

Design a Secure and Practical Metering Scheme

Wen Chung Kuo

Department of Computer Science and
Information Engineering
National Formosa University
Taiwan, Republic of China
E-mail: simonkuo@nfu.edu.tw

Chun-Ju Fu

Department of Electronic Engineering
Institute of Computer & Communication
National Cheng Kung University
Taiwan, Republic of China

Chi-Sung Laih

Department of Electronic Engineering
Institute of Computer & Communication
National Cheng Kung University
Taiwan, Republic of China
E-mail: laihs@eembox.ncku.edu.tw

Abstract—More and more metering schemes based on some special protocols or mechanisms are used to calculate the number of visits which a website receives in a certain time frame. Then, the amount of money that an advertiser pay to a website depends on this number of visits. Until now, many protocols were suggested for metering schemes in literature. However, the discussion mainly focused on the theory of the metering scheme. How to design or realize the metering scheme is unclear. In this paper, a general transforming method which can transform a secret sharing scheme into a metering scheme is proposed. Then we follow this methodology to design a practical metering scheme based on a modular approach.

Keywords: Metering scheme, Secret sharing scheme, Authentication.

I. INTRODUCTION

Due to the rapid development of science technology and network transfer rate, more and more digital data, such as documents, pop music, e-books, *etc.*, are transmitted through Internet. So, Internet and website environment provide marketers with the new tools and convenience that can increase the success of their marketing efforts. However, advertisers need to know what their advertising result is. Therefore, a mechanism or protocols needed to measure the popularity of the websites and the selecting situations of advertisement. In order to reach this goal, metering schemes were proposed. Until now, more and more metering schemes were used to calculate the interactions between websites and clients.

On Internet, the metering traffic of websites is usually performed by automatic programs, which are installed at web servers to collect the access information from clients. So, they can also be used to collect the traffic information according to which the advertising fee is decided. For example, the amount of money paid from an advertiser to websites can be depending on the number of visits. From the view of economy consideration, we believe that website may practice fraud. When websites control the collection processes or the stored date, there may be cause many serious security worries. It shows that metering schemes should be secure enough to prevent this kind of fraud behaviors.

So far, there was much research focusing on metering the website's traffic. In 1997, Pitkow [14] discussed how to uniquely identify users and pay for the usage of proxies and cache providing access for many clients (or many visits) who registered at the servers for a single visit. Novak and

Hoffman [11] argued that it was critical to standardize the web measurement processes. Then they provided an overview of current practice and considerations that affect the question of what stands for network advertisement. But both of these two discussions were not concerned with the security issues.

In 1998, Franklin and Malkhi [6] were the first to use the rigorous technical approaches to discuss the metering schemes. Their scheme required neither a third party involved nor the record of any party that could be used to trace the identity of clients. Actually, it offered only a "lightweight security" and there is a great possibility of inflating the metering results.

Later, a secure metering scheme to prevent web servers from inflating the numbers of their visits was proposed by Naor and Pinkas [10]. Their scheme is based on a modified version of polynomial secret sharing scheme introduced by Shamir[15]. There are three main parties included in their scheme, i.e. m servers, n clients and an audit agency A . Here, the audit agency A is similar to a trust third party and it is responsible for appraising the popularity of the servers. They applied a modified version of secret sharing schemes for the audit agency to detect fraudulent intentions by some corrupt servers. At the same time, it can protect the metering processes from disrupting by ill-disposed clients. Their schemes are designed to work for τ time frames and it is supposed to be secure within such time frames. A server is able to figure out its proof for a certain time frame t if it has received a number of clients larger or equal to a fixed threshold h . Then the audit agency will make a certain payment to the server according to its proof. In other words, the audit agency cannot pay anything to the server if its audience does not achieve the default threshold. Since their pioneered studies, many metering schemes such as [2], [3], [4], [5], [8], [9], [12], [13] have been proposed in literature.

