computationally secure, assuming the intractability of the Diffie-Hellman problem in M. The use of the one-way function h ensures that no attack based on the multiplicative property of the group M will succeed.

Shareholder addition/deletion. Based on polynomial interpolation, this scheme allows shareholders to be dynamically added or deleted without having to redistribute new shares secretly to the existing shareholders. If a shareholder, say P_l, enters, D sends s_l (for Version 1) and s_{u+l} (for Version 2) secretly to P_l and informs all shareholders about x_l (for Version 1) and x_{u+l} (for Version 2). As mentioned in the protocol description, Version 1 works on the assumption (without a physical solution) that any deleted shares will no longer be acceptable for the reconstruction of any secret. Version 2 does not rely on this assumption, but makes use of a disenrollment scheme that allows at most u shareholders to quit without affecting the other shares while maintaining the threshold value t. During dynamic changes to the set of shareholders, in Version 1, D shares a 'short term secret' as before, and in Version 2, D only needs to compute renewed values of W_r by using the renewed shares. However, in the Cachin protocol and the Pinch protocol, D has to re-post the pair (g_X, T_X) for each new minimal trusted set X involving any added and deleted shareholders on the noticeboard.

For a detailed discussion on threshold schemes with disenrollment, the reader is referred to [1]. Charnes, Pieprzyk and Safavi-Naini [5] propose a secret sharing

scheme with disenrollment capability. Although sharing only a single secret, their scheme uses the idea of 'initial conditions' which are converted to working shares by exponentiation of a primitive element. If a working share is invalidated then a new set of working shares is created by distributing a new primitive element. They make use of a combiner to maintain the reconstruction of the secret. To cope with the possibility that an 'initial condition' may be compromised they propose using a family of secret sharing schemes. A dealer selects and contributes l independent Shamir threshold schemes to n shareholders. The reconstruction of a secret starts by using the first threshold scheme. If an 'initial condition' in the first threshold scheme is compromised, all the shareholders switch to a new threshold scheme. In a similar way to the scheme proposed in this paper, the Charnes et al. method can tolerate the loss of a specified number of shares. However it is not suitable for the case where a number of shareholders can quit and then all of the shares held by them may be compromised.

Polynomial reuse. If t is kept fixed, this scheme allows multiple use of the polynomial f, because no polynomial value is revealed during secret reconstruction.

Suppose a hierarchical access scheme is required, i.e. some shareholders are allowed to have greater rights to access a secret than the other shareholders. We can use the same polynomial of degree $t-1$ for Version 1 and $t+u-1$ for Version 2, but let some shareholders hold more than one share, as is suggested by Shamir [13].

In Version 2, as soon as more than u shares of a polynomial are expected to be invalid (e.g., when the corresponding shareholders will leave), D must terminate use of the polynomial in order to prevent an unauthorised group from reconstructing a secret by using unauthorised shares.

Share verification. The scheme makes use of two verification sequences (V and V_r) so that each share can be verified when it is accepted by a shareholder as his private share and when it is contributed by the shareholder for reconstructing a secret. Assuming the intractability of the Diffie-Hellman problem, these two sequences will not compromise the security of shares.

Sun and Shieh [15] propose a polynomial based secret sharing scheme with share verification by broadcasting a verification sequence, in which the Diffie-Hellman scheme is used to distribute shares. This protocol does not allow multiple use of a polynomial, i.e., for each renewed secret D has to choose a new polynomial. As Hwang and Chang point out [7], it is difficult for D to construct many polynomials with the same degree $t-1$ such that all the polynomial values are fresh. Furthermore, it will result in a weakness of the protocol, namely that by comparing an old share with each value of a current verification sequence, a shareholder may find that he holds more than one valid share. Hwang and Chang then propose a modification to improve the share verification technique, but again, their protocol does not allow multiple use of a polynomial. The scheme proposed in this paper has no such problem. The reason is that not many polynomials with the same order and the same variables are needed.

