CS 7880 Graduate Cryptography November 8, 2017

Lecture 18: Identification Schemes, Schnorr Signatures

Lecturer: Daniel Wichs *Scribe: Vikrant Singhal*

1 Topic Covered

- Identification Schemes
- Schnorr Identification Scheme
- Schnorr Signatures

2 Identification Schemes

Identification schemes help to convince "verifiers" about the identity of someone without letting others (read adversaries) convince the "verifiers" about the identity of that particular person. For example, if I were to log in to many websites using the same password, then a secure identification scheme should prevent other websites from logging in to a particular website using my identity. We consider the following aspects of an identification scheme.

2.1 Structure

Let $(pk, sk) \leftarrow \mathsf{Gen}(1^n)$, where n is a security parameter, pk is the public key, and sk is the secret key. The protocol comprises a prover, P, and a verifier, V.

$$\begin{array}{ccc} P(pk, sk) & & V(pk) \\ & \xleftarrow{\qquad} & \\ & \xrightarrow{\qquad} & \\ & \xleftarrow{\qquad} & \\ & & \downarrow 0/1 \end{array}$$

The protocol is randomised, and V outputs either 0 or 1.

2.2 Correctness

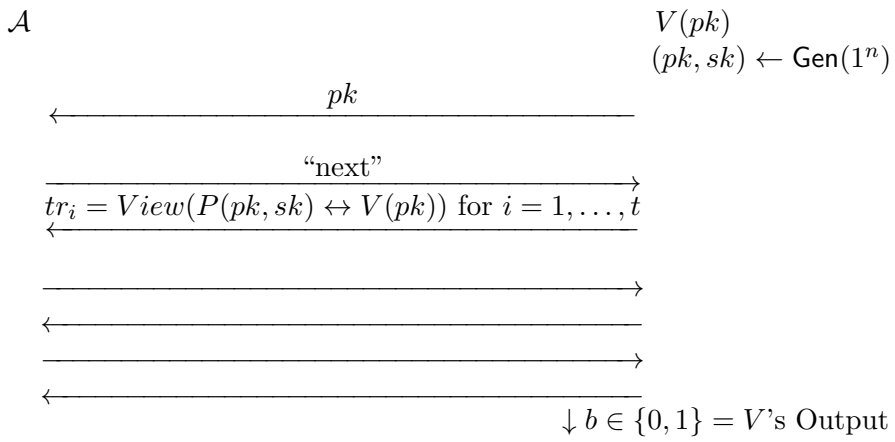
We require the following notion of correctness.

$$\mathbb{P}\left[Output(P(pk, sk) \leftrightarrow V(pk)) = 1\right] = 1$$

The verifier should accept if we run the protocol correctly.

2.3 Honest Verifier Security

Let $View(P(pk, sk) \leftrightarrow V(pk))$ denote everything that the verifier sees when the protocol is running. It is also referred to as the transcript of the protocol from the perspective of the verifier. Let $IDGame_{\mathcal{A}}(n)$ be the following security game.

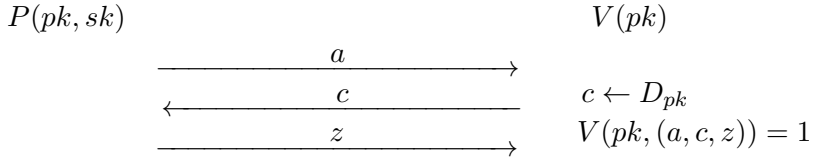


Here, $t = \mathsf{poly}(n)$. We say that the scheme is secure if,

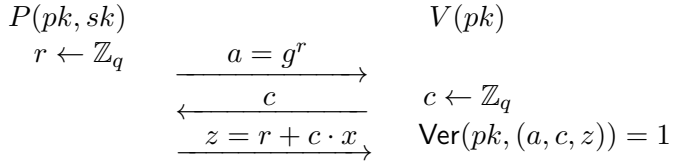
$$\mathbb{P}\left[IDGame_{\mathcal{A}}(n) = 1\right] = \mathsf{negl}(n).$$

3 Schnorr Identification Scheme

If we have a digital signature scheme, then it is simple to construct an identification scheme from it. The verifier, V, picks a random string and asks the prover, P, to sign it, and when that signature is transmitted to V, he can verify it. But here, we build an identification scheme from Discrete Log, which would have the following form.



Let $(G, g, q) \leftarrow \mathsf{Gen}(1^n)$, where G is a group with prime order, q, and g is the generator of G. Then we have the following scheme.



$\mathsf{Ver}(pk, (a, c, z))$: If $a \cdot h^c = g^z$, output 1. Else, output 0.