After our analysis, we found that there are some related features existing between the proposed metering schemes and the secret sharing schemes. Hence, there maybe exists a methodology which can transform secret sharing schemes into metering ones. Here, a generalized transforming method is proposed in this paper. Then we combine it with another threshold scheme constructed by Chinese Remainder Theorem (CRT) to generate a practical metering scheme which can overcome the drawbacks in [2], [3], [4], [5], [8], [9].

The rest of this paper is organized as follows. In next

section, we state the relationship between the secret sharing scheme and the metering scheme. Then, we review the Asmuth-Bloom threshold scheme in detail and provide an example to explain this scheme. A practical metering-scheme algorithm is proposed in Section 3 and then we analyze the security of our scheme in Section 4. Finally, Section 5 presents our conclusions.

II. RELATIONSHIPS BETWEEN SECRET SHARING SCHEMES AND METERING SCHEMES

A secret of the one group, such as a key K , can be split into several shares distributed to everyone in this group. To recover the secret K , only some of the shares are required. Such a problem is called secret sharing or secret splitting problem. In 1979, the most common secret sharing scheme was proposed by Shamir[15]. The secret K (or master key) is broken into n different "shadows", V_1, V_2, \dots, V_n , each of which is chosen from a set V such that it satisfies the following conditions in his scheme.

- 1) Knowing h or more than h different V_i can recover K .
- 2) Knowing only $h - 1$ or less different V_i cannot recover K .

Any such system is referred to as a (h, n) -threshold scheme or "h-out-of-n" secret sharing scheme.

Secret sharing schemes are originally used for key management issues, i.e., they are used to protect the master key from being leaked to attackers. However, we found that secret sharing schemes might be applicable to the metering scheme by using some transformation rules. When we deployed the h-out-of-n secret sharing scheme, there were a trusted third party and n participants. Let the trusted third party as a dealer whose job is to break a secret K into n shadows and distribute them to n participants, C_1, C_2, \dots, C_n . Then, people who collect any h shadows are able to reconstruct the secret K . The dealer is also responsible to verify the reconstructed secrets. Now, let $F(\star)$ be a share secret function and $F^{-1}(\star)$ be the function used to generate the secrets, where " \star " means the input parameter of these functions. When sharing a secret, " \star " indicates the secret K . So, $F(\star) = F(K)$, i.e., as a secret K is broken into several shadows V_1, V_2, \dots, V_n , each participant C_i gets a fixed shadow V_i for $i = 1, 2, \dots, n$. Relatively, when recovering the secret K , " \star " indicates the shadows, i.e., $F^{-1}(V_1, V_2, \dots, V_h) = K$.

Analogously, in the metering schemes, the audit agency is similar to the dealer in the secret sharing schemes. The audit agency A manages to split a major secret ϑ into n shadows and distribute these shadows to n clients. If a server has the number of visits equal to the threshold value h and verifies all the shadows are correct, he will certainly reconstruct the secret. Therefore, the server can prove its visits without handing over all information of clients to the audit agency. However, the same secret cannot reuse in different time frame t in the metering scheme. Once a sever has gotten the correct *proof*, it might forge the number of visits. In other words, we expected that our metering scheme is able to make a sever figure out different secret for different time frames. The audit agency

A does not need to communicate with clients to renew the shadows. All shadows held by clients will vary automatically according to the time frame t and specific servers. We find that the secrets in the metering process must be related to the parameter of t and the identity of sever (for the sever S_j , its identity is j).

Here, we denote the mathematical method used to share secrets in the metering scheme as $H(\star)$ and to generate the secret in the metering scheme as $H^{-1}(\star)$, where " \star " means the input parameters of these methods. When we are going to share a secret ϑ , the function $H(\star) = H(\vartheta)$ is used to distribute the shadows of the secret ϑ to C_1, C_2, \dots, C_n . Relatively, when we are going to regenerate the secret ϑ , " \star " indicates the collected shadows. So, $\vartheta = H^{-1}(C_1, C_2, \dots, C_h)$. Here, the shadows held by clients are concerned with a specific parameter z , where z denotes the operation between t and j (i.e. $z = t \circ j$).

