Gong Harn Public-key Cryptosystem (Gong-Harn, 1999, 2001)

Susana Sin
Department of Electrical and Computer Engineering
University of Waterloo
Waterloo, ON, N2T 2G1, Canada
Email: ssjsin@engmail.uwaterloo.ca

Introduction

- Gong-Harn Public Key Cryptosystem (GH-PKC) is based on third-order linear feedback shift register (LFSR) sequence with a particular phase.
- Security is based on the difficult in solving discrete logarithm (DL) problem in GH(q^3) where $q = p$ or $q = q^2$, depending on the implementation, and p is a prime.
- For an implementation of GH-PKC over GF(p), the security of the implementation is based on the difficult in solving DL problem in GF(p^3).
  - ie. In order to implement the GH-PKC over GF(p) with 1024-bit security, a 341-bit p is required!

Third-order Characteristic Sequence

- Irreducible polynomial $f(x)$ of degree 3 over GF(p):

$$f(x) = x^3 - ax^2 + bx - 1$$

- If initial state is:

$$s_0 = 3, s_1 = a, s_2 = a^2 - 2b$$

- Then the sequence generated by $f(x)$ is called a third-order characteristic sequence.
- We denote the k^{th} term in the sequence generated by $f(x)$ as:

$$s_k = s_k(a,b)$$

Profile of Third-Order Characteristic Sequences

- Period Q is factor of p^2+p+1
- Trace Representation:

$$s_k = Tr(\alpha^k) = \alpha^k + \alpha^{kq} + \alpha^{kq^2}, k = 0,1,\ldots$$

where α is a root of $f(x)$ in the extension field GF(q^3).

Reciprocal Sequence

- Given the $f(x)$ above, the reciprocal polynomial is:

$$f^{-1}(x) = x^3 - bx^2 + ax - 1$$

- By choosing the corresponding initial states as shown above, the sequence generated by $f^{-1}(x)$ is also a third order characteristic sequence.
- The k^{th} generated by $f^{-1}(x)$ is the $-k^{th}$ term generated by $f(x)$:

$$s_k(b,a) = s_{-k}(a,b)$$

Commutative Law

- Let $f(x) = x^3 - ax^2 + bx - 1$ be irreducible over GF(q) and $\{s_i\}$ be the characteristic sequence generated by $f(x)$. Then for any positive integers k and e:

$$s_k(s_e(a,b), s_{-e}(a,b)) = s_{ek}(a,b)$$

Dual State Fast Evaluation Algorithm (DSEA)

- To compute the $s_{\pm k}$ sequence terms.
- Binary representation of k:

$$k = \sum_{i=0}^{n} k_i 2^{n-i} = k_0 2^n + k_1 2^{n-1} + \cdots + k_n$$

- Let $T_0 = k_0 = 1$ and $T_j = k_j + 2T_{j-1}$ for $1 \le j \le n$. $\Rightarrow T_n = k$. Let $t = T_{j-1}$ and $t' = T_j$
- For $k_j = 0$:

$$s_{t'+1} = s_t s_{t+1} - a s_{-t} + s_{-(t+1)}$$
$$s_{t'} = s_t^2 - 2s_{-t}$$
$$s_{t'-1} = s_t s_{t-1} - b s_{-t} + s_{-(t+1)}$$

For $k_j = 1$:

$$s_{t'+1} = s_{t+1}^2 - 2s_{-(t+1)}$$
$$s_{t'} = s_t s_{t+1} - a s_{-t} + s_{-(t-1)}$$
$$s_{t'-1} = s_t^2 - 2s_{-t}$$

Computation of a Previous Sequence Term

- Given (s_k, s_{k+1}) and its dual. Determine $s_{\pm(k-1)}$ terms.
- Let $delta = s_{k+1}s_{-(k+1)} - s_1 s_{-1}$
- Then $s_{\pm(k-1)}$ terms can be computed by:

$$s_{k-1} = \frac{e s_{-(k+1)} - s_{-1} D(e)}{delta}$$
$$s_{-(k-1)} = \frac{D(e) s_{k+1} - s_1 e}{delta}$$

$$D(s_k) = s_{-k}$$
$$e = -s_{-1}D(c_1) + c_2$$
$$c_1 = s_1 s_{k+1} - s_{-1} s_k$$
$$c_2 = s_k^2 - 3s_{-k} + (b^2 - a)s_{-(k+1)}$$

- Note that delta cannot be zero!
- Experimental data shows that *delta* will be zero if k is either $p-1$ or p.
- If either a or b is zero, *delta* will be zero if either s_k or s_{k+1} is zero.

Computation of Mixed Terms $s_{\pm u(k+v)}$ using (s_{k-1}, s_k, s_{k+1}) and its dual

1. Compute the sequence terms $s_{\pm(k+v)}$
   - Use a general result for LFSR sequence:
   - Define *Transitional Matrix A* and *State Matrix Mn*:

$$A = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & -b \\ 0 & 1 & a \end{bmatrix} \qquad M_n = \begin{bmatrix} s_{n-2} & s_{n-1} & s_n \\ s_{n-1} & s_n & s_{n+1} \\ s_n & s_{n+1} & s_{n+2} \end{bmatrix}$$

   - Note:

$$\underline{s}_k \cdot A = (s_k, s_{k+2}, s_{k+2}) \cdot \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & -b \\ 0 & 1 & a \end{bmatrix}$$
$$= (s_{k+1}, s_{k+3}, s_k - bs_{k+1} + as_{k+2}) = \underline{s}_{k+1}$$

