Identification and Zero Knowledge

Ahto Buldas   Aleksandr Lenin

Nov 25, 2019

Introduction

How should Alice prove to Bob that she is who she claims to be?

Requirements:

- Honest parties (like Alice) are be able to prove their identities correctly
- Dishonest third party (Carol) should not be able to impersonate Alice.

Bob can also be dishonest and, after getting the information from Alice, impersonate her to someone else.

Example

Bob is a computer on which Alice has an account.

Why not to use passwords?

There are problems:

1. Alice must transmit her password secretly to prevent stealing it by eavesdropping
2. Alice must have previously set up the password for identification purposes
3. As Bob also has Alice's password, he can impersonate Alice

Challenge-Response

Suppose we have a public key cryptosystem.

Challenge-response protocol using encryption:

- *Challenge*: Bob picks a random message r and sends Alice a ciphertext $\mathsf{E}_A(r)$.
- *Response*: Alice decrypts the message and responds with r.

Challenge-response protocol using signatures:

- *Challenge*: Bob picks a random message r and sends r to Alice
- *Response*: Alice signs the message and sends the signature $\mathsf{S}_A(r)$ to Bob

Challenge-Response: Concerns

If Alice has to frequently identify herself to Bob

May Bob send specially crafted messages that finally help him to impersonate Alice?

Alice would like to reveal no information, except that she is Alice.

Interactive Proof

Alice knows a proof that two graphs G_0 and G_1 with the same vertext set V not isomorphic.

Alice wants to convince Bob that she knows a proof, without revealing the proof.

Alice and Bob execute the next protocol k-times:

1. Bob randomly picks $b \leftarrow \{0,1\}$ and a permutation π of V, and sends Alice the graph $H = \pi(G_b)$.
2. Alice finds b' such that $H \cong G_{b'}$ and sends b' to Bob.

Bob accepts, if $b = b'$ in all k rounds.

If Alice has a proof then she will always find the right graph.

If Alice cannot distinguish the graphs, she succeeds with probability $\frac{1}{2^k}$.

Zero-Knowledge Proof

Completeness: If the claim is true, Bob always accepts.

Soundness: If the claim is false, Bob should only accept with very small probability.

Zero-knowledge: There is an efficient simulator M that is able to generate the transcript of the protocol (with the same probability distribution as in the protocol) knowing only the truth value of the claim.

Hence, Bob only learns the truth value of the claim, because if he knows the truth value, he can simulate the communication himself.

Example

Alice knows an isomorphism $\psi\colon G_0 \to G_1$ between graphs G_0 and G_1.

Alice wants to convince Bob that she knows such an isomorphism.

Theprotocol has t rounds. In each round:

1. Alice picks a random permutation π and sends Bob $H = \pi(G_0)$.
2. Bob chooses a random $b \leftarrow \{0, 1\}$ and sends b to Alice.
3. Alice computes a permutation σ so that $G_b = \sigma(H)$ and sends σ to Bob. She takes $\sigma = \pi^{-1}$ if $b = 0$ and $\sigma = \psi\pi^{-1}$ if $b = 1$.
4. Bob checks that $G_b = \sigma(H)$.

Completeness: If Alice's claim is true, Bob always accepts.

Soundness: If $G_0 \not\cong G_1$, Bob accepts with probability $\frac{1}{2^k}$.

Zero-knowledge: The simulator M works as follows:

1. Chooses random $b' \leftarrow \{0,1\}$ and a random permutation π
2. Computes $H' \leftarrow \pi(G_{b'})$
3. Computes $\sigma' \leftarrow \pi^{-1}$

Always $G_{b'} = \sigma'(H')$.

In case Alice knows an isomorphism $\psi \colon G_0 \to G_1$, the transcripts (H, b, σ) and (H', b', σ') have identical probability distributions.

Therefore, Bob learns nothing about ψ.

Homework exercise: Prove that (H, b, σ) and (H', b', σ') have identical probability distributions.

Fiat-Shamir Identification Scheme

Fiat, Shamir (1986) in the context of account holder and cash machine interaction. U.S. Army tried classify the scheme as top secret.

Alice has RSA modulus $n = pq$ and a value $y \in \mathbb{Z}_n$.