A. Review the Asmuth-Bloom threshold scheme

Now, we conduct a study on the (h, n) -threshold scheme proposed by Asmuth and Bloom[1], and then try to extend it to a metering scheme. There are two phases in their threshold scheme, i.e., environment formation phase and secret recovery phase. Here, we roughly review this metering scheme as follows. In-depth treatments of the notations can be found in [1].

Part 1: Environment formation phase

In [1], they first let the shadows be congruence class of a number associated with the secret K . The audit agency A chooses a set of integers p, d_1, d_2, \dots, d_n , which must satisfy the following conditions.

- 1) $d_1 < d_2 < \dots < d_n$;
- 2) $\gcd(d_i, d_j) = 1$, for $i \neq j$;
- 3) $\gcd(p, d_i) = 1$, for all i ;
- 4) $\prod_{i=1}^h d_i > p \prod_{i=1}^{h-1} d_{n-i+1}$;
- 5) $p > K$.

Where $\gcd(d_i, d_j)$ denotes the great common divisor of two integers d_i and d_j . In Condition (1), we choose n incremental integers, which are labeled as d_1 to d_n and are relatively prime in Condition (2). Then, we pick a random prime number p to satisfy the Conditions (3 - 5).

Next, let $M = \prod_{i=1}^h d_i$ and r be a random integer within the range $[0, \lfloor \frac{M}{p} - 1 \rfloor]$. The audit agency A computes the parameter $K' = K + rp$ and $0 < K' < \prod_{i=1}^h d_i$.

Finally, in order to decompose K into n shadows, the audit agency A computes $K' = K_i \pmod{d_i}$ and distributes the pair (K_i, d_i) as shadows to each client. From Condition (4), any h different clients can generate the parameter K' . People who get the parameters K' and p are able to recover the secret K . On the other hand, if only $h - 1$ shadows were known, essentially no information about the secret K can be recovered.

Part 2: Secret recovery phase

If h pairs of different shadows $(K_{i1}, d_{i1}), (K_{i2}, d_{i2}), \dots, (K_{ih}, d_{ih})$ are known, the parameter K' can be figured out

by performing the CRT. Let $M_1 = d_{i_1} \cdots d_{i_h}$. We denote the abbreviation of CRT computations as following $K' = CRT(M_1, d_{i_1}, d_{i_2}, \dots, d_{i_h}, K_{i_1}, K_{i_2}, \dots, K_{i_h}) \bmod M_1$. Due to $M_1 \geq M$, we can uniquely determine K' and compute the secret K by K' modulo p [7].

B. An example of Asmuth-Bloom threshold scheme

Now, we give an example to describe the Asmuth-Bloom threshold scheme.

Example 1: We assume that the secret K is 3, the threshold value $h = 3$ and the total number of clients $n = 5$, the prime number $p = 5$ ($\gcd(K, p) = \gcd(3, 5) = 1$) and five numbers $d_1 = 11$, $d_2 = 13$, $d_3 = 17$, $d_4 = 19$, and $d_5 = 23$.

Part 1: Environment formation phase

- 1) Check whether these selected parameters satisfy the conditions (1-5) or not.
Calculate $M = d_1 \times d_2 \times d_3 (= 2431) > p \times d_4 \times d_5 (= 2185)$.
- 2) Choose a random number r between 0 and $\lfloor \frac{2431}{5} - 1 \rfloor$. Here, we pick $r = 481$.
- 3) Distribute the shadows. Let $K' = K + rp = 3 + 481 \times 5 = 2408$ and then

$$\begin{aligned} K_1 &= K' \pmod{d_1} = 10, \\ K_2 &= K' \pmod{d_2} = 3, \\ K_3 &= K' \pmod{d_3} = 11, \\ K_4 &= K' \pmod{d_4} = 14, \\ K_5 &= K' \pmod{d_5} = 16. \end{aligned}$$