Claim 1 *$\exists$ PPT, Sim, with $(a, c, z) \leftarrow Sim(pk)$, such that $\forall (pk, sk) \leftarrow Gen(1^n)$,*

$$View(P(pk, sk) \leftrightarrow V(pk)) \equiv Sim(pk).$$

So, Sim is a simulator, who simulates the transcripts himself without knowing the secret key. This is good for the security of the system because it implies that the adversary could not have done much, even on viewing multiple transcripts, because he could have generated the transcripts himself. So, he could not have learned anything more than the public key.

Proof: If Sim samples $c, z \leftarrow \mathbb{Z}_q$, and sets $a = \frac{g^z}{h^c}$, then the distribution of a given c, z is exactly the same as in $View$. In other words, the distribution of (a, c, z) is the same in both cases. The simulator basically generates the transcripts by running the protocol in a different order. □

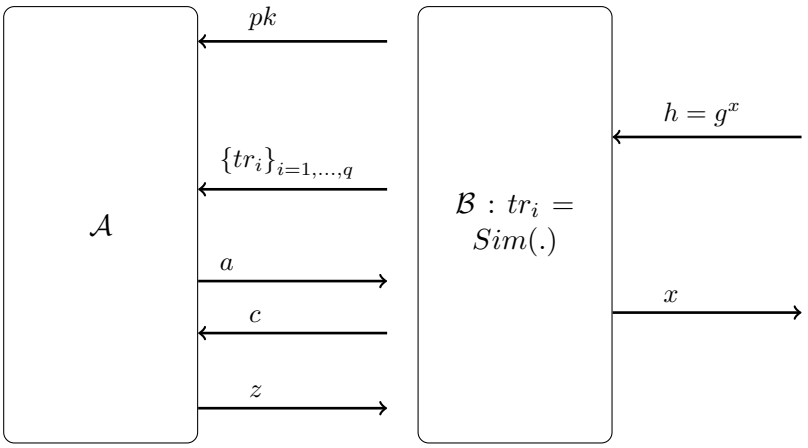
Claim 2 *$\exists$ PPT, Ext with $x' \leftarrow Ext(pk, a, (c, z), (c', z'))$, such that whenever $c \neq c'$ and $Ver(pk, (a, c, z)) = Ver(pk, (a, c', z')) = 1$, then $x' = x$.*

Proof: $Ver(pk, (a, c, z)) = Ver(pk, (a, c', z')) = 1$
$\iff a.h^c = g^z$
$\iff a.h^{c'} = g^{z'}$
$\Rightarrow h^{c-c'} = g^{z-z'}$
$\Rightarrow h = g^{\frac{z-z'}{c-c'}}$ $\quad [c - c' \neq 0$, and q is prime]
Ext outputs $x' = \frac{z-z'}{c-c'}$. □

The aforementioned claims are somewhat contradictory. The first claim says that an adversary is able to create as many transcripts as he wants without knowing the secret key. On the other hand, the second claim says that if the adversary can verify two different transcripts with the same a, then he can break the system as he can recover the secret key. But in the first claim, any given value of a is never repeated in any sampled view (with high probability) because the choices of c and z are random each time.

Theorem 1 *The above construction is a secure identification scheme under Discrete Log.*

Proof: Assume $\exists$ PPT, $\mathcal{A}$, such that $\mathbb{P}[IDGame_{\mathcal{A}}(n) = 1] = \varepsilon(n) \neq \mathsf{negl}(n)$. We construct a reduction, $\mathcal{B}$, that solves Discrete Logarithm.



Here, a technique called, "rewinding," is used: $\mathcal{B}$ takes a snapshot upto the point it sees a. Then it rewinds and sends a fresh c' to the adversary, and hopes that $Ver(pk, (a, c', z')) = 1$.

Let w be the total state of $\mathcal{A}$ after it sends a. Let $p_w = \Pr[W = w]$, where W is the random variable denoting the transcript up to the point, where $\mathcal{B}$ takes the snapshot, and let $\varepsilon_w = \mathbb{P}[IDGame_{\mathcal{A}}(n) = 1 | W = w]$. Then,

$$\mathbb{P}[\mathcal{A} \text{ wins}] = \sum_w p_w \cdot \varepsilon_w = \varepsilon(n).$$

Let $\delta_w = \mathbb{P}[\mathcal{B} \text{ wins} | W = w]$. $\mathcal{B}$ wins when $c \neq c'$ and when $\mathcal{A}$ wins on both occasions. So,

$$\delta_w = \varepsilon_w^2 \left(1 - \frac{1}{q}\right) \geq \varepsilon_w^2 - \frac{1}{q}.$$