Alice proves to Bob that she knows x such that $y = x^2 \bmod n$

The following protocol is executed k times:

1. Alice picks $r \leftarrow \mathbb{Z}_n^*$, computes $a \leftarrow r^2 \bmod n$ and sends a to Bob
2. Bob picks a random $b \leftarrow \{0, 1\}$ and sends b to Alice.
3. Alice computes $z = x^b r \bmod n$ and sends z to Bob.

Bob accepts if $z^2 \equiv y^b a \pmod{n}$ in all k rounds.

Completeness: $z^2 \equiv (x^b r)^2 \equiv x^{2b} r^2 \equiv y^b a \pmod{n}$

Soundness: If y is not a square $\bmod n$, then Alice cannot answer correctly for in both cases $b = 0$ and $b = 1$. Indeed, let z_0 and z_1 be Alice's responses in these cases, respectively.

We have that $z_0^2 \equiv a \pmod{n}$ and $z_1^2 \equiv ya \pmod{n}$. Hence, $z_1^2 \equiv y z_0^2 \pmod{n}$ and hence y must be a square $\bmod n$, a contradiction.

Zero-knowledge: The simulator M:

1. Chooses random $b \leftarrow \{0, 1\}$ and a random $r \leftarrow \mathbb{Z}_n^*$
2. Computes $a' \leftarrow r^2 y^{-b} \bmod n$ and $z' \leftarrow r$

As $z'^2 \equiv r^2 \equiv y^b r^2 y^{-b} \equiv y^b a' \pmod{n}$ and the transcripts (a, b, z) and (a', b, z') have equal probability distributions, the protocol is ZK.

Homework exercise: Prove that (a, b, z) and (a', b, z') have identical probability distributions.

ZK vs HVZK: Honest Verifier Zero Knowledge

Bob can also be malicious and choose b arbitrarily

For perfect zero knowledge, the simulator M has to simulate Bob

M has to choose $a' \leftarrow r^2 y^{-b} \bmod n$ before bob-s challenge b'

If $b' \neq b$, then M simply executes Bob again until $b' = b$

Elliptic Curves

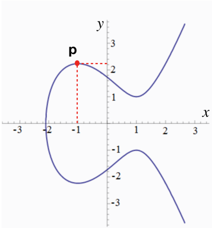
Elliptic curve is a set of points $\mathbf{p} = (x, y)$, where x and y are elements of a finite field $\mathbf{F}$ (of prime order $p > 2$) satisfying the equation:

$$y^2 = x^3 + ax + b$$

along with a point at infinity, denoted by ∞.

The NIST curve P-256 uses $p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$ and

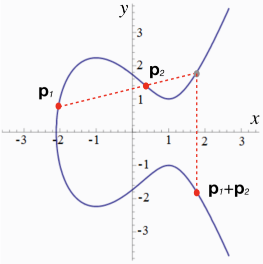
$$y^2 = x^3 - 3x + \text{41058363725152142129326129780047268409114441015993725554835256314039467401291} \quad .$$

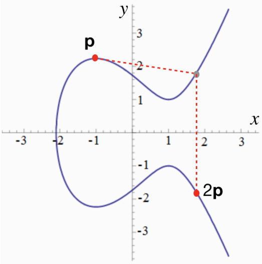


Operations with Points

There is a binary operation $+$ on the points of a curve:

- To compute $\mathbf{p}_1 + \mathbf{p}_2$, draw a line from $\mathbf{p}_1$ to $\mathbf{p}_2$, intersect it with the curve and reflect in the x-axis.
- To compute $2\mathbf{p} = \mathbf{p} + \mathbf{p}$, take the tangent at that point, intersect it with the curve and then reflect in the x-axis.





Properties of $+$

The set of points is an *Abelian group* under $+$: for any points $\mathbf{a}, \mathbf{b}, \mathbf{c}$:

- *associativity*: $\mathbf{a} + (\mathbf{b} + \mathbf{c}) = (\mathbf{a} + \mathbf{b}) + \mathbf{c}$
- *commutativity*: $\mathbf{a} + \mathbf{b} = \mathbf{b} + \mathbf{a}$
- *zero*: $\mathbf{a} + \infty = \mathbf{a}$
- *inverses*: there is a point $-\mathbf{a}$ such that $\mathbf{a} + (-\mathbf{a}) = \infty$