Part 2: Secret recovery phase

If we get three of the shadows, we can compute K . By choosing K_1, K_3 and K_4 , we have

$$M_1 = d_1 \times d_3 \times d_4 = 11 \times 17 \times 19 = 3553. \quad (1)$$

Then applying CRT,

$$\begin{aligned} K' &= CRT(M_1, d_1, d_3, d_4, K_1, K_3, K_4) \\ &= CRT(3553, 11, 17, 19, 10, 11, 14) \\ &= 2408 \pmod{3553}. \end{aligned}$$

Consequently,

$$K' = K \pmod{p} = 3 \pmod{5}. \quad (2)$$

III. A PRACTICAL METERING-SCHEME ALGORITHM

According to [10], there are many clients C_1, C_2, \dots, C_n , many servers S_1, S_2, \dots, S_m , and an audit agency A in their scheme. The audit agency A is a special party that is responsible for dealing with measuring the interaction. It is not necessary for clients and servers to trust each other, but they all trust the audit agency A for the purpose of metering.

In particular, the goal of the metering system is to measure the number of visits that a server receives. Here, the visit can be defined according to the information such as a page hit or any other relevant information. Therefore, these operations of the proposed metering scheme have the following general structures.

Initialization phase:

First, the audit agency A chooses a random secret ϑ . Then the audit agency A uses ϑ as a parameter to produce initial messages for each client and server. The initial message of each client C_i is considered as its shadow. The audit agency A sends such messages to all receivers through a secure channel.

Regular operation phase:

The client C_i uses its own shadow and the initial message of S_j to compute a response and then gives it to S_j . Similarly, the server S_j will receive the initial message of S_j and a response from the client C_i when C_i visits S_j . The server S_j will keep the shadow if it is workable.

End of time frame:

The server S_j sends a request to audit agency A to demand the proof generation. When S_j makes the request, audit agency A will give another challenge to him. If the server S_j has received a certain number of visits during a certain time frame, he can response the challenge by calculating all the received shadows and the initial message.

A. A Practical Metering-Scheme Model

As explained before, we extend the above threshold scheme[1] to meet the requirement of our proposed scheme. Because the metering scheme must work over several time frames and several servers, the secret K cannot be revealed. However, the traditional threshold scheme can be only performed once. Here, we reconstruct the Asmuth-Bloom threshold scheme to satisfy the requirement of metering scheme. The initial message of each client is related to every time frame t . The audit agency selects a random interval of number sequence for each client. For the client C_i , the audit agency A selects a specific interval between d_{iva} and d_{ivb} . In the time frame t , the client chooses a random prime d_{it} in the interval. Then, he applies the random number r into the polynomial function $f(z)$, where $G(z) = K + f(z)p$ and z denotes the operation between t and j (i.e. $j \circ t$). Each value of $f(z)$ is restricted to be between 0 and $\lfloor \frac{M}{p} - 1 \rfloor$. As the above definitions, we choose a set of integers based on the following parameter selection rules.

Parameter selection rules

Rule 1a set of intervals $(d_{1va}, d_{1vb}), (d_{2va}, d_{2vb}), \dots, (d_{nva}, d_{nvb})$, where $d_{1va} < d_{2va} < \dots < d_{nva}$ and $d_{1vb} < d_{2vb} < \dots < d_{nvb}$.

Rule 2 $p > K$.

Rule 3 $G(z) = K + f(z)p$, where $f(z)$ is a polynomial and its value is within the range $[0, \lfloor \frac{M}{p} - 1 \rfloor]$.

These parameter selection rules are applied to our proposed metering scheme scenario as follows:

(i) At the initialization phase:

According to the parameter selection rule 1 and 2, the audit agency A chooses a secret K , a prime number p and a set of intervals $(d_{1va}, d_{1vb}), (d_{2va}, d_{2vb}), \dots, (d_{nva}, d_{nvb})$ where $(d_{iva} < d_{ivb})$.

The audit agency A randomly selects a polynomial $f(z)$ (which is a polynomial and its value is within the range $[0, \lfloor \frac{M}{p} - 1 \rfloor]$) and then computes $G(z) = K + f(z)p$.

Finally, he gives the information $(G(z), (d_{iva}, d_{ivb}))$ as a shadow to each client C_i through a private channel and generates the identity message j to each server S_j .

