On-line Secret Sharing

Christian Cachin

Institute for Theoretical Computer Science
ETH Zürich
CH-8092 Zürich, Switzerland
E-mail: cachin@inf.ethz.ch

Abstract. We propose a new construction for computationally secure secret sharing schemes with general access structures where all shares are as short as the secret. Our scheme provides the capability to share multiple secrets and to dynamically add participants on-line, without having to re-distribute new shares secretly to the current participants. These capabilities are gained by storing additional authentic (but not secret) information at a publicly accessible location.

1 Introduction

Secret sharing is an important and widely studied tool in cryptography and distributed computation. Informally, a secret sharing scheme is a protocol in which a dealer distributes a secret among a set of participants such that only specific subsets of them, defined by the *access structure*, can recover the secret at a later time.

Secret sharing has largely been investigated in the *information-theoretic* security model, requiring that the participants' shares give no information on the secret, i.e. that the respective probability distributions are independent. Called *perfect* secret sharing schemes, they require that for every participant the number of bits needed to represent a share must be at least as large as the number of bits required to describe the secret itself (analogous to Shannon's theorem about key size for a perfectly secure cipher).

If the access structure allows any subset of k or more of the n participants to reconstruct the secret but not $k-1$ or less, the secret sharing scheme is called a *threshold scheme*. It can be implemented with Shamir's construction [14] based on polynomial interpolation. Secret sharing schemes for general monotone access structures are known, based on monotone circuit constructions [1, 9]. The surveys by Stinson [16] and Simmons [15] provide a general description of secret sharing schemes.

For many access structures it can be proved that some shares have to be considerably larger than the secret in perfect schemes [5]. Moreover, there exist families of special access structures on n participants where the size of some shares must grow unboundedly as $n \to \infty$ [6].

In the schemes described so far, the set of participants remains unchanged until the secret is recovered. Blakley *et al.* [2] study threshold schemes with *disenrollment* capabilities, where a participant is free to leave and to give away his share. The dealer then shares a new secret by broadcasting a message over a public channel. For perfect threshold schemes with m-fold disenrollment it can be shown that the size of the initially distributed shares must grow linearly in m [2]. These results are extended to general dynamic access structures by Blundo *et al.* [3]. Schemes for distributing multiple secrets are examined in [4].

Recently, Krawczyk [10] introduced a construction for *computationally* secure threshold secret sharing schemes where the shares can be shorter than the secret and that uses a secure encryption function. Basically, this protocol works as follows: The (potentionally large) secret is encrypted with a symmetric encryption function. The result is distributed among the participants using an information dispersal protocol [13] based on error correcting codes. Any k out of the n participants can reconstruct the encrypted secret. To prevent an unauthorized set of participants from learning anything about the secret, the secret key used for encryption is distributed among the participants using a conventional, unconditionally secure secret sharing scheme (e.g. Shamir's threshold scheme [14]).

Much research in the area of secret sharing has concentrated on the size of the shares. Although the size of the shares is important because the shares have to be transmitted and stored secretly, this is not the only information the participants must know to reconstruct the secret. Additional knowledge needed is, for example, the identity of the participants or the description of the protocol, including the access structure. These parameters are publicly known, but at the same time it is vital that they are authentic, i.e. no malicious participant has changed these descriptions. This is particularly important if the participants are computer systems that receive the descriptions over a potentially insecure communications link.

We propose a novel computationally secure secret sharing scheme for general access structures where all shares are as short as the secret. Our scheme provides the capability to share multiple secrets and to dynamically add participants on-line, without having to re-distribute new shares secretly to the current participants. These capabilities are traded for the need of storing additional authentic (but not secret) information at a publicly accessible location, e.g. on a bulletin board. Alternatively, this information can be broadcast to the participants over a public channel. The protocol gains its security from any one-way function. In particular, our construction has the following properties:

- All shares that must be transmitted and stored secretly once for every participant are as short as the secret.
- Multiple secrets can be shared with different access structures requiring only one share per participant for all secrets. This includes the ability for the dealer to change the secret after the shares have been distributed.
- The dealer can distribute the shares on-line: When a new participant is added and the access structure is changed, already distributed shares remain valid. Apart from the new participant's share that is secretly transmitted to him, only publicly readable information has to be changed.