For every $\mathbf{a} \in \mathcal{G}$ and positive integer z we define the *scalar multiple*

$$z\mathbf{a} = \overbrace{\mathbf{a} + \mathbf{a} + \ldots + \mathbf{a}}^{z}$$

Scalar multiplication is *bilinear*:

$$(z + z')\mathbf{a} = z\mathbf{a} + z'\mathbf{a} \qquad \text{and} \qquad z(\mathbf{a} + \mathbf{a}') = z\mathbf{a} + z\mathbf{a}'$$

Edwards Curves

Twisted Edwards curve is a set of points $\mathbf{p} = (x, y)$, where x and y are elements of a finite field $\mathbf{F}$ (of order p^k where $p > 2$ is prime) satisfying the equation:

$$ax^2 + y^2 = 1 + dx^2y^2 \quad .$$

Twisted Edwards curves are equivalent to a certain class of elliptic curves. They have universal addition also applicable to point doubling:

$$(x_1, y_1) \boldsymbol{+} (x_2, y_2) = \left(\frac{x_1y_2 + x_2y_1}{1 + dx_1x_2y_1y_2}, \frac{y_1y_2 - ax_1x_2}{1 - dx_1x_2y_1y_2} \right)$$

If $a = 1$, the curve is called an *Edwards curve*.

The twisted Edwards curve Edwards25519 uses $k = 1$, $p = 2^{255} - 19$, $a = -1$, and

$d =$ 3709570593466943934313808350875456518954211387984321901638878553 30 85940283555

Schnorr Identification Scheme

Use public and constant point $\mathbf{g}$ of an elliptic curve with order q

Secret key is a random number x

Public key is the point $\mathbf{x} = x\mathbf{g}$

Protocol: Alice proves that she knows x such that $\mathbf{x} = x\mathbf{g}$:

1. Alice picks a random number $r \leftarrow \mathbb{Z}_q^*$, computes $\mathbf{r} \leftarrow r\mathbf{g}$ and sends $\mathbf{r}$ to Bob
2. Bob generates a random number $\rho \leftarrow \mathbb{Z}_q^*$ and sends ρ to Alice
3. Alice computes $s \leftarrow (r - \rho x) \bmod q$ and sends s to Bob.

Bob accepts if $\mathbf{r} = s\mathbf{g} + \rho\mathbf{x}$.

Schnorr Identification Scheme and Fiat-Shamir Heuristics

Replacing random challenge ρ with a hash $H(\mathbf{r})$

1. Alice picks a random number $r \leftarrow \mathbb{Z}_q^*$, computes $\mathbf{r} \leftarrow r\mathbf{g}$
2. Alice computes $\rho \leftarrow H(\mathbf{r})$ and $s \leftarrow (r - \rho x) \bmod q$, and sends $(\mathbf{r}, \rho)$ to Bob

Bob:

1. Computes $\mathbf{r} \leftarrow s\mathbf{g} + \rho\mathbf{x}$
2. Accepts if $H(\mathbf{r}) = \rho$

Schnorr Signature

Use public and constant point $\mathbf{g}$ of an elliptic curve with order q

Gen: Secret key is a random number x

Pub: Public key is the point $\mathbf{x} = x\mathbf{g}$

Sig: The signature of a hashed message m is a pair (s, ρ) with

$$s = (r - \rho x) \bmod q \ \ ,$$

where $r \leftarrow \mathbb{Z}_q$ is a random number and $\rho = H(m||r\mathbf{g})$

Ver: A signature (s, ρ) on m is verified by

1. Computing $\mathbf{r} \leftarrow s\mathbf{g} + \rho\mathbf{x}$
2. Checking that $H(m||\mathbf{r}) = \rho$

ECDSA

Use public point $\mathbf{g}$

Gen: Secret key is a random number x

Pub: Public key is the point $\mathbf{x} = x\mathbf{g}$

Sig: The signature of a hashed message m is a pair (s, ρ), where ρ is the x-coordinate of the point $r\mathbf{g}$, r is a random number, and

$$s = \frac{m + \rho x}{r} \bmod q$$

Ver: A signature (s, ρ) on m is verified by checking that ρ is the x-coordinate of the point:

$$\frac{m}{s}\mathbf{g} + \frac{\rho}{s}\mathbf{x}$$