Other properties. It is easy to see that this scheme has the following further

properties. Each share is the same size as one of the secrets. The number of 'short term secrets' which can be reconstructed by reusing a single polynomial is only limited to the number of fresh values r. Because the values r are randomly chosen in $\mathbb{Z}_q$, the number of possible choices can be very large. The new scheme broadcasts of the order of m elements whereas the Cachin and Pinch schemes require a noticeboard of size $\binom{m}{t}$ to implement (t, m)-threshold secret sharing.

6 Conclusions

We have proposed a new (t, m)-threshold secret sharing scheme which permits the reconstruction of multiple secrets and dynamic changes to the set of shareholders without distributing new shares to current members of the scheme, and which allows the verification of individual shares during the share distribution and secret reconstruction.

7 Acknowledgment

The authors would like to thank the anonymous referees for their comments and bringing references [5] and [11] to our attention.

References

1. Blakley, B., Blakley, G.R., Chan, A.H., Massey, J.L.: Threshold schemes with disenrollment. In E.F. Brickell, editor, Lecture Notes in Computer Science 740, Advances in Cryptology - Crypto '92 (Springer-Verlag, Berlin, 1993) 540–548
2. Blakley, G.R.: Safeguarding cryptographic keys. In the Proceedings of AFIPS 1979 NCC, Vol. 48, Arlington,Va. (1979) 313–317
3. Brickell, E.F., Stinson, D.R.: The detection of cheaters in threshold schemes. In S. Goldwasser, editor, Lecture Notes in Computer Science 403, Advances in Cryptology - CRYPTO '88 (Springer-Verlag, Berlin, 1988) 564–577
4. Cachin, C.: On-line secret sharing. In C. Boyd, editor, Lecture Notes in Computer Science 1025, 5th IMA Conference on Cryptography and Coding (Springer-Verlag, Berlin, 1995) 190–198
5. Charnes, C., Pieprzyk, J., Safavi-Naini, R.: Conditionally secure secret sharing schemes with disenrollment capability. In Proceedings of the 2nd ACM Conference on Computer and Communications Security (Fairfax, Virginia, USA, 1994) 89–95
6. Diffie, W., Hellman, M.E.: New directions in cryptography. IEEE Transactions on Information Theory **22** (1976) 644–654
7. Hwang, S., Chang, C.: A dynamic secret sharing scheme with cheater detection. In Lecture Notes in Computer Science 1172, ACISP '96 (Springer-Verlag, Berlin, 1996) 48–55
8. Krawczyk, H.: Secret sharing made short. In Lecture Notes in Computer Science 773, Advances in Cryptology – CRYPTO '93 (Springer-Verlag, Berlin, 1993) 136–146
9. Laih, C.S., Harn, L., Lee, J.Y., Hwang, T.: Dynamic threshold scheme based on the definition of cross-product in an n-dimensional linear space. Journal Information Science and Engineering **7** (1991) 13–23

10. Pedersen, T.P.: Distributed provers with applications to undeniable signatures. In D. W. Davies, editor, Lecture Notes in Computer Science 547, Advances in Cryptology - Eurocrypt '91 (Springer-Verlag, Berlin, 1991) 221–238
11. Pedersen, T.P.: Non-interactive and information-theoretic secure verifiable secret sharing. In J. Feigenbaum, editor, Lecture Notes in Computer Science 576, Advances in Cryptology - Crypto '91 (Springer-Verlag, Berlin, 1992) 129–140
12. Pinch, R.G.E.: On-line multiple secret sharing. Electronics Letters **32** (1996) 1087–1088
13. Shamir, A.: How to share a secret. Communications of the ACM **22** (1979) 612–613
14. Shannon, C.E.: Communication theory of secrecy systems. Bell System Technical Journal **28** (1949) 656–715
15. Sun, H.-M., Shieh, S.-P.: Construction of dynamic threshold schemes. Electronics Letters **30** (1994) 2023–2025
16. Zhang, Y., Hardjono, T., Seberry, J.: Reusing shares in secret sharing schemes. The Computer Journal **37** (1994) 199–205

This article was processed using the LaTeX macro package with LLNCS style