   - Two properties:

i. $\underline{s}_v = \underline{s}_0 \cdot A^v = \underline{s}_1 \cdot A^{v-1} = \underline{s}_2 \cdot A^{v-2} = \cdots = \underline{s}_{v-1} \cdot A^1$

ii. $M_v = M_0 \cdot A^v \Rightarrow A^v = M_0^{-1} \cdot M_v$,*if* $\det(M_0) \neq 0$

   - If $\det(M_0) \neq 0$, then

$$\underline{s}_{k+v} = \underline{s}_k \cdot A^v = \underline{s}_k \cdot (M_0^{-1} \cdot M_v) = \underline{s}_k \cdot \left(\begin{bmatrix} s_{-2} & s_{-1} & s_0 \\ s_{-1} & s_0 & s_1 \\ s_0 & s_1 & s_2 \end{bmatrix}^{-1} \begin{bmatrix} s_{v-2} & s_{v-1} & s_v \\ s_{v-1} & s_v & s_{v+1} \\ s_v & s_{v+1} & s_{v+2} \end{bmatrix} \right)$$

   where (s_{v-1}, s_v, s_{v+1}) and its dual can be computed using DSEA algorithm and

$$s_{v-2} = s_{v+1} - as_v + bs_{v-1}$$
$$s_{v+2} = as_{v+1} - bs_v + s_{v-1}$$

   - In particular, the s_{k+v} term is equal to $\underline{s}_{k-1}$ multiplied by the middle column of $M_0^{-1} \cdot M_v$
   - For Matrix M0 to be invertible, we need $\det(M_0) \neq 0$:

$$\det(M_0) = (b^2 - 2a)\left[3(a^2 - 2b) - a^2\right] - b\left[b(a^2 - 2b) - 3a\right] + 3\left[ab - 3 \cdot 3\right] \neq 0$$

   - By choosing a and b correspondingly, *det(M_0)* can be guaranteed to be non-zero!
2. Compute $s_{\pm u}(s_{(k+v)}, s_{-(k+v)})$ using DSEA algorithm

GH Diffie-Hellman Key Agreement Protocol

System Parameters:

$f(x) = x^3 - ax^2 + bx - 1$, an irreducible polynomial over GF(p), where p is a prime number. Period of the third-order characteristic sequence is denoted by Q.

	Alice	**Bob**
Private Key:	Chooses K_A, $0 < K_A < Q$, $\gcd(K_A, Q) = 1$	Chooses K_B, $0 < K_B < Q$, $\gcd(K_B, Q) = 1$
Public Key Pair:	(s_{K_A}, s_{-K_A})	(s_{K_B}, s_{-K_B})
	$\xrightarrow{(s_{K_A}, s_{-K_A})}$	
	$\xleftarrow{(s_{K_B}, s_{-K_B})}$	
Common Key Pair:	$s_{\pm K_A}(s_{K_B}, s_{-K_B})$	$s_{\pm K_B}(s_{K_A}, s_{-K_A})$

$$s_{\pm K_A}(s_{K_B}, s_{-K_B}) = s_{\pm K_B}(s_{K_A}, s_{-K_A})$$

GH Digital Signature Algorithm (GH DSA)

- ElGamal-like signature algorithm
- Alice:
  - Private Key: Choose x, with $0 < x < Q$ and $\gcd(x, Q) = 1$
  - Public Key: The $s_{\pm x}$ terms generate by $f(x)$
- Signing Process:
  1. Randomly choose k, with $0 < k < Q$ and $\gcd(k,Q) = 1$. Use DSEA algorithm to compute (s_{k-1}, s_k, s_{k+1}) and its dual such that $\gcd(s_k, Q) = 1$. $r = s_k$
  2. Compute $h = h(m)$, where $h()$ is a hash function
  3. Solve for t in the signing equation: $h \equiv xr + kt \bmod Q \Rightarrow t \equiv k^{-1}(h - rx) \bmod Q$
- (r, t) is the digital signature of the message m. Alice sends Bob (m, r, t) together with (s_x, s_{x+1}), (s_k, s_{k+1}) and their duals.

Signature Verification

- Verifying Process:
  1. Compute $s_{\pm(x-1)}$ and $s_{\pm(k-1)}$ using (s_x, s_{x+1}), (s_k, s_{k+1}) and their duals.
  2. If $\gcd(t, Q) = 1 \Rightarrow$ Case 1, else $\Rightarrow$ Case 2

Case 1: To verify

$$s_{\pm x} = s_{\pm r^{-1}(h - kt)} = s_{\mp r^{-1}t(-ht^{-1} + k)} = s_{\pm u(k+v)}$$

i. Compute $u = -r^{-1}t \bmod Q$, $v = -ht^{-1} \bmod Q$

ii. Compute mixed terms $s_{\pm u(k+v)}$

iii. Verify sequence terms

Case 2: To verify

$$s_{\pm kt} = s_{\pm(h - rx)} = s_{\mp r(-hr^{-1} + x)} = s_{\pm u(x+v)}$$

i. Compute $u = -r \bmod Q$, $v = -hr^{-1} \bmod Q$

ii. Compute mixed terms $s_{\pm u(x+v)}$

iii. Compute $s_{\pm kt}$

iv. Verify sequence terms

References:

- G. Gong and L. Harn, A new approach for public key distribution, *Proceedings of China-Crypto'98*, May 1998, Chengdu, China
- G. Gong and L. Harn, The GH public-key cryptosystems, the Proceedings of the 8th Annual Workshop on Selected Areas in Cryptography, Toronto, Aug 16-18, 2001