$$\frac{V_{o2}}{V_{in}} = \frac{G_1G_2G_3}{s^2C_1C_2G_1 + sC_2G_2G_3 + G_1G_2G_3} \qquad (2)$$

and

$$\frac{V_{o3}}{V_{in}} = \frac{-sC_2G_1G_2}{s^2C_1C_2G_1 + sC_2G_2G_3 + G_1G_2G_3} \qquad (3)$$

Thus, the circuit realises a noninverting notch signal at V_{o1}, a noninverting lowpass signal at V_{o2} and an inverting bandpass signal at V_{o3}. Component matching conditions are not required in the design.

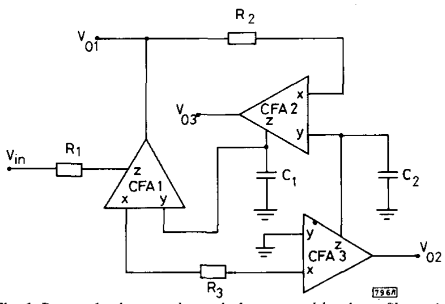

Fig. 1 *Proposed voltage-mode notch, lowpass and bandpass filter using three current-feedback amplifiers*

Taking into account the nonidealities of a CFA, namely, $i_z = \alpha i_x$, $v_x = \beta v_y$ and $v_o = \gamma v_z$ where $\alpha = 1-e_i$ and $e_i(>1)$ denotes the current tracking error, $\beta = 1-e_v$ and $e_v(>1)$ denotes the input voltage tracking error and $\gamma = 1-e'_v$ and $e_v(>1)$ denotes the output voltage tracking error. The resonance angular frequency ω_o and quality factor Q are given by

$$\omega_o = (G_2G_3/C_1C_2)^{1/2}(\alpha_2\alpha_3\beta_1\beta_2)^{1/2} \qquad (4)$$

and

$$Q = G_1(C_1/C_2G_2G_3)^{1/2}(1/\alpha_1\gamma_1)(\alpha_3\beta_2/\alpha_2\beta_1)^{1/2} \qquad (5)$$

Note that ω_o and Q are orthogonally adjustable. The sensitivities of ω_o and Q to active and passive components are

$$S^{\omega_o}_{\alpha_2,\alpha_3,\beta_1,\beta_2} = 1/2 = S^Q_{\alpha_3,\beta_2} = -S^Q_{\alpha_2,\beta_1} \qquad S^Q_{\alpha_1,\gamma_1} = -1$$
$$S^{\omega_o}_{G_2,G_3} = -S^{\omega_o}_{C_1,C_2} = 1/2 \quad S^Q_{C_2,G_2,G_3} = -S^Q_{C_1} = -1/2$$
$$\text{and} \quad S^Q_{G_1} = 1$$

all of which are small.

Finally, to verify the theoretical prediction of the proposed network, a lowpass filter prototype has been realised with discrete components. The experimental network in Fig. 1 was built with $R_1 = R_2 = R_3 = 10\text{k}\Omega$, and $C_1 = C_2 = 1\text{nF}$. Three commercial CFAs (AD844s) were used. The measured frequency response of the lowpass filter shown in Fig. 2 agrees well with theory.

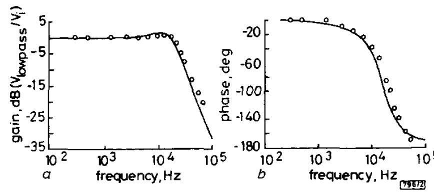

Fig. 2 *Amplitude-frequency response and phase-frequency response*

a Amplitude-frequency
b Phase-frequency

——— ideal response
O measured response

Conclusions: In this Letter, three current-feedback amplifiers, two grounded capacitors and three floating resistors are employed to construct a voltage-mode notch, lowpass and bandpass filter in which the CFA-based notch filter is proposed for the first time. The new filter offers the following advantageous features:

(i) realisation of notch, lowpass and bandpass signals from the same configuration

(ii) no requirements for critical component matching conditions

(iii) orthogonal adjustment of ω_o and Q

(iv) use of only two grounded capacitors which makes the circuit suitable for integrated-circuit implementation

(v) small active and passive sensitivities

(vi) very low output impedance which makes the voltage-mode circuit cascadable.

