Discrete-Log Based Signatures

Overview of DL-based Signatures

- Discrete-Log-based signatures can be implemented using Elliptic Curves.
  - They are therefore more efficient than RSA-based signatures (signatures are far smaller).
- DL-based are preferred in Bitcoin
- Bitcoin currently uses ECDSA = Elliptic Curve Digital Signature Algorithm
- We will be learning about Schnorr signatures.
- Similar to ECDSA but have some better properties.
- Many proponents of switching Bitcoin signatures to Schnorr signatures.

Outline

- We will first construct an **Identification Scheme**
    - A way to prove knowledge of a secret key corresponding to a public key without revealing the secret key
    - Provides a form of "zero knowledge"
    - E.g. public key = g^x, secret key = x.
    - Prove that I know x, without revealing what x is
    - If I reveal x, someone can impersonate me next time.
- Use the **Fiat-Shamir transform** to convert an **Identification Scheme** into a **Signature Scheme**.

Identification Schemes

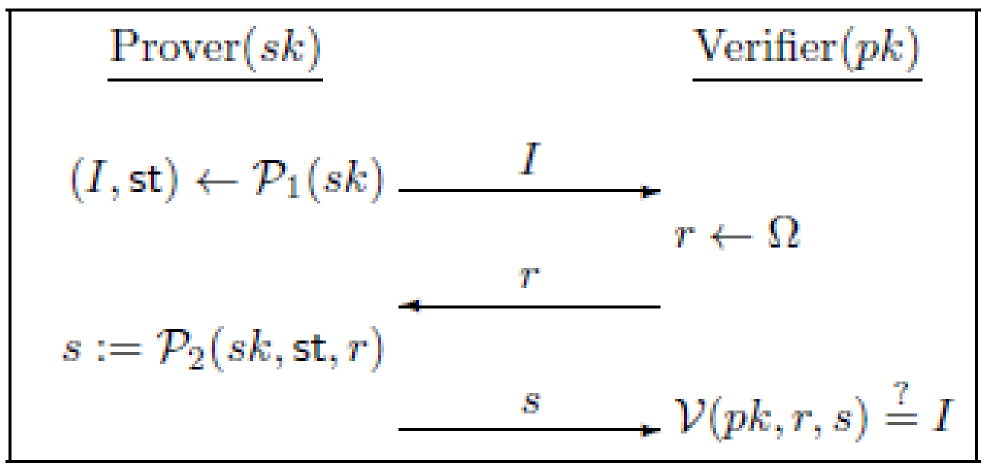

FIGURE 12.1: A 3-round identification scheme.

Identification Schemes

The identification experiment $\mathsf{Ident}_{\mathcal{A},\Pi}(n)$:

1. $\mathsf{Gen}(1^n)$ *is run to obtain keys* (pk, sk).
2. *Adversary* $\mathcal{A}$ *is given* pk *and access to an oracle* $\mathsf{Trans}_{sk}(\cdot)$ *that it can query as often as it likes.*
3. *At any point during the experiment,* $\mathcal{A}$ *outputs a message* I. *A uniform challenge* $r \in \Omega_{pk}$ *is chosen and given to* $\mathcal{A}$, *who responds with* s. *(We allow* $\mathcal{A}$ *to continue querying* $\mathsf{Trans}_{sk}(\cdot)$ *even after receiving c.)*
4. *The experiment evaluates to 1 if and only if* $\mathcal{V}(pk, r, s) \stackrel{?}{=} I$.

DEFINITION 12.8 *Identification scheme* $\Pi = (\mathsf{Gen}, \mathcal{P}_1, \mathcal{P}_2, \mathcal{V})$ *is* secure against a passive attack, *or just* secure, *if for all probabilistic polynomial-time adversaries* $\mathcal{A}$, *there is a negligible function* negl *such that:*