The scheme is secure given any secure one-way function in the sense that a non-qualified set of participants running a polynomial-time algorithm cannot determine the secret with non-negligible probability. To prevent an attack by exhaustive search, however, the set of possible secrets must not be too small. Our construction solves an open problem of [10], albeit in a somewhat different way than proposed there.

Compared to traditional, unconditionally secure secret sharing schemes the proposed method is very flexible and uses only small shares. The differences lie in the additional use of publicly accessible information and in the security model. As for the use of authentic storage, we note that public information is needed in all traditional secret sharing schemes and that, authenticity usually costs much less than secrecy to implement.

Regarding the security model, computational security is theoretically weaker than information-theoretic or perfect security. On the other hand, for many applications that use a perfectly secure protocol, the cost of generating the needed random bits is prohibitively high and the bits are generated by a computationally secure pseudo random number generator. This makes the perfectly secure protocol vulnerable to adversaries with unlimited computing power.

The proposed scheme has many practical applications in situations where the participants and the access rules or the secret itself frequently change. No new shares have to be distributed secretly when new participants are included or participants leave. Such situations often arise in key management, escrowed [7] and fair [12] encryption systems, to name a few.

Consider, e.g., a high security area in a laboratory or in a bank where employees and managers are not permitted during off-hours. Only groups of one manager and at least two employees may enter and a secret sharing scheme is used to share the access code. If, for example, a manager is fired, he will disclose his share. With our scheme, only the access code and the bulletin board have to be updated—the other managers and employees do not have to be given new shares.

Another example is a group of frequently changing participants and alternating size where always two thirds of the current group members are needed to invoke some action, for example to reconstruct a master key used for escrowing keys of malicious users.

The paper is organized as follows: The basic scheme is presented in Section 3 and extended for sharing multiple secrets in Section 4. On-line secret sharing is then described in Section 5.

2 Preliminaries

We first need to formalize some aspects of a secret sharing scheme. A secret sharing scheme is a protocol between a set of participants $\mathcal{P} = \{P_1, \ldots, P_n\}$ and a dealer D, where $D \notin \mathcal{P}$ is assumed. The *access structure* $\Gamma \subseteq 2^{\mathcal{P}}$ is a family of subsets of $\{P_1, \ldots, P_n\}$ containing the sets of participants qualified to recover the secret. It is natural to require Γ to be monotone, that is, if $X \in \Gamma$ and $X \subseteq X' \subseteq \mathcal{P}$, then $X' \in \Gamma$. A *minimal* qualified subset $Y \in \Gamma$ is a set of participants such that $Y' \notin \Gamma$ for all $Y' \subset Y, Y' \neq Y$. The *basis* of Γ, denoted by Γ_0, is the family of all minimal qualified subsets. Note that Γ_0 uniquely determines Γ and vice versa.

For simplicity, we assume that the secret K is an element of a finite Abelian group $\mathbf{G} =< G, + >$ with $l = \log_2 |G|$. $\mathbf{G}$ could be the set of l-bit strings under bitwise addition modulo 2.

A *computationally secure secret sharing scheme* [10] is a protocol between D and the members of $\mathcal{P}$ to share a secret K, respective to an access structure Γ such that

a) the dealer D transmits a share S_i secretly to participant P_i, for $i = 1, \ldots, n$,
b) all qualified sets of participants $X \in \Gamma$ can efficiently compute K from their set of shares $\{S_i | P_i \in X\}$, and
c) every unqualified subset of participants $X \notin \Gamma$ running any polynomial-time algorithm cannot determine K with non-negligible probability.