28 September 1994
Electronics Letters Online No: 19941416

C.-M. Chang, C.-S. Hwang and S.-H. Tu (*Dept. of Electrical Engineering, Chung Yuan Christian University, Chung-Li, Taiwan 32023, Republic of China*)

References

1 WILSON, B.: 'Recent developments in current conveyors and current-mode circuits', *IEE Proc. G*, 1990, **137**, (2), pp. 63–77
2 SVOBODA, J.A., MCGORY, L., and WEBB, S.: 'Applications of a commercially available current conveyor', *Int. J. Electron.*, 1991, **70**, (1), pp. 159–164
3 FABRE, A.: 'Gyrator implementation from commercially available transimpedance operational amplifiers', *Electron. Lett.*, 1992, **28**, (3), pp. 263–264
4 FABRE, A.: 'Insensitive voltage-mode and current-mode filters from commercially available transimpedance opamps', *IEE Proc. G*, 1993, **140**, (5), pp. 319–321
5 LIU, S.J., and HWANG, Y.S.: 'Realisation of R-L and C-D impedances using a current feedback amplifier and its applications', *Electron. Lett.*, 1994, **30**, (5), pp. 380–381
6 BHUSAN, M., and NEWCOMB, R.W.: 'Grounding of capacitors in integrated circuits', *Electron. Lett.*, 1967, **3**, pp. 148–149
7 PAL, K., and SINGH, R.: 'Inductorless current conveyor allpass filter using grounded capacitors', *Electron. Lett.*, 1982, **18**, pp. 47

Construction of dynamic threshold schemes

H.-M. Sun and S.-P. Shieh

Indexing term: Cryptography

An (m, n) threshold scheme is to decompose a shared secret into n shares in such a way that the shared secret cannot be reclaimed unless any m shares are collected. A new dynamic threshold scheme that allows the shared secret to be updated without changing the shares is proposed.

Introduction: In 1979, Blakley and Shamir [1, 2] introduced the concept of threshold schemes which are mainly used to protect the master key of a secure system from being lost, destroyed and modified. The main idea underlying an (m, n) threshold scheme is to divide the shared secret (master key) K into n shares S_is $(1 \le i \le n)$ in such a way that the shared secret K cannot be reclaimed unless m shares are collected. The security of a threshold scheme is classified into two levels: information theoretic security (perfect security) and computational security [3]. A threshold scheme is perfectly secure if any $m-1$ or less shares provide no information about the shared secret K [4], and it is computationally secure if for any $m-1$ or less shares it is computationally infeasible to determine the shared secret K in polynomial time [5].

In conventional threshold schemes, the corresponding shares must be updated accordingly when the shared secret is renewed. It is time-consuming and inconvenient if the shares need be changed frequently, especially when the number n of the shares is large. In

1989 [6], Laih *et al.* proposed the concept of dynamic threshold schemes in which the shared secret can be renewed while the originally issued shares remain intact. In their paper, they proposed a relatively dynamic threshold scheme in which the secrecy decreases as the number of changes to the shared secret increases. However, a perfectly secure dynamic threshold scheme has not been proposed. In 1994, Sun and Shieh [7] showed that the necessary condition for the existence of a perfectly secure dynamic threshold scheme is that the length of the share should be $T+1$ times longer than that of the shared secret, where T ($T \geq 0$) denotes the times that the shared secret can be renewed. Thus, the length of the share and the length of the shared secret determine the times that the shared secret can be renewed. This suggests that perfectly secure dynamic threshold schemes are infeasible because the length of a share is proportional to T. Recently, Zheng *et al.* [3] proposed a computationally secure (m, n) dynamic threshold scheme based on the use of the pseudorandom function family and the universal hash function family. Their dynamic threshold scheme has the advantages that the length of the share is constant, and the times that the shared secret can be renewed are unlimited. However, their (m, n) dynamic threshold scheme is somewhat limited because

(i) the number n is constrained to $n = O(l)$, where l denotes the length of the shared secret

(ii) it needs to maintain a large public function with the size of $O(2^n)$, where n denotes the number of shareholders.

In this Letter, we construct a computationally secure (m, n) dynamic threshold scheme which provides the same functions but better strength. Our scheme not only resolves the two problems described above, but also detects and identifies any cheater who attempts to deceive other shareholders by presenting a forged share in the threshold scheme. Our scheme is based on the difficulty of solving the discrete logarithm [8] and can be described in terms of the following three phases.

