Part I

Discrete Logarithm Problem

Schnorr Identification Scheme

Suppose a card reader wants to authenticate a card.

System parameters: a group $\mathcal{G}$ of order q and a generator g of $\mathcal{G}$.

The card (C) holds the private key, $x \in \mathcal{Z}_q$.

The card reader (R) knows the public key, $y = g^x$.

Schnorr Identification Scheme

C chooses a random number $k \in \mathcal{Z}_q$

$C \to R: r = g^k$ ▷ commitment

R chooses a random challenge $e \in \mathcal{Z}_q$

$R \to C: e$ ▷ challenge

$C \to R: s = k + xe \bmod q$ ▷ response

R accepts the identification if $g^s = ry^e$

The function of the random value k is to blind the secret key x, so that it can be reused multiple times.

Schnorr Identification Scheme

Correctness and Security

The scheme is correct. In fact, for any x, r, e:

$$g^s = g^k (g^x)^e = r y^e$$

The scheme is secure against:

- An adversary impersonates a card C
- Eavesdropping

Part II

Elliptic Curve Cryptography

2. Elliptic Curve Cryptography

Elliptic Curves over $\mathbb{Z}_p$

When working over $\mathbb{Z}_p$, with p odd prime, an elliptic curve is defined as the solutions of the equation:

$$y^2 \equiv x^3 + bx + c \pmod{p}$$

with the addition of the point at infinity ∞.
We will consider nonsingular curves, i.e. with $4b^3 + 27c^2 \bmod p \neq 0$. The condition ensures that there are no repeated roots.
Each curve over $\mathbb{Z}_p$ has its own number of points N, which is difficult to find. Roughly, there are $p + 1$ points plus an error term, which depends on the specific curve.

Theorem (Hasse's Theorem)