To make the definition of security rigorous, we have to resort to asymptotics and consider the family of probability distributions of K indexed by the length l of the secret. Under this definition, for any unqualified subset $X \notin \Gamma$ running any algorithm A to recover K in time polynomial in l, the output of A must be equal to the correct K only with probability less than l^{-c}, for all constants c and suitably chosen $l > l_c$.

We will make use of a one-way function on G, $f : G \to G$ such that $f(x)$ is easy to compute for all $x \in G$ (i.e. can be computed in time polynomial in l) and that it is computationally infeasible, for a given $y \in G$, to find an $x \in G$ such that $f(x) = y$. The notion can be made rigorous analogous to the definition of security.

To achieve reasonable security, the security parameter l and thus the set of possible secrets have to be chosen large enough. Today, many secure one-way functions exist with typical l ranging from 64 to 128.

3 The Basic Scheme

Our protocol uses a publicly accessible location where the dealer can put up non-forgeable information that can be accessed by all the participants. We will refer to this location as the *bulletin board*. Alternatively, if communication and storage were not too expensive, the dealer could broadcast the information to the participants instead of storing it centrally. Implicitly, such a bulletin board is present in all existing secret sharing schemes and contains at least the number of participants n and the access structure Γ.

The basic protocol to share a secret $K \in G$ works as follows:

1. The dealer randomly chooses n Elements $S_1, \ldots, S_n$ from G according to the uniform distribution.
2. For all $i = 1, \ldots, n$, the dealer transmits S_i over a secret channel to P_i.

3. For each minimal qualified subset $X \in \mathit{\Gamma}_0$, the dealer computes

$$T_X = K - f(\sum_{x:P_x \in X} S_x)$$

and publishes $\mathcal{T} = \{T_X | X \in \mathit{\Gamma}_0\}$ on the bulletin board.

Addition and subtraction are performed in $\mathbf{G}$. To recover the secret K, a qualified set of participants Y proceeds similarly:

1. The members of Y agree on a minimal qualified subset $X \subseteq Y$.
2. The members of X add their shares together to get $V_X = \sum_{x:P_x \in X} S_x$ and apply the one-way function f to the result V_X.
3. They fetch T_X from the bulletin board and compute $K = T_X + f(V_X)$.

One can easily verify the completeness of the protocol: every qualified subset $X \in \mathit{\Gamma}$ can recover K.

Analyzing the security is only slightly more complicated: The relation between K and the shares is given by the $|\mathit{\Gamma}_0|$ equations

$$K = T_X + f(V_X)$$

for all $X \in \mathit{\Gamma}_0$, where the $V_X = \sum_{x:P_x \in X} S_x$ are all computed from different sets of shares. In the following, we denote by V_X the sum $\sum_{x:P_x \in X} S_x$ for any set of participants X. An unqualified subset $U \notin \mathit{\Gamma}$ cannot compute any of the $V_X, X \in \mathit{\Gamma}_0$ directly. So the members of U cannot compute K by exploiting one equation alone. However, they can link several equations through K or through any V_X. Linking two equations via K, one obtains relations of the form

$$T_Y - T_Z = f(V_Y) - f(V_Z)$$

with $Y, Z \in \mathit{\Gamma}_0$, of which the right sides are unknown to the members of U. Except for the unlikely case that $V_Y = V_Z$ which can be recognized on the bulletin board from $T_Y = T_Z$, this if of no use to them.

Linking two equations via V_W with $W \cap U = \emptyset$ and $W \cup U' \in \mathit{\Gamma}_0$, $W \cup U'' \in \mathit{\Gamma}_0$, for $U' \subset U$, $U'' \subset U$, and $U' \neq U''$ yields

$$f^{-1}(K - T_{W \cup U'}) - f^{-1}(K - T_{W \cup U''}) = V_{U'} - V_{U''},$$

thus nothing what the members of U could not have computed by themselves.