(*a*) *Initial phase:* Let p be a large prime (e.g. the length of p is 512 bit) such that the discrete logarithm problem (mod p) is inaccessible [8], and g be a primitive element over $GF(p)$. Each shareholder U_i has a public share y_i ($= g^{S_i} \bmod p$) and a secret share S_i, where S_i is randomly chosen between 0 and $p-1$.

(*b*) *Dispersion phase:* We assume that the shared secret is K, where $0 \leq K \leq p-1$. If K is larger than $p-1$, it is divided into blocks, each of which is smaller than p. Each block is protected by the dynamic threshold scheme. The dealer takes the following steps to set up the relationship between the shared secret K and the shares S_is. Note that this phase needs to be repeated only when the shared secret K needs to be renewed.

(i) The dealer selects a polynomial of degree $m-1$, $f(x) = a_{m-1} \cdot x^{m-1} + \cdots + a_1 \cdot x + K \pmod p$, where $a_1, \cdots, a_{m-1}$ are randomly chosen between 0 and $p-1$.

(ii) The dealer selects a random number r and publishes the value of d ($= g^r \bmod p$).

(iii) The dealer computes $b_i = f(i) \cdot (y_i)^r \pmod p$ and $c_i = g^{f(i)} \pmod p$ for $1 \leq i \leq n$, and then publishes b_i and c_i to all shareholders.

(*c*) *Recovery phase:* Without loss of generality, we assume that U_1, U_2, …, and U_m are m shareholders who want to restructure the shared secret K. Each shareholder U_i who owns S_i first obtains the value of $(y_i)^r$ by computing $(d)^{S_i} = (g^r)^{S_i} = (g^{S_i})^r = (y_i)^r \pmod p$, and then computes $f(i) = b_i \cdot [(y_i)^r]^{-1} \pmod p$, for $1 \leq i \leq m$. To detect the cheater, we need only to verify the validity of $f(i)$ by $g^{f(i)} \stackrel{?}{=} c_i$, for $1 \leq i \leq m$. If all these $f(i)$s are valid, $f(x)$ can be reconstructed from the Lagrange interpolating polynomial as follows:

$$f(x) = \sum_{k=1}^{m} (f(i) \cdot \prod_{j=1, j \neq k}^{m} \frac{(x-j)}{(k-j)}) \pmod p$$

Therefore, the shared secret K can be obtained by $f(0)$.

In the three phases above, the public information of our scheme is p, g, d, y_i, b_i, c_i for $1 \leq i \leq n$. It is clear that the size of the public information of our scheme is $O(n)$, where n denotes the number of the shareholders.

Security analysis:

(i) Without loss of generality, we assume that U_1, U_2, …, and U_{m-1} are $m-1$ shareholders who attempt to restructure the shared secret K. In this case, they can obtain only $m-1$ values $f(i)$ for $1 \leq i \leq m-1$. Thus $f(x)$ cannot be determined uniquely because $f(x)$ has m unknown variables.

(ii) We assume that an intruder (who may or may not be a shareholder) who does not know S_i attempts to obtain the value of $f(i)$. He knows only the public information d ($= g^r$), y_i ($= g^{S_i}$), c_i ($= g^{f(i)}$), and b_i ($= f(i) \cdot (y_i)^r$ or $f(i) \cdot (d)^{S_i}$). To derive r or S_i or $f(i)$ from d, y_i, and c_i, the intruder has to cope with the difficulty of solving the discrete logarithm problem [8]. Hence, it is infeasible for the intruder to obtain any one of r, S_i, and $f(i)$. In addition, he cannot derive $f(i)$ from b_i because r and S_i are unknown.

(iii) Consider the case when the shared secret is renewed in the threshold scheme. Assume that K' is the new shared secret and $h(x)$ is the new polynomial in the dispersion phase. Because $h(x)$ is independent of $f(x)$, the only possibility of finding K' ($= h(0)$) is to find S_i from $f(x)$ first and then derive $h(x)$ from S_i. We also assume that the polynomial $f(x)$ is public after the shared secret is renewed. Thus, $f(i)$ ($1 \leq i \leq n$) can be computed by any intruder. This implies that the value of $(d)^{S_i}$ ($= f(i)^{-1} \cdot b_i \bmod p$) can be computed by any intruder. However, it is infeasible for the intruder to compute S_i from the value of $(d)^{S_i}$ because the intruder faces the difficulty of solving the discrete logarithm problem again [8].