$$|N - p - 1| < 2\sqrt{p}$$

Elliptic Curves over $\mathbb{Z}_p$

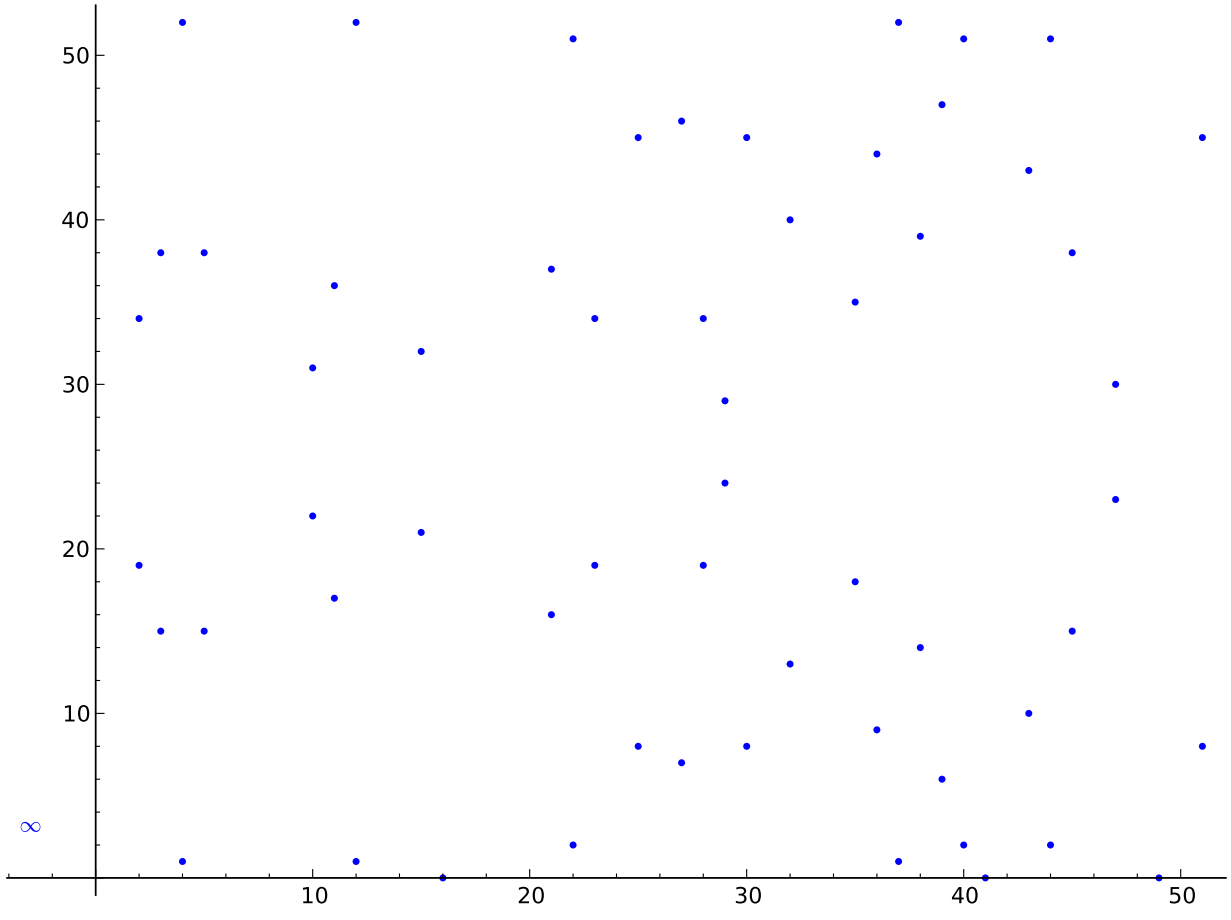

Figure: The curve $y^2 \equiv x^3 + 4x + 27 \pmod{53}$

Elliptic Curves over $\mathbb{Z}_p$

Addition law

The addition law is analogous to the real case.

Point Addition over Elliptic Curves

Given $y^2 \equiv x^3 + bx + c \pmod{p}$ and the point P_1 and P_2, to compute $P_3 = P_1 + P_2$ we have the following formulas:

$$m = \begin{cases} (y_2 - y_1)(x_2 - x_1)^{-1} \bmod p & \text{if } P_1 \neq P_2 \\ (3x_1^2 + b)(2y_1)^{-1} \bmod p & \text{if } P_1 = P_2 \end{cases}$$

$$x_3 = m^2 - x_1 - x_2 \bmod p$$

$$y_3 = m(x_1 - x_3) - y_1 \bmod p$$

Properties and Special Cases

Some properties

- $(E, +)$ is an Abelian group. Therefore $P_1 + P_2 = P_2 + P_1$.
- The identity element is the point at infinity. Therefore $P + \infty = P$
- If $P = (x, y)$, then $-P = (x, -y)$.

If m cannot be calculated, then we are in a special case and $P_3 = \infty$.

- If $P_1 = (x, y)$ and $P_2 = (x, -y)$ are symmetric w.r.t. the x-axis.
- If $P_1 = P_2 = (x, 0)$ lies on the x-axis

Elliptic Curves over $\mathbb{Z}_p$ as Finite Groups

- Elliptic Curves over $\mathbb{Z}_p$ form a finite, cyclic group with order N. The optimal case for cryptographic usage is when N is prime.
- Given a generator point A and a random point B, finding an integer $0 \leq k < N$ such that $B = kA$ is a DLP. It is estimated that finding the discrete log over a curve with $p \simeq 160$ bit has a complexity similar to finding the log over $\mathbb{Z}_q$ with $q \simeq 1880$ bit.
- Building a curve for cryptographic usage is a difficult task, mainly because there is no easy way to calculate N for a curve with random parameters. Further, several families of curves are subject to mathematical attacks. Therefore standarization bodies publish several sets of curves with various sizes.

2. Elliptic Curve Cryptography

- Elliptic Curves over $\mathbb{Z}_p$
- Applications to Cryptography

1. Exercises

Elliptic Curve Diffie-Hellman Key Exchange (ECDHKE)

ECDHKE Protocol

Common Input:

- a security parameter n
- a curve E, a generator G, its order $q \geq 2^n$
- a key derivation function $KDF(\cdot)$ that maps a point of the curve into $\{0,1\}^n$

Protocol:

1. Alice chooses $x \leftarrow \mathbb{Z}_q$ uniformly at random and computes $H_1 := xG$
2. Alice sends H_1 to Bob
3. Bob receives H_1. He chooses $y \leftarrow \mathbb{Z}_q$ uniformly at random and computes $H_2 := yG$. Bob sends H_2 to Alice and outputs the key $k_B := KDF(yH_1)$.
4. Alice receives H_2 and outputs the key $k_A := KDF(xH_2)$.