The size of $\mathcal{T}$ deserves some consideration. In general, $\mathcal{T}$ and the bulletin board are of size $O(2^n)$. However, note that for almost all general $\mathit{\Gamma}$ the description of $\mathit{\Gamma}$ itself is of the same size. This does not apply to threshold schemes that can be described by a list of participants plus two parameters (t, n) and for which $\mathcal{T}$ contains $\binom{n}{t}$ elements. But threshold schemes may be more important in theory than in practice: Reflecting on the way large companies and organizations are structured hierarchically today, it seems unlikely that a threshold scheme with more than several thousands of members will be realized by them.

In case the size of the bulletin board is limited, the authenticity of $\mathcal{T}$ can also be guaranteed by a digital signature of $\mathcal{T}$ by the dealer. If $\mathcal{T}$ is large, then parts of it could be signed such that not the entire table has to be read to validate a single entry.

Only one member of $\mathcal{T}$ has to be accessed to recover the secret. Therefore, the bulletin board could be implemented dynamically as a sever that broadcasts the desired entry upon request, together with a signature.

The shares of the participants in X are the inputs to a computation that ultimately yields K. For the basic scheme where one secret is shared once, the shares do not have to be kept secret during this computation. However, in the next two sections where additional capabilities of the scheme will be introduced, the shares and the result of their addition have to be kept secret. We will then assume that the participants can compute $f(V_X)$ without revealing their shares.

The protocol also allows the dealer to change the secret after the shares have been distributed by modifying $\mathcal{T}$ on the bulletin board.

4 Sharing Multiple Secrets

To share multiple secrets $K^1, K^2, \ldots$ with different access structures $\Gamma^1, \Gamma^2, \ldots$ among the same set of participants $\mathcal{P}$, the dealer distributes the shares S_i only once but prepares $\mathcal{T}^1, \mathcal{T}^2, \ldots$ for each secret. However, straightforward repetition of the basic scheme is not secure. Consider a set of participants X qualified to recover both K^1 and K^2: Any group $Y \in \Gamma^1$ can obtain K^2 as

$$K^2 = T_X^2 + T_Y^1 + f(V_Y) - T_X^1,$$

because $K^1 = T_X^1 + f(V_X) = T_Y^1 + f(V_Y)$ and because $f(V_X)$ is the same for K^1 and K^2. So $K^h - T_X^h$ must not be the same for different secrets.

To remedy this deficiency we replace f by a family $F = \{f_h\}$ of one-way functions so that different one-way functions are employed for different secrets. The following protocol is used to share m secrets K^h with access structures Γ^h for $h = 1, \ldots, m$:

1. The dealer randomly chooses n Elements $S_1, \ldots, S_n$ from G according to the uniform distribution.
2. For all $i = 1, \ldots, n$, the dealer transmits S_i over a secret channel to P_i.
3. For each secret K^h to share (with $h = 1, \ldots, m$) and for each minimal qualified subset $X \in \Gamma_0^h$, the dealer computes

$$T_X^h = K^h - f_h\Big(\sum_{x:P_x \in X} S_x\Big)$$

and publishes $\mathcal{T}^h = \{T_X^h | X \in \Gamma_0^h\}$ on the bulletin board.

To recover some secret K^h, a set of participants $Y \in \Gamma^h$ proceeds similarly:

1. The members of Y agree on a minimal qualified subset $X \subseteq Y$.
2. The members of X compute $V_X = \sum_{x:P_x \in X} S_x$ and apply f_h to V_X.
3. They fetch T_X^h from the bulletin board and compute $K^h = T_X^h + f_h(V_X)$.

The scheme does not demand a particular order for the reconstruction of the secrets. The required family F of one-functions can easily be obtained from f by setting $f_h(x) = f(h + x)$ when h is represented suitably in G.

Because a different one-way function f_h is used for each secret K^h, the information $\mathcal{T}^h$ on the bulletin board corresponding to K^h is (computationally) independent of the other $\mathcal{T}^{h'}$ and $K^{h'}$ for $h \neq h'$. Thus, the security of the protocol is the same as for the basic protocol.