(iv) We assume that there exists a cheater, the shareholder U_i, who presents a forged share S_i^* in the recovery phase. The forged share S_i^* gives the forged value $f(i)^*$. The forged value $f(i)^*$ will not satisfy the equation: $g^{f(i)^*} = c_i$ because g and c_i are public. Therefore, the behaviour of the cheater can be detected.

Conclusions: In this Letter, we propose a new (m, n) dynamic threshold scheme which allows the shared secret to be renewed without changing the shares. The scheme has the advantages that the length of the share is constant, and the times that the shared secret can be renewed are unlimited. The public information of our dynamic threshold scheme is $O(n)$ which is better than $O(2^n)$ of the existing scheme, where n denotes the number of shareholders. In addition, our dynamic threshold scheme has the capability of detecting cheaters who attempt to deceive other shareholders.

Acknowledgments: This research was supported by the National Science Council of Taiwan, ROC under contract NSC-83-0404-E-009-106.

17 October 1994
Electronics Letters Online No: 19941411

H.-M. Sun and S.-P. Shieh (*Department of Computer Science and Information Engineering, National Chiao Tung University, Hsinchu 30050, Taiwan*)

References

1 BLAKLEY, G.R.: 'Safeguarding cryptographic keys'. Proc. NCC, 1979, (AFIPS Press, Montvale, NJ), Vol. 48, pp. 313–317

2 SHAMIR, A.: 'How to share a secret', *Commun. ACM*, 1979, **22**, (11), pp. 612–613

3 ZHENG, Y., HARDJONO, T., and SEBERRY, J.: 'Reusing shares in secret sharing schemes', *The Computer Journal*, 1994, **37**, (3), pp. 199–205

4 SHANNON, C.E.: 'Communication theory of secrecy systems', *Computer Security Journal*, 1990, **VI**, (2), pp. 7–66

5 DENNING, D.E.: 'Cryptography and data security' (Reading, Addison-Wesley, 1983)

6 LAIH, C.S., HARN, L., LEE, J.Y., and HWANG, T.: 'Dynamic threshold scheme based on the definition of cross-product in an n-dimensional linear space', *J. Information Science and Engineering*, 1991, **7**, pp. 13–23 (Also appears in Advances in cryptology: Eurocrypt'89, Springer-Verlag, Berlin, 1990, pp. 286–298)

7 SUN, H.M., and SHIEH, S.P.: 'On dynamic threshold schemes', to appear in *Inf. Process. Lett.*, 1994

8 ELGAMAL, T.: 'A public key cryptosystem and a signature scheme based on discrete logarithms', *IEEE Trans.*, 1985, **IT-31**, pp. 469–472

Design of generalised ElGamal type digital signature schemes based on discrete logarithm

L. Harn and Y. Xu

Indexing term: Cryptography

The ElGamal type digital signature schemes have received wide attention recently. ElGamal type signature schemes can provide 'subliminal' channel, message recovery, multisignature, etc. The authors investigate the design criteria of ElGamal type signature scheme and develop a complete list of all variations.

Introduction: A digital signature is analogous to an ordinary written signature used for signing messages. It must be unique and private to the user. At this time, there are two most popular public-key algorithms which can provide a digital signature: the RSA scheme [1]; and the ElGamal scheme [2].

A modification of the ElGamal signature was proposed by Agnew, Mullin and Vanstone (AMV) [3] in 1990. Instead of solving $m = xr + ks \bmod p-1$, the signer solves the congruence $m = xs + kr \bmod p-1$. The advantage of this modified scheme over the ElGamal scheme is that, in order to compute the signature by solving the congruence for s, the signer only needs to compute x^{-1} in Z_{p-1}^* once, instead of computing k^{-1} in Z_{p-1}^* for every signature, where x is the secret key for the signer and k is an integer randomly selected by the signer for signing every message. Yen and Laih [4] also proposed a variation of the ElGamal type signature scheme. In 1994, Harn [5, 6] proposed two other variations of ElGamal type schemes.

In 1989, Schnorr [7] proposed an ElGamal type signature scheme to shorten the signature. Later, the digital signature algorithm (DSA) was proposed [8] by NIST, also based on a very similar approach. These two schemes have also been developed based on the original ElGamal signature scheme.

A recent paper, Nyberg and Rueppel [9], pointed out that all ElGamal type signature schemes have variants giving message recovery and also analysed six of the simplest ElGamal type variations in $GF(p)$. Being motivated by their paper, we have developed a complete list of 18 ElGamal type signature schemes in this Letter.