Comments on ECDHKE

No particular differences from standard DHKE.

The curve points are generally encoded in compressed form for transmission. For a given x there are at most two points, one having y odd and the other having y even. Therefore it is sufficient to send the x-coordinate and the y-coordinate mod 2. The receiver can recover the full point by using the curve's equation.

EC Integrated Encryption Scheme (ECIES)

Setup

ECIES is the most used hybrid encryption scheme over elliptic curves. It is used by Alice for sending messages to Bob.

ECIES Setup

Bob chooses

- a key derivation function KDF, a MAC scheme, a symmetric encryption scheme Enc
- a curve, a generator G
- his private key x and public key $K_B = xG$

EC Integrated Encryption Scheme (ECIES)

Encryption

Encryption

To encrypt a message m, Alice does the following

1. generate a random nonce $r \leftarrow \mathbb{Z}_q^*$ with $q = \text{ord}(G)$
2. calculate $R = rG$ and $P = (x_P, y_P) = rK_B$
3. derive a shared secret $s = x_P$ and compute the MAC and symmetric encryption keys $k_E \| k_M = KDF(s)$
4. encrypt the message $c = \text{Enc}_{k_E}(m)$
5. compute the MAC tag $t = MAC_{k_M}(c)$
6. send $R \| c \| t$

Decryption exploits the relation $P = xR = xrG = rK_B$.

ECDSS/ECDSA

Setup

The analogous of DSS/DSA over EC.

ECIES Setup

Bob chooses

- a hash function $H(\cdot)$.
- a curve, a point G with order q (G may not be a generator)
- his private key x and public key $K_B = xG$

ECDSS/ECDSA

Signature

ECDSS Signature (Sign)

On input a message $m \in 0,1^*$

1. Choose a security nonce $k \leftarrow \mathbb{Z}_q^*$ and set

$$R := kG$$
$$s := (H(m) + xx_R)k^{-1} \bmod q$$

2. Output (R, s)

ECDSS/ECDSA Signature Verification

ECDSS Signature (Vrfy)

On input a message m and a signature (R, s)

1. Compute

$$u_1 := H(m)s^{-1} \bmod q$$
$$u_2 := x_R s^{-1} \bmod q$$

2. Output 1 if

$$R = u_1 G + u_2 K_B$$

2. Elliptic Curve Cryptography

Diffie-Hellman Key Exchange

Alice e Bob exchange a session key using the Diffie-Hellman protocol. They publish an elliptic curve
$E : y^2 = x^3 + 18x + 2 \bmod 29$. This curve has $n = 27$ points.
They also publish $P = (4, 14)$.
Alice sends the message $A = aP = (7, 6)$ and receives the message $B = bP = (9, 9)$.

1. Verify that P is a generator of the curve.
2. Compute b using the Pohlig-Hellman algorithm.
3. Compute the session key.

Diffie-Hellman Key Exchange

Solution

Verify that P is a generator of the curve.

The number of points $n = 27 = 3^3$ is not prime. There are elements with order 3, 9 or 27. A generator is a point with order 27. Consider the following table:

$\text{ord}(P)$	$3P$	$9P$	$27P$
3	∞	∞	∞
9	$\neq \infty$	∞	∞
27	$\neq \infty$	$\neq \infty$	∞

The only way to verify that the order of P is 27 is verifying that the order is not 3 or 9. If $9P = \infty$, then the order must be 27. Since:

$$9P = 9(4, 14) = 2^3P + P = (10, 14)$$

then P is a generator.

Diffie-Hellman Key Exchange

Solution

Compute b using the Pohlig-Hellman algorithm.

