19

An Hierarchical Threshold Scheme with Unique Partial Keys

Hannes Hassler
Vesna Hassler
Reinhard Posch
Graz University of Technology
Institute for Applied Information Processing and Communications,
Klosterwiesgasse 32/I, A-8010 Graz.
Telephone: ++43 316 8735514. Fax: ++43 316 8735520.
email: {hhassler,vhassler,rposch}@iaik.tu-graz.ac.at

Abstract

We will present an extension of Shamir's threshold scheme (Shamir 1979). Shamir's scheme demonstrates how to divide a master key D into n pieces so that it is easily reconstructed from any k pieces, where even complete knowledge of $k-1$ pieces reveals nothing about D. Shamir calls it a *(k,n) threshold scheme*. We propose a method that enables the creation of hierarchical information threshold schemes with s security levels, so that k_l partial keys (shares) are required for computation of a master key D_l for level l. The higher the security level, the more partial keys required. We call our scheme a (k_l, s, n) *multithreshold scheme*, $l = 1, \cdots, s$.

Keywords

cryptography, threshold schemes

1 INTRODUCTION

We will consider the following scenario: A group of ten experts is responsible for operating a protected system. In order to carry out some minor changes in the system, at least three of them must be present. If they want to change more sensitive elements, more group members must agree to this. As more of them become willing to participate, more sensitive system elements can be accessed. This problem can be solved using a "conventional" threshold scheme, such as e.g. Shamir's scheme (Shamir 1979), in such a way that for each security level a different master key, as well as a corresponding set of individuals' partial keys (shares), are predetermined. Thus each member would obtain as many shares as there are security levels. However, we wish to solve the problem so that each member of the group has to obtain *only one share.*

In a threshold scheme, the *master key* for access to the protected system elements

S. K. Katsikas et al. (eds.), *Information Systems Security*

may be obtained if and only if a certain number (*threshold*) of shares is submitted. The algorithm that transforms the shares into the master key is called a *master key constructor* (Erickson 1993). In our example there are several *security* or *information* levels. Depending on the number of shares submitted to the constructor, a different master key must be obtained. The problem may be solved by a *hierarchical information* threshold scheme, where each user is given only one share that is valid for all security levels.

Shamir considers the following example: A company signs all its checks digitally. Each executive is given a share to a $(3, n)$ threshold scheme. In other words, a check can be signed by any three out of n executives. However, from a financial point of view, there is a big difference between a 1000$ check and a 100,000$ one. Thus, more powerful signatures (i.e. shares) are required for greater amounts. We can meet this requirement using a *hierarchical authority* scheme with *weighted* shares.

With the multithreshold scheme presented in this paper both of the problems described above can be solved using a unique set of shares.

2 THRESHOLD SCHEMES

Shamir (Shamir 1979) and Blakley (Blakley 1979) each working independently were the first to propose threshold schemes. They showed how to divide a key D among n persons such that any subgroup of k persons could reconstruct it. D is usually referred to as the master key or the secret, and may be either the information that is being protected, or a cryptographic key that allows access to that information. Shamir's scheme is based on polynomial interpolation and Blakley's scheme on linear projective geometry.

In addition to these two schemes, there are many other authors that proposed (k, n) threshold schemes. Among these are schemes based on the intersection properties of finite geometries (Beutelspacher and Vedder 1987, Beutelspacher 1988), Steiner systems (Stinston and Vanstone 1988), error correcting codes (Davida et al. 1980), Chinese Remainder Theorem (Asmuth and Blakley 1982, Asmuth and Bloom 1983) or matrix multiplication (Karnin et al. 1983). Simmons (Simmons 1989, 1990, 1991) deals with generalized secret-sharing schemes and the additional capabilities they should possess for "real world" applications.

2.1 Hierarchical Information

In a hierarchical threshold scheme the information may be divided up into several levels which may be seen as information or security levels. The higher levels may be accessed by submitting more shares (partial keys, see Fig.1). Erickson (Erickson 1993) explores the variations and expansions of the existing methods for threshold schemes to accommodate hierarchical information. Erickson considers hierarchical authority schemes as well, i.e. the possibility of issuing more powerful (weighted) shares to some of the participants.

