Bahan Kuliah IF4020 Kriptografi

Elgamal Signature Scheme dan Schnorr Signature Scheme

Oleh: Rinaldi Munir

Program Studi Teknik Informatika ITB
2024

Sumber materi di dalam PPT ini:

1. Fatimah Al-Ubaidy, *Digital Signature*, in course *Data Security*, Mustansiriyah University, Faculty of Engineering, Computer Engineering Dept.
2. Chirag's Blog, Elgamal and Schnorr scheme of Digital Signature | Which scheme is best Elgamal or Schnorr?, https://www.chiragbhalodia.com/2021/11/elgamal-schnorr-scheme.html

ELGAMAL DIGITAL SIGNATURE SCHEME

Elgamal encryption scheme is designed to enable encryption by a user's public key with decryption by the user's private key. The Elgamal signature scheme involves the use of the private key for digital signature generation and the public key for digital signature verification.

Preliminary:

If q is a prime number and α is a primitive root of q, then

1. For any integer m, $\alpha^m \equiv 1 \pmod{q}$ if and only if $m \equiv 0 \pmod{q-1}$.
2. For any integers, i, j, $\alpha^i \equiv \alpha^j \pmod{q}$ if and only if $i \equiv j \pmod{q-1}$.

Elgamal Key Generation:

As with Elgamal encryption, the global elements of **Elgamal digital signature** are a prime number ***q*** and **α**, which is a primitive root of ***q***. User **A** generates a private/public key pair as follows:

1. Generate a random integer X_A, such that $1 < X_A < q - 1$.
2. Compute $Y_A = \alpha^{X_A} \bmod q$.
3. A's private key is X_A; A's pubic key is $\{q, \alpha, Y_A\}$.

Elgamal Digital Signature Generation:

To sign a message ***M***, user **A** first computes the hash ***m* = H(*M*)**, such that ***m*** is an integer in the range **$0 \le m \le q - 1$**. **A** then forms a digital signature as follows:

Sumber: Fatimah Al-Ubaidy, Digital Signature

Elgamal Digital Signature Generation (Continued):

1. Choose a random integer K such that $1 \leq K \leq q - 1$ and $\gcd(K, q - 1) = 1$. That is, K is relatively prime to $q - 1$.
2. Compute $S_1 = \alpha^K \bmod q$. Note that this is the same as the computation of C_1 for Elgamal encryption.
3. Compute $K^{-1} \bmod (q - 1)$. That is, compute the inverse of K modulo $q - 1$.
4. Compute $S_2 = K^{-1}(m - X_A S_1) \bmod (q - 1)$.
5. The signature consists of the pair (S_1, S_2).

Elgamal Digital Signature Verification:

Any user **B** can verify the signature as follows:

1. Compute $V_1 = \alpha^m \bmod q$.
2. Compute $V_2 = (Y_A)^{S_1}(S_1)^{S_2} \bmod q$.

The signature is valid if $\boldsymbol{V_1} = \boldsymbol{V_2}$. Let us demonstrate that this is so.

Example (1):

Let us start with the prime field GF(19); that is, $\boldsymbol{q}$ = 19. It has primitive roots {2, 3, 10, 13, 14, 15}. We choose $\boldsymbol{\alpha}$ = 10.

Alice generates a key pair as follows:

1. Alice chooses $X_A = 16$.
2. Then $Y_A = \alpha^{X_A} \bmod q = \alpha^{16} \bmod 19 = 4$.
3. Alice's private key is 16; Alice's pubic key is $\{q, \alpha, Y_A\} = \{19, 10, 4\}$.

Sumber: Fatimah Al-Ubaidy, Digital Signature

Example (1): (Continued)

(Sign) Suppose Alice wants to sign a message with hash value $\boldsymbol{m} = 14$.