As noted above, the shares have to be protected from the eyes of other participants during the reconstruction phase. Otherwise, these participants could subsequently recover other secrets they are not allowed to know. We assume therefore that the computation of $f_h(V_X)$ is performed without revealing the set of inputs $\{S_i | P_i \in X\}$. Possible ways of achieving this include the presence of a trusted device to perform the computation or the use of a distributed circuit evaluation protocol [8].

The protocol does not impose any limitation on m except for $|\mathcal{F}|$, the size of the family of hash functions, such that any number of secrets can be distributed via the bulletin board while the shares of the participants remain the same.

5 On-line Secret Sharing

In many situations, the participants of a secret sharing scheme do not remain the same during the entire life-time of the secret. The access structure itself may change, too, if it is adapted to the new constellation of participants. In analogy to the monotonicity of the access structure we will assume that the changes to the access structure are monotone, i.e. participants are only added and qualified subsets remain qualified.

We define a *computationally secure on-line secret sharing scheme* to be a protocol between a dealer D and the members of a sequence of sets of participants $\mathcal{P}(0), \mathcal{P}(1), \ldots$ with $\mathcal{P}(t) \subset \mathcal{P}(t+1)$ for all $t \geq 0$ to share a secret K, respective to a sequence of access structures $\Gamma(0), \Gamma(1), \ldots$ with $\Gamma(t) \subseteq \Gamma(t+1)$ for all $t \geq 0$, such that

a) the shares S_i for $P_i \in \mathcal{P}(0)$ form a computationally secure secret sharing scheme for K respective to $\Gamma(0)$,
b) at time $t > 0$ the dealer D transmits a share S_i secretly to every participant $P_i \in \mathcal{P}(t) \setminus \mathcal{P}(t-1)$,
c) for all $t \geq 0$, every qualified set of participants $X \in \Gamma(t)$ can efficiently compute K from their set of shares $\{S_i | P_i \in X\}$, and
d) for all $t \geq 0$, all unqualified subsets of participants $X \notin \Gamma(t)$ running any polynomial-time algorithm cannot determine K with non-negligible probability.

The basic scheme from Section 3 satisfies the above definition when the dealer operates step-by-step, distributing the shares to the new participants and updating the bulletin board accordingly for every step. In particular, at step $t > 0$, D chooses a random S_i for every $P_i \in \mathcal{P}(t) \setminus \mathcal{P}(t-1)$ and publishes the T_X with

$X \in \Gamma_0(t)$ and $X \notin \Gamma_0(t-1)$. The previously issued shares are not invalidated and no shares have to be retransmitted.

The protocol of Section 4 to share multiple secrets can be extended similarly for on-line sharing of multiple secrets.

6 Extensions

The flexibility and the simplicity of the protocols allow many extensions to handle additional situations. We briefly discuss removing participants as opposed to adding them in on-line schemes and the secrecy of the shares during reconstruction in multi-secret schemes.

If a participant P_i is removed or disenrolled at time t, he will publish his share S_i, eventually enabling an unqualified set $X \notin \Gamma(t')$ to recover K if $X \cup S_i \in \Gamma(t')$ for some $t' \geq t$ if such an X exists. But in contrast to traditional secret sharing schemes, if a new secret is chosen to be shared, the dealer needs only update the bulletin board and no information has to be transmitted secretly. The same situation arises if the sequence of access structures is allowed to be non-monotonic.

The proposed multi-secret sharing protocols are only secure if the members of a qualified set X do not disclose their shares when a secret K^h is reconstructed. Otherwise, some schemes for $K^{h'}, h' \neq h$ could be compromised. If the shares cannot be hidden to carry out this computation, the protocols can be modified as follows: Similar to Lamport's one-time user authentication scheme [11], S_i is replaced by $f^{(N-h)}(S_i)$ for all $i = 1, \ldots, n$ in the h-th scheme (N is a predefined constant and $f^{(h)}(x)$ denotes h-fold repeated application of f to x). The secrets $K^h, h = 1, \ldots, n$ have to be recovered in increasing order. Thus, after the reconstruction of K^h, only values $f^{(N-h')}(S_i)$ for $h' > h$ are needed to reconstruct additional secrets, but these values cannot be computed from $f^{(N-h)}(S_i)$ if the one-way function is secure. The drawback is, apart from the fixed order of reconstruction, that N has to be chosen in advance and poses an upper limit on m, the number of secrets that can be distributed.