The multilevel concurrency scheme presented by Simmons (Simmons 1989) is a compartmented threshold scheme based on finite geometries that can be adapted for hierarchical information (Erickson 1993). However, as in each compartmented scheme, the threshold for each compartment must be achieved in order to access higher security levels. The construction of a scheme for two security levels is simple, but for more than two levels the design complexity increases significantly. In our scheme there are no compart-

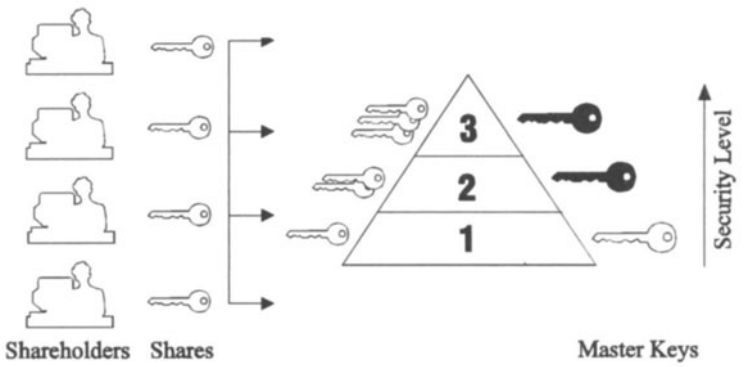

Figure 1 *Hierarchical threshold scheme*

ment thresholds. The number of shadows needed to access any of the security levels can be designed arbitrarily, even after the shares have been distributed. The increase in the number of levels does not influence the design complexity of the scheme.

The hierarchical threshold scheme based on the schemes presented in Beutelspacher 1990 and Beutelspacher and Vedder 1987 provides hierarchical authority as well. Lower authority shareholders may access a higher security level if and only if a certain number of higher authority shares are presented to the master key constructor. Our first design requirement is that all shares have equal authority, although the scheme can be extended to support hierarchical authority.

The hierarchical threshold scheme can also be based on Steiner systems (Erickson 1993). However, the problem is that only a few such systems are known. In other words, insufficient schemes of this type exist, making the method impractical.

In Harari 1983 a hierarchical threshold scheme based on error correcting codes is proposed which also includes hierarchical authority. A restriction of the scheme is that there must be a certain bit distance between two adjacent security levels in order for the higher one to be protected. Shares for higher authorities and higher security levels are therefore of increasing length. We do not impose such a restriction, i.e. all shares have the same length.

An extension of the system proposed by Mignotte (Mignotte 1983) is a hierarchical threshold scheme based on the Chinese Remainder Theorem (Erickson 1993). However, in this scheme a different share for each security level must exist. In an alternate extension the storage space required for the master key increases greatly with higher levels. In our method the same share is used to access any of the security levels. The storage space requirements remain the same for all security levels.

3 THE MULTITHRESHOLD SCHEME

We propose a hierarchical information threshold scheme which is based on polynomial interpolation and the discrete logarithm problem. In our scheme the same (unique) share may be used to access any of the security levels. The level that may be accessed will depend only on the number of submitted shares. The storage requirements for master keys remain the same for all security levels and for any number of levels. Additional storage is needed for the scheme parameters which are public. The scheme is very simple to set up.

Hierarchical authority can also be implemented if a shareholder obtains an "extended" share which in fact represents a set of "simple" shares. The extension to the share is half the size of the simple share, no matter what the weight of the extended share is. In other words, if a simple share is of length $2n$, an extended share is of length $3n$ (the secret is of lengtn n). Protection against cheating may be achieved in the same way as in Shamir's scheme (Tompa and Woll 1986).