1. Alice chooses $K = 5$, which is relatively prime to $q - 1 = 18$.
2. $S_1 = \alpha^K \bmod q = 10^5 \bmod 19 = 3$ (see Table 2.7).
3. $K^{-1} \bmod (q - 1) = 5^{-1} \bmod 18 = 11$.
4. $S_2 = K^{-1}(m - X_A S_1) \bmod (q - 1) = 11\,(14 - (16)(3)) \bmod 18 = -374 \bmod 18 = 4$.

(Verify): Bob can verify the signature as follows:

1. $V_1 = \alpha^m \bmod q = 10^{14} \bmod 19 = 16$.
2. $V_2 = (Y_A)^{S_1}(S_1)^{S_2} \bmod q = (4^3)(3^4) \bmod 19 = 5184 \bmod 19 = 16$.

Thus, the signature is valid because $\boldsymbol{V_1} = \boldsymbol{V_2}$.

NIST Digital Signature Approach:

NIST has published Federal Information Processing Standard **FIPS 186**, known as the Digital Signature Algorithm (**DSA**). The **DSA** makes use of the Secure Hash Algorithm (**SHA**). The **DSA** was originally proposed in 1991. Several expanded versions of the standard were then issued as FIPS 186-2, FIPS 186-3 and FIPS 186-4 in response to public feedback concerning the security of the scheme. This latest version also incorporates digital signature algorithms based on RSA and on elliptic curve cryptography.

Sumber: Fatimah Al-Ubaidy, Digital Signature

SCHNORR DIGITAL SIGNATURE SCHEME

- Seperti Elgamal, skema tanda-tangan Schnorr juga didasarkan pada logaritma diskrit
- Skema Schnorr meminimumkan komputasi yang dibutuhkan dalam pembangkitan tanda-tangan digital

- Pekerjaan utama untuk pembangkitan tanda-tangan digital tidak bergantung pada pesan dan dapat dilakukan selama waktu diam (idle) prosesor.

- Skema Schnorr didasarkan pada penggunaan sebuah bilangan prima p, yang memiliki faktor prima (p - 1) dari sebuah integer q; yaitu $p \equiv 1 \pmod{q}$. Secara khusus kita menggunakan $p = 2^{1024}$ dan $q = 2^{160}$. Jadi, p adalah bilangan 1024-bit number, dan q adalah bilangan 160-bit, yang juga adalah Panjang nilai hash SHA-1

Algoritma

Pembangkitan Kunci Publik dan Kunci Privat:

Step-1: Pilih bilangan prima p dan q, sedemikian sehingga q adalah faktor prima dari p-1, yaitu $p \equiv 1 \pmod{q}$.

Step-2: Pilih sebuah integer α, sedemikian sehingga $\alpha^q \equiv 1 \pmod{p}$. Nilai α, p, dan q adalah *global public key* yang dapat digunakan bersama di dalam kelompok pengguna.

Step-3: Pilih sebuah bilangan acak s dengan syarat $0 < s < q$. Ini adalah kunci privat pengguna.

Step-4: Hitung $v \equiv \alpha^{-s} \pmod{p}$. Ini adalah kunci publik pengguna.

Pembangkitan Tanda-tangan Digital:

Step-1: Pilih bilangan acak r dengan syarat $0 < r < q$ dan hitung $x = \alpha^r \bmod p$. Ini adalah tahap pre-processing yang independent dari pesan M yang ditandatangani.

Step-2: Sambungkan (concate) pesan M dengan x lalu hitung nilai hashnya:

$$e = H(M \,||\, x)$$

Step-3: Hitung $y = (r + se) \bmod q$. Tanda-tangan digital terdiri dari pasangan (e, y).

Verifikasi tanda-tangan digital

Step-1: Hitung x’

$x' = \alpha^{y} v^{e} \bmod p$

Step-2: Verifikasi e = H (M || x’).

Di sini, H (M || x’) = H (M || x).

Untuk memastikan verifikasi ini bekerja, amatilah bahwa

$x' \equiv \alpha^{y} v^{e} \equiv \alpha^{y} \alpha^{-se} \equiv \alpha^{r} \equiv x \pmod p$