$$\Pr[\mathsf{Ident}_{\mathcal{A},\Pi}(n) = 1] \leq \mathsf{negl}(n).$$

The Schnorr Identification Scheme

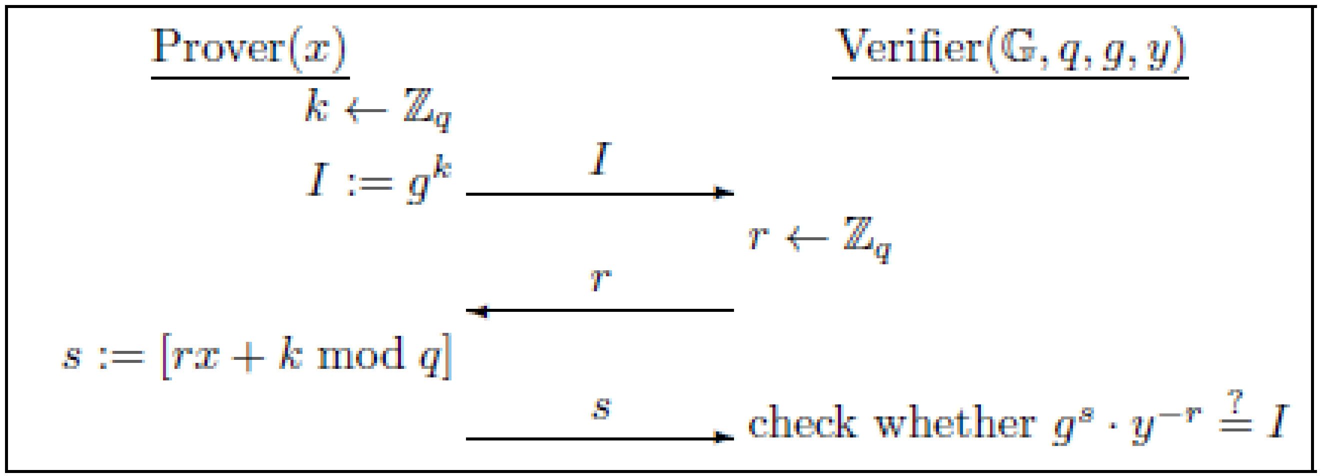

FIGURE 12.2: An execution of the Schnorr identification scheme.

Security Analysis

Theorem: If the Dlog problem is hard relative to G then the Schnorr identification scheme is secure.

Security Analysis

Idea of proof:

- Oracle can generate correctly distributed transcripts without knowing x.
  - How?

Security Analysis

Idea of proof:

- Given an attacker A who successfully responds to challenges with non-negligible probability, can construct an attacker A' who extracts the discrete log x of y by **rewinding**.

From Identification Schemes to Signatures: The Fiat-Shamir Transform

CONSTRUCTION 12.9

Let $(\mathsf{Gen}, \mathcal{P}_1, \mathcal{P}_2, \mathcal{V})$ be an identification scheme, and construct a signature scheme as follows:

- Gen: on input 1^n, simply run $\mathsf{Gen}(1^n)$ to obtain keys pk, sk.

  The public key pk specifies a set of challenges Ω_{pk}. As part of key generation, a function $H : \{0,1\}^* \to \Omega_{pk}$ is specified, but we leave this implicit.

- Sign: on input a private key sk and a message $m \in \{0,1\}^*$, do:

  1. Compute $(I, \mathsf{st}) \leftarrow \mathcal{P}_1(sk)$.
  2. Compute $r := H(I, m)$.
  3. Compute $s := \mathcal{P}_2(sk, \mathsf{st}, c)$

  Output the signature (r, s).

- Vrfy: on input a public key pk, a message m, and a signature (r, s), compute $I := \mathcal{V}(pk, r, s)$ and output 1 if and only if $H(I, m) \stackrel{?}{=} r$.

The Fiat-Shamir transform.

Security Analysis

Theorem: Let Π be an identification scheme, and let Π' be the signature scheme that results by applying the Fiat-Shamir transform to it. If Π is secure and H is modeled as a random oracle, then Π' is secure.

The Schnorr Signature Scheme

CONSTRUCTION 12.12

Let $\mathcal{G}$ be as described in the text.

- Gen: run $\mathcal{G}(1^n)$ to obtain $(\mathbb{G}, q, g)$. Choose uniform $x \in \mathbb{Z}_q$ and set $y := g^x$. The private key is x and the public key is $(\mathbb{G}, q, g, y)$. As part of key generation, a function $H : \{0,1\}^* \to \mathbb{Z}_q$ is specified, but we leave this implicit.
- Sign: on input a private key x and a message $m \in \{0,1\}^*$, choose uniform $k \in \mathbb{Z}_q$ and set $I := g^k$. Then compute $r := H(I, m)$, followed by $s := [rx + K \bmod q]$. Output the signature (r, s).
- Vrfy: on input a public key $(\mathbb{G}, q, g, y)$, a message m, and a signature (r, s), compute $I := g^s \cdot y^{-r}$ and output 1 if $H(I, m) \stackrel{?}{=} r$.

The Schnorr signature scheme.