(ii) During the regular operation phase:

When a client C_i visits a server S_j in a certain time frame t , the client C_i randomly selects a number d_{it} ($d_{iva} < d_{it} < d_{ivb}$) and computes $G(j \circ t) = K_{i,j}^t \pmod{d_{it}}$. Then the client C_i gives the information $\{K_{i,j}^t, d_{it}\}$ to S_j .

(iii) End of the time frame t phase:

If the number of visits are over the threshold value, the server will combine the CRT method and these shadows to generate its *proof*. If the number of visits do not reach the threshold, the server S_j can ask for some shadows from the audit agency A until he is able to generate his proof. Finally, the audit agency A will pay something to the server S_j according to his proof and the number of shadows, which the server asked.

Now, we give another example to explain why our proposed scheme is feasible.

Example 2: The audit agency A chooses the parameters K , h , n , p , d_1 , d_2 , d_3 , d_4 and d_5 which are the same as those in example 1. Now, we assume that the server S_j has its own identity message $j = 2$ and the audit agency A picks a random function $f(z) = 4z^2 + 3z + 3$. Then he computes $G(z) = K + f(z)p = 3 + 5(4z^2 + 3z + 3) = 20z^2 + 15z + 18$ and gives it to each client C_i . We also assume that any client is able to use $G(z)$ to calculate its shadow, but he is not able to denote the function $G(z)$ to anybody.

In regular operation, when the client C_1 visits the server S_2 in the time frame $t = 3$, it receives the identity $j = 2$ from S_2 . The client C_1 computes $G(2 \times 3) = 828 \pmod{11} = 3 \pmod{11}$ and gives the pair $(3, 11)$ to the server S_2 . Then the clients C_3 and C_4 visit the server S_2 in the same time frame. They give their shadows $(12, 17)$ and $(11, 19)$ to S_2 .

At the end of the time frame $t = 3$, the server has received three visits in the time frame, and then he performs CRT(3553, 11, 17, 19, 3, 12, 11) to find the *proof*. The server S_2 gives the value of *proof* to the audit agency A . After the audit agency gets the *proof*, it verifies that the secret $K = 3$ is equal to the *proof*. If the *proof* is correct, the audit agency A will make a full payment to the server S_2 .

IV. SECURITY ANALYSIS

The security analysis consists of three attacks as follows. First, if a server has enough shadows, i.e. h clients, he can generate the correct *proof*. Otherwise, he can not generate the corresponding *proof* without receiving h clients. Second, if some clients reveal their shadow-generating function to a corrupt server in a certain time frame t , the server can not forge the shadows of these clients in other time frames. Third, if some clients denote their shadow-generating function to a corrupt server, the server can not forge the shadows of other honest clients. However, these attacks cannot have emerged in

the proposed scheme.

Attack 1: A server is not able to generate the correct *proof* without receiving enough shadows.

In order to recover secret K , $G(j \circ t)$ in t needs to be found. If $(K_{i1,j}^t, K_{i2,j}^t, \dots, K_{ih,j}^t)$ are revealed, by CRT, K modulo M_1 can be known where $M_1 = d_{i1t} d_{i2t} \dots d_{iht}$. As $M_1 \geq M$, can be uniquely determined $G(j \circ t)$ and K . On the other hand, if only $h - 1$ or less shadows are revealed, essentially no information about the secret K can be found.

Attack 2: If some clients reveal their shadow-generating function, then the probability of a corrupt server being able to forge the shadows of these clients in other time frames can be ignored.

We assume that there are n clients in a metering system. When a client C_i joins this metering system, he will receive his own shadowing-generating function G and an interval (d_{iva}, d_{ivb}) from the audit agency A . Let R be a prime and $d_{iva} < R < d_{ivb}$. If a client C_i reveals its shadow-generating function $G(z)$ to a corrupt sever in a certain time frame t , the maximum possibility that the server can guess the correct shadows is $\frac{1}{nR}$ and the server only has one time to guess the possible d_{it} during another time frame t_2 .