Now we will present a method which enables us to distribute shares among n participants (one share per participant) such that any subgroup of k_l participants can reconstruct the corresponding master key D_l, $l = 1, \cdots, s$. This means, the number of submitted shares determines which security level may be accessed. We call our scheme a (k_l, n, s) *multithreshold scheme*, $l = 1, \cdots, s$.

As with Shamir's scheme, we calculate modulo p, p being a large prime. There is also a polynomial of degree m which has to be interpolated. For simplicity, let $k_l = l$, $l = 1, \cdots, s$ (the number of presented shares) denote the security level. For instance, if there are three shares available ($k_l = 3$), the security level $l = 3$ will be accessible. With no shares, there are m unknowns. The possession of one share reduces the number of unknowns by one. To access the master key for a certain security level some public constants have to be computed when setting up the scheme. The lower security level, the more public constants are available and hence fewer shares required to access it.

3.1 Mathematical Background

Our scheme is based on polynomial interpolation: Given m distinct points in the 2-dimensional plane $(x_1, y_1) \cdots (x_m, y_m)$ there is one and only one polynomial $f(x)$ of degree $m-1$ such that $f(x_i) = y_i$ for all i. As with Shamir in Shamir 1979 we use modular arithmetic. The set of integers modulo a prime number q forms a finite field in which discrete interpolation is possible. The coefficients of the polynomial are kept secret. q is chosen to be a prime divisor of $p-1$, where p is another large prime. Both p and q are public values. Our scenario is similar to that of The digital signature standard 1992.

Additionally, for s security levels we need s integers $\lambda_k > 1$, such that $\lambda_k = h_k^{\frac{(p-1)}{q}} \bmod p$, where $1 \leq h_k \leq p-1$ is a random integer. Each $\lambda_k, k = 1, \cdots, s$, is a generator of order q in GF(p), i.e. $\lambda_k^r \bmod p = \lambda_k^{r \bmod q} \bmod p$.

3.2 Set Up. Key Generation

To generate an adequate polynomial, we choose m pairs of integers (x_i, y_i) at random, $x_i, y_i < q$. Each participant receives one such pair (x_i, y_i), i.e. his/her share.

The set of integers modulo a prime q forms a finite field which enables interpolation. We interpolate a polynomial $f(x)$ of degree $m-1$, $f(x) = a_0 + a_1 x + \cdots + a_{m-1}x^{m-1} \bmod q$, such that $y_i = q(x_i) \bmod q$ for each i.

Let the master key of security level k be $\lambda_k^{a_0} \bmod p$. The master key constructor provides $m-k$ public constants $c_{k,1} = \lambda_k^{-a_1}, \cdots, c_{k,m-k} = \lambda_k^{-a_{m-k}}$ for the kth level. For s security levels there are $m = \sum_{k=1}^{s}(m-k) = ms - s(s+1)/2$ public constants.

The security of the scheme relies on the fact that the discrete logarithm cannot be computed within reasonable time. Hence, even with the knowledge of $c_{k,i}$ and λ_k one cannot compute a_i. With current technology it is beyond reach to deduce x from $\lambda^x \bmod q$ when q is in the range of at least 500 bits.

3.3 Master Key Recovery

If we assume here that k participants cooperate providing k shares, i.e. pairs (x_i, y_i) to access the kth security level. The following computations are performed by the trusted key distribution center (KDC). KDC establishes a set of k equations with m unknowns, the a_i's. We refer to it as a matrix $\vec{y} = X\vec{a}$, where $\vec{y} = \{y_i\}$ and $X = \{x_{ij}\}$, $x_{ij} = x_i^j$, (x_i, y_i) being available shares, and $\vec{a} = \{a_j\}$ the unknown coefficients of the polynomial $f(x)$, $j = 0, \cdots, m-1, i = 0, \cdots, k-1$.

KDC takes the first column and the last $k-1$ columns of X to form the reduced matrix $X_r = \{x_i^j\}$, $i = 0, \cdots, k-1, j = 0, m-k+1, m-k+2, \cdots, m-1$. It computes only the first row of the inverse of $X_r \bmod q$. KDC uses this inverse (i.e. its first row) to form the modified system of equations, $X_r^{-1}\vec{y} = X_r^{-1}X\vec{a}$.