Let call$x$ the unknown b. We must solve the equation:

$$(9,9) = x(4,14)$$

The Pohlig-Hellman algorithm allows us to easily compute:

$$x = x_0 + 3x_1 + 9x_2 \pmod{27}$$

where $0 \leq x_0, x_1, x_2 \leq 2$.

Diffie-Hellman Key Exchange

Solution

The term x_0 is solution of the equation:

$$\begin{aligned}
\frac{27}{3}(9,9) &= \frac{27}{3}x_0(4,14) \\
9(9,9) &= 9x_0(4,14) \\
(10,14) &= x_0(10,14) \\
x_0 &= 1
\end{aligned}$$

Diffie-Hellman Key Exchange

Solution

The term x_1 is solution of the equation:

$$\begin{aligned}
\frac{27}{9}\big((9,9) - x_0(4,14)\big) &= \frac{27}{3}x_1(4,14) \\
3\big((9,9) - (4,14)\big) &= 9x_1(4,14) \\
3(7,6) &= 9x_1(4,14) \\
(10,14) &= x_1(10,14) \\
x_1 &= 1
\end{aligned}$$

Diffie-Hellman Key Exchange

Solution

The term x_2 is solution of the equation:

$$\begin{aligned}
\frac{27}{27}\big((9,9) - (x_0 + 3x_1)(4,14)\big) &= \frac{27}{3}x_2(4,14) \\
(7,6) - 3(4,14) &= 9x_2(4,14) \\
\infty &= 9x_2(4,14) \\
\infty &= x_2(10,14) \\
x_2 &= 0
\end{aligned}$$

Therefore $b = x = x_0 + 3x_1 + 9x_2 = 1 + 3 + 0 = 4$.

Diffie-Hellman Key Exchange

Solution

Compute the session key.

The session key is the point $K = bA = 4(7,6) = (3,24)$.

DSA Signature

Alice signs using DSA scheme. She publishes the curve $E : y^2 = x^3 + 2x + 2 \bmod 13$ and a point $A = (2, 1)$ with order $q = 5$. She chooses a secret a, computes $B = aA = (2, -1)$ and publishes B.
Alice signs the message $m = 4$, chooses a random k and computes:

$$R = kA = (6, -3) = (x_R, y_R)$$
$$s = k^{-1}(m + ax_R) = 4 \pmod 5$$

Alice publishes the signed message: $(m, R, s) = (4, (6, -3), 4)$.
Successively Alice signs the message $m = 2$ and obtains: $(m, R, s) = (2, (6, -3), 3)$.
Questions:

1. Verify that the order of A is q.
2. Verify Alice's signature.
3. Taking advantage of Alice's mistake, compute a.

DSA

Solution

Verify that the order of A is q.

Since $q = 5$ is prime, the order of A is 5 if and only if $5A = \infty$.
We compute:

$$5A = 2^2(2,1) + (2,1) = 2(6,-3) + (2,1) = (2,12) + (2,1) = \infty$$

The order of A is 5.

DSA

Solution

Verify Alice's signature.

$$
\begin{aligned}
u_1 &= s^{-1}m = 4 \cdot 4 = 1 \pmod 5 \\
u_2 &= s^{-1}x_R = 4 \cdot 6 = 4 \pmod 5 \\
V &= u_1 A + u_2 B = A + 4B = (2,1) + 4(2,-1) = \\
&= (2,1) + (2,1) = (6,-3) = R
\end{aligned}
$$

The signature is valid.

DSA

Solution

Compute the secret key a

Alice used the same k twice,so we can write the following equation:

$$\begin{aligned} s_1k - m_1 &= ax_R = s_2k - m2 \pmod q \\ (s_1 - s_2)k &= m_1 - m_2 \pmod q \\ (4-3)k &= 4-2 \pmod 5 \\ k &= 2 \end{aligned}$$

Now we substitute the value of k in the equation $sk - m = ax_R$ and obtain:

$$\begin{aligned} a &= x_R^{-1}(sk - m) \pmod q \\ a &= 6^{-1}(4 \cdot 2 - 4) = 4 \pmod 5 \end{aligned}$$