Attack 3: If some clients denote their shadow-generating function to a corrupt server, the server can not forge the shadows of other honest clients.

A client C_{i1} denotes its shadow-generating function $G(z)$ to a server. We assume that a metering system has n clients and the average number of prime numbers between d_{i1va} and d_{i1vb} is R . When the corrupt server gets the shadow-generating function, it may want to get the shadow of another client C_{i2} . The max possibility that the server can guess the correct shadows is $\frac{1}{n} \times \frac{1}{nR}$ and the server only has one time to guess the possible d_{i2t} during another time frame t .

V. CONCLUSIONS

In this paper, we have introduced a formal transformation from the threshold scheme to the metering scheme. Then, a practical metering scheme is proposed by using these transformation rules to overcome the drawbacks in [2], [3], [4], [5], [8], [9].

ACKNOWLEDGMENT

The authors would like to thank this work was supported by the National Science Council, R.O.C., under contract No.NSC-94-2218-E-150-001 and NSC-94-2218-E-006-021.

REFERENCES

- [1] C. Asmuth, and J. Bloom, "A modular approach to key safeguarding," IEEE Transaction on Information Theory, Vol. IT-29, No. 2, pp. 208-210, Mar. 1983.
- [2] C. Blundo, A. De Bonis and B. Masucci, "Metering Schemes with Pricing," in Proc. International Symposium on Distributed Computing - DISC 2000, LNCS-1914, pp. 194-208, 2000.
- [3] C. Blundo, A. De Bonis, B. Masucci and D. R. Stinson, "Dynamic Multi-Threshold Metering Schemes," in Proc. Annual Workshop on Selected Areas in Cryptography - SAC 2000, LNCS-2012, pp.130-144, 2001.
- [4] C. Blundo, S. Martin, B. Masucci and C. Padr'o, "A Linear Algebraic Approach to Metering Schemes," Submitted by Cryptology ePrint Archive: Report 2001/087.

- [5] C. Blundo, A. De Bonis and B. Masucci, "Bounds and Constructions for Metering Schemes," *Communications in Information and Systems*, Vol. 2, No. 1, pp. 1-28, 2002.
- [6] M. K. Franklin and D. Malkhi, "Auditable Metering with Lightweight Security," *Journal of Computer Security*, Vol. 6, No. 4, pp. 237-255, 1998.
- [7] D. Knuth, *The art of Computer Programming*, Vol. 2: Semi numerical Algorithms, 3rd Ed, Reading, Massachusetts, Addison-Wesley, 1997.
- [8] B. Masucci and D. R. Stinson, "Metering Schemes for General Access Structures," in *Proc. European Symposium on Research in Computer Security - ESORICS 2000*, LNCS-1895, pp. 72-87, 2000.
- [9] B. Masucci and D. R. Stinson, "Efficient metering schemes with pricing," *IEEE Transactions on Information Theory*, Vol. 47 Issue.7, pp. 2835-2844, 2001.
- [10] M. Naor and B. Pinkas, "Secure and Efficient Metering schemes," in *EUROCRYPT '98*, LNCS-1403, pp. 576-590, 1998.
- [11] T. P. Novka and D. L. Hollman, "New metrics for new media: toward the development of Web measurement standards," *World Wide Web Journal*, Vol. 2 No. 1, pp. 213-246, 1997.
- [12] W. Ogata and K. Kurosawa, "Provably Secure Metering Schemes," in *Proc. Advances in Cryptology - ASIACRYPT 2000*, LNCS-1976, pp. 388-398, 2000.
- [13] W. Ogata and K. Kurosawa, "Bounds for Robust Metering Schemes and Their Relationship with A2-code," *Proceedings of Advances in Cryptology - ASIACRYPT 2002*, LNCS-2501, pp. 64-80, 2002.
- [14] J. Pitkow, "In search of reliable usage data on the WWW," *The Sixth International WWW conf.*, Apr. 1997.
- [15] A. Shamir, "How to share a secret," *Communications of the ACM*, Vol. 22, No.11, pp. 612-613, Nov., 1979.