Now KDC needs only the first equation of the modified system. It is in the form of $v = a_0 v_0 + \cdots + a_{m-k}v_{m-k}, v_0 = 1$. All higher coefficients $a_i, i > m-k$, disappear when multiplied by X_r^{-1}. KDC brings a_i's, $1 \le i \le m-k$, to one side and, using the public constants available for this security level, computes $\lambda_k^v c_{k,1}^{v_1} \cdots c_{k,m-k}^{v_{m-k}} \bmod p = \lambda_k^{a_0}$. $\lambda_k^{a_0}$ is the master key for level k.

3.4 Hierarchical Authority

If we want to issue an extended share with weight w_i to a participant with higher authority, we simply choose the pairs $(x_i, y_i), (x_i + K, y_i + K), \cdots, (x_i + K(w_i - 1), y_i + K(w_i - 1))$, where $K = \frac{(p-1)}{q}$. The extended share of this participant is then (x_i, y_i, w_i). Note that in this case there are $n \le m$ issued shares, so that $\sum_{i=1}^{n} w_i = m$ holds. A simple share has weight $w_i = 1$. Weighted shares can only be determined prior to polynomial interpolation (set up).

3.5 Key Management

The trusted key distribution center (KDC) is responsible for selecting all parameters. After issuing the shares, KDC may "blow itself up", i.e. delete all parameters of the scheme that must be kept secret.

It is possible to distribute less shares among participants than necessary for accessing the highest possible security level s that may be accessed if and only there are enough

cooperating shares to interpolate the polynomial. Thus the coefficients of the polynomial can never be revealed either to honest participants or to tamperers.

If a share is revoked, the security manager may choose a new set of m shares which (set) contains still valid shares and does not contain the revoked one. It is therefore not necessary to change the shares of unaffected users. This set of shares is then used to compute a new polynomial. All public constants have to be recomputed as well.

Such a scheme can be used for system administration tasks. In some operating systems, there is only one root password that gives the system administrator the right to read/write/delete all files in the system. If there are several "root passwords" that are mapped to shares for our multithreshold scheme, then the submission of only one of the shares does not allow e.g. the deletion of other users' directories and files.

Furthermore, it is possible to skip some security levels. Only such security levels for which public constants are available can be accessed. In other words, the shares are useless until some pretetermined condition (preposition) is satisfied. This type of threshold schemes is called *prepositioned* (Simmons 1992). Under "normal circumstances" the participants are not able to compute the master key. If e.g. a missile battery goes to an advanced state of alert, depending on the "alert level" a certain number of public constants can be communicated thus enabling the battery to access this alert level's master key.

4 SUMMARY

In this paper an extension of Shamir's (k, n) threshold scheme (Shamir 1979) is described. We can establish a hierarchy of information as defined by Erickson (Erickson 1993) in a straightforward way. The mathematical background of the scheme is polynomial interpolation and the discrete logarithm problem. What we need in order to establish a scheme is a polynomial in a field of a large prime number p with secret coefficients, and a generator of order q in GF(p), so that $q \mid p - 1$. Some public constants also have to be precomputed and stored in memory. Each participant obtains only one share which then can be used for computing of any security level's master key. It is also possible to issue weighted shares (hierarchical authority) with a 50% increase in the share size. A useful property of the scheme is that a share may be revoked without affecting other shares or threatening security.

REFERENCES

Asmuth, C.A., Blakley, G.R. (1982) Pooling, splitting and reconstructing information to overcome total failure on some channels of communication. *Proc. IEEE Computer Society 1982 Symp. on Security and Privacy*, 156–169.

Asmuth, C., Bloom, J. (1983) A Modular Approach to Key Safeguarding. *IEEE Transactions on Information Theory*, **IT-30**, 208–10.