We get the following.

$$\begin{aligned}
\mathbb{P}[\mathcal{B} \text{ wins}] &= \sum_w p_w \delta_w \\
&\geq \sum_w p_w \cdot {\varepsilon_w}^2 - \frac{1}{q} \qquad \left[\because \mathbb{P}\left[c = c'\right] = \frac{1}{q}\right] \\
&= \mathbb{E}\left[{\varepsilon_w}^2\right] - \frac{1}{q} \\
&\geq \mathbb{E}\left[\varepsilon_w\right]^2 - \frac{1}{q} \qquad [\because \text{By Jensen's Inequality}] \\
&= \varepsilon^2(n) - \frac{1}{q} \neq negl(n) \qquad [\because \mathsf{nonnegl}(n) - \mathsf{negl}(n) = \mathsf{nonnegl}(n)]
\end{aligned}$$

So, when $\mathcal{A}$ breaks the identification scheme, we are able to solve the Discrete Log problem with non-negligible probability, thus, contradicting our assumption. Therefore, the construction is secure under Discrete Log. □

4 Schnorr Signatures

We cannot use the identification scheme directly to construct the signature scheme because the identification scheme is interactive. We want the signature to be performed in one shot. Instead, we pretend that the signee (the prover in the identification scheme) has an "imaginary friend", who uses the Random Oracle (RO) to return a value to him. He finally outputs another value based on the value he recieved, which is then his signature. Also, Gen for this scheme is the same as Gen for the identification scheme. So, we have $(pk, sk) \leftarrow \mathsf{Gen}(1^n)$, where $sk = x$ and $pk = h = g^x$.

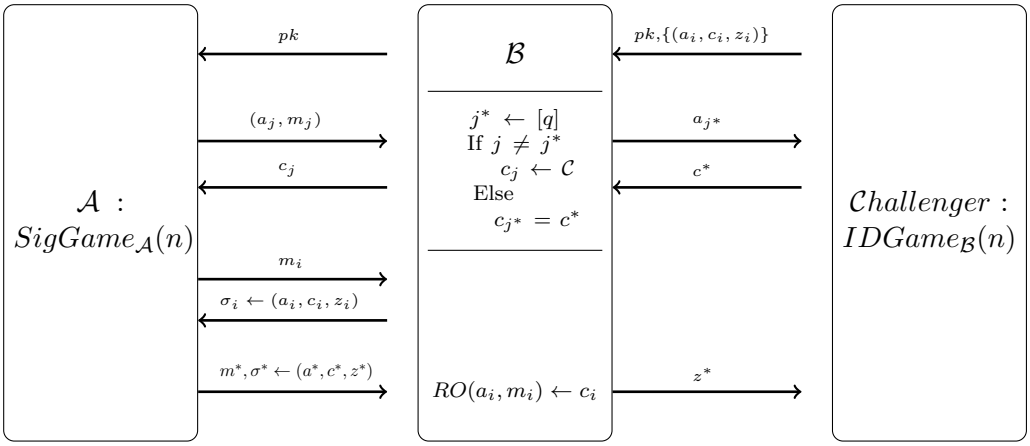
$\mathsf{Sign}_{sk}(m)$ "Imaginary Friend"

$$\begin{aligned}
&\xrightarrow{a = g^r (r \leftarrow \mathbb{Z}_q)} \\
&\xleftarrow{c \leftarrow RO(a, m)} \\
&\xrightarrow{z = r + c \cdot x}
\end{aligned}$$

Output: $\sigma \leftarrow (a, c, z)$

$\mathsf{Verify}_{pk}(m, \sigma = (a, c, z))$: If $c = RO(a, m)$ and $g^z = ah^c$, output 1. Else, output 0.

Note that we don't really need c in σ as RO is known to public, but we keep it anyway for simplicity. We prove the security of this scheme under RO model, and for that, we assume that there is an adversary, $\mathcal{A}$, that breaks its security. Then we would have a reduction, $\mathcal{B}$, that would break the security of the identification scheme. Assume that $\mathcal{B}$ makes q Random Oracle queries.



We want each a_i to have very high entropy. Some meaningful class of secure ID schemes has this property. Schnorr also does because the choice of a is random since r is random.

In the reduction, RO is programmed, such that $RO(a_i, m_i) = c_i$ for every signature query, m_i. $\mathcal{A}$ wants random c_i, which is what it gets because $\mathcal{B}$ gets $\{(a_i, c_i, z_i)\}$ at the beginning.