References

1. J. Benaloh and J. Leichter, *Generalized secret sharing and monotone functions*, in Advances in Cryptology — CRYPTO '88, S. Goldwasser, ed., vol. 403 of Lecture Notes in Computer Science, Springer-Verlag, 1990, pp. 27–35.
2. B. Blakley, G. R. Blakley, A. H. Chan, and J. L. Massey, *Threshold schemes with disenrollment*, in Advances in Cryptology — CRYPTO '92, E. F. Brickell, ed., vol. 740 of Lecture Notes in Computer Science, Springer-Verlag, 1993, pp. 540–548.
3. C. Blundo, A. Cresti, A. De Santis, and U. Vaccaro, *Fully dynamic secret sharing schemes*, in Advances in Cryptology — CRYPTO '93, D. R. Stinson, ed., vol. 773 of Lecture Notes in Computer Science, Springer-Verlag, 1994, pp. 110–125.

4. C. Blundo, A. De Santis, G. Di Crescenzo, A. G. Gaggia, and U. Vaccaro, *Multi-secret sharing schemes*, in Advances in Cryptology — CRYPTO '94, Y. G. Desmedt, ed., vol. 839 of Lecture Notes in Computer Science, Springer-Verlag, 1994, pp. 150–163.
5. R. M. Capocelli, A. D. Santis, L. Gargano, and U. Vaccaro, *On the size of shares for secret sharing schemes*, in Advances in Cryptology — CRYPTO '91, J. Feigenbaum, ed., vol. 576 of Lecture Notes in Computer Science, Springer-Verlag, 1992, pp. 101–113.
6. L. Csirmaz, *The size of a share must be large*, in Advances in Cryptology — EUROCRYPT '94, A. De Santis, ed., vol. 950 of Lecture Notes in Computer Science, Springer-Verlag, 1995, pp. 13–22.
7. D. E. Denning and M. Smid, *Key escrowing today*, IEEE Communications Magazine, 32 (1994), pp. 58–68.
8. O. Goldreich, S. Micali, and A. Wigderson, *How to play any mental game or a completeness theorem for protocols with honest majority*, in Proc. 19th ACM Symposium on Theory of Computing (STOC), 1987, pp. 218–229.
9. M. Ito, A. Saito, and T. Nishizeki, *Secret sharing scheme realizing general access structure*, in Proceedings of IEEE Globecom '87, 1987, pp. 99–102.
10. H. Krawczyk, *Secret sharing made short*, in Advances in Cryptology — CRYPTO '93, D. R. Stinson, ed., vol. 773 of Lecture Notes in Computer Science, Springer-Verlag, 1994, pp. 136–146.
11. L. Lamport, *Password authentication with insecure communication*, Comm. ACM, 24 (1981).
12. S. Micali, *Fair public-key cryptosystems*, in Advances in Cryptology — CRYPTO '92, E. F. Brickell, ed., vol. 740 of Lecture Notes in Computer Science, Springer-Verlag, 1993, pp. 113–138.
13. M. O. Rabin, *Efficient dispersal of information for security, load balancing, and fault tolerance*, J. Assoc. Comput. Mach., 36 (1989), pp. 335–348.
14. A. Shamir, *How to share a secret*, Comm. ACM, 22 (1979), pp. 612–613.
15. G. J. Simmons, *An introduction to shared secret and/or shared control schemes and their application*, in Contemporary Cryptology: The Science of Information Integrity, G. J. Simmons, ed., IEEE Press, 1991, pp. 441–497.
16. D. R. Stinson, *An explication of secret sharing schemes*, Designs, Codes and Cryptography, 2 (1992), pp. 357–390.