Generalised ElGamal type signature schemes: Let p be a large prime and α be a primitive number in $GF(p)$. Each user selects a secret key $x \in [1, p-1]$ and computes a public key $y = \alpha^x \bmod p$. For each message $m \in [1, p-1]$ to be signed a new random integer $k \in [1, p-1]$ is privately selected. Instead of signing the message m directly, all ElGamal type signature schemes should sign the one-way hash result of m. For simplicity, we will ignore the one-way hash function in the following discussion.

In all ElGamal type signature schemes [3–9] the commitment part r of the signature is computed as

$$r = \alpha^k \bmod p$$

The other part s of the signature is computed differently. In the original ElGamal scheme, s is solved with the knowledge of the signer's secrets, x and k, as

$$m = ks + rx \bmod \varnothing(p)$$

where k should be selected such that $\mathrm{GCD}(k, \varnothing(p)) = 1$. The triplet $(m; (r,s))$ constitutes the signed message and is sent to the verifier.

The signature (r, s) is accepted by evaluating whether the equality

$$\alpha^m = r^s y^r \bmod p$$

holds true.

Without loss of generality, we can represent the generalised equation for all ElGamal type signature schemes as

$$ax = bk + c \bmod \varnothing(p)$$

where (a, b, c) are three parameters from the set of values (m, r, s). More specifically, each parameter can be a mathematical combination of (m, r, s). For example, the parameter a can be rm, or r, etc. The verification equation is determined accordingly as

$$y^a = r^b \alpha^c \bmod p$$

In the following we will discuss the form of the above generalised signature equation and some restrictions applied on parameters (a, b, c) based on the security considerations.

(*a*) Because x and k are two secret numbers and the verifier does not know these two values, x and k should be treated as two different terms in the above equation. Otherwise, if we combine these two secret parameters together (i.e. for example, if $xk = rm + s \bmod \varnothing(p)$, then $y^k = \alpha^{rm+s} \bmod p$ or $r^x = \alpha^{rm+s} \bmod p$), the verifier cannot verify the signature.

(*b*) To claim that (r, s) is a signature for the message, the message m itself should be included in the signature equation and can be in any of parameters (a, b, c).

(*c*) To provide proper security of algorithms, r and s should also be included in parameters (a, b, c). Thus, there are five parameters in the equation. If r is contained in parameter b, the verification equation is very similar to the scheme proposed by Agnew, Mullin and Vanstone [3], in which r will appear in both the base and the exponent of the same base (i.e. $\alpha^m = y^s r^r \bmod p$). Otherwise, r will appear in both the base and the exponent of a different base (i.e. $\alpha^m = r^s y^r \bmod p$) as in the original ElGamal scheme [2].

(*d*) For security reasons, s and m cannot be combined together in any of parameter (a, b, c). For example, if $x = rk + sm \bmod \varnothing(p)$. Then only by modifying the partial signature s of a legitimate signature (r, s) corresponding to the message m, can it forge a signature (r, s') of another message m', where $m' = \beta m \bmod \varnothing(p)$ and $s' = \beta^{-1} s \bmod \varnothing(p)$.

(*e*) For security reasons, s and r cannot be combined together. For example, if $mx = k + rs \bmod \varnothing(p)$ and the corresponding verification equation is $y^m = r\alpha^{rs} \bmod p$. The attacker can first randomly select an integer R and computes r' to satisfy $y^m = r'\alpha^R \bmod p$. The forged signature is (r', s'), where $r's' = R \bmod \varnothing(p)$.

(*f*) r can be combined with m. For example, if $x = rmk + s \bmod \varnothing(p)$. This is due to the fact that the partial signature r is locked by the secret number k and it is impossible to forge a signature by changing r only.

(*g*) There must be three separate terms as specified in the equation. For example, if $(m+r)x = sk \bmod \varnothing(p)$, then it can forge signature (r, s') for another message m', where $m-m' = \beta \bmod \varnothing(p)$ and $s' = (1 - \beta(m+r)^{-1})s \bmod \varnothing(p)$.

(*h*) The generalised signature equation contains five parameters: three parameters, (m, r, s), are public information, x is the fixed secret key of the signer and k is a random secret value for each message. Because the number of secret parameters is always one larger than the number of linear equations available to the attacker, the signature scheme is secure based on the discussion in the original ElGamal paper [2].

If we neglect the difference between $+d$ and $-d$, and the difference between d and d^{-1}, where $d \in (x, k, m, r, s)$, we can list all