Beutelspacher, A. (1988) Enciphered geometry: Some applications of geometry to cryptography. *Annals of Discrete Mathematics*, **37**, 59–68, North Holland

Beutelspacher, A. (1990) How to say 'no'. *Proceedings of Eurocrypt'89*, 491–6, Springer Verlag, Berlin

Beutelspacher, A., Vedder, K. (1987) Geometric Structures as threshold schemes. *Proceedings of the 1987 IMA Conf. on Cryptography and Coding Theory*, Cirencester (England), Oxford University Press

Blakley, G.R. (1979) Safeguarding Cryptographic Keys. *Proceeding of AFIPS 1979 Nat. Comp. Conf.*, **48**, 313–7

Davida, G.I., DeMillo, R.A., Lipton, J.R. (1980) Protecting shared cryptographic keys. *Proc. IEEE Computer Society 1980 Symp. on Security and Privacy*, 100–2, Oakland (California, USA)

Erickson, D.L. (1993) Threshold Schemes with Hierarchical Information. Technical Report **9323**, University of Waterloo, Ontario (Canada)

Harari, S. (1983) *Secret Sharing Systems*, Springer Verlag, 105–10, Vienna (Austria)

Karnin, E.D., Greene, J.W., Hellman, M.E. (1983) On Sharing Secret Systems. *IEEE Transactions on Information Theory*, **IT-29**, 35–41

Mignotte, M. (1983) How to share a secret. *Cryptography*, 371–5

Shamir, A. (1979) How to Share a Secret *Comm. ACM*, **22**, 612–3

Simmons, G.J. (1989) How to (Really) Share a Secret. *Proceedings of Crypto'88*, 390–448

Simmons, G.J. (1990) Prepositioned Shared Secret and/or Shared Control Schemes. *Proceedings of Eurocrypt'89*, 436–67

Simmons, G.J. (1991) Geometric Shared Secret and/or Shared Control Schemes. *Proceedings of Crypto'90*, 216–41

Simmons, G.J., ed. (1992) *Contemporary Cryptology*, IEEE Press

Stinston, D.R., Vanstone, S.A. (1988) A combinatorial approach to threshold schemes. *Siam J.Disc.Math.*, **1**, 230–6

The digital signature standard (1992) *Comm. ACM*, **35**, 36–40

Tompa, M., Woll, H. (1986) How to share a secret with cheaters. *Proceedings of Crypto '86*, 261–5, Springer Verlag, Berlin

BIOGRAPHY

Hannes HASSLER received a M.Sc. and Ph.D. in Computer Science from the University of Technology in Graz (TU-Graz), Austria, in 1989 and 1994, respectively. From 1990-1994 he was assistant researcher at the Institute for Applied Information Processing and Communications (IAIK) of TU-Graz. From April 1994 - April 1995 he was a guest researcher at Kyoto University, Japan. Since April 1995 he has been a guest researcher at Nagoya University, Japan. His current research interests include computer arithmetic, VLSI design and cryptography.

Vesna HASSLER (née Ristić) received a B.Sc. and M.Sc. in Electrical Engineering from Zagreb University, Croatia, in 1988 and 1991, respectively. From 1989-1992 she was assistant researcher at the Faculty of Electrical Engineering in Zagreb. Since 1992 she has been assistant researcher at IAIK of TU-Graz. In December 1995 she received a Ph.D. in Computer Engineering and Communications from TU-Graz. Her research interests include network security, Web security, cryptography and secure applications.

Prof. Reinhard POSCH received a M.Sc. in Mathematics in 1973 and a Ph.D. in Informatics and Telecommunication in 1977 from TU-Graz. In 1984 he was approved as

Lecturer ("Habilitation") for Applied Information Processing and Information Technology. Since 1984 he has been a full professor. He has been head of IAIK since 1986. He is Austrian representative in IFIP TC6 (Communication) and IFIP TC11 (Computer Security). He has been vice chairman of TC11 since 1995. At present he is also coordinator of the Austrian EUROCHIP (ESPRIT II) project and coordinator of the OCG IT-Security group. His current research activities include networking, security, smart cards and VLSI design.