Key-exchange protocol:

Defn Let Π be a key-exchange protocol between two parties A and B. Let r_A and r_B be the random coins of A and B. We denote $OUT_A^{\Pi}(r_A, r_B, 1^n)$ and $OUT_B^{\Pi}(r_A, r_B, 1^n)$ be the output of A and B respectively after the execution of the protocol. We call Π is correct if

* $Trans_{\Pi}(r_A, r_B, 1^n)$:= it outputs the seq. of msgs exchanged between A and B during the execution of the protocol.

$$\Pr\left[OUT_A^{\pi}(r_A, r_B, 1^n) \neq OUT_B^{\pi}(r_A, r_B, 1^n)\right] \leq negl(n).$$

Security of a key-exchange protocol: $-KE^{eav}_{A,\pi}(n)$

w.r.t a passive adversary (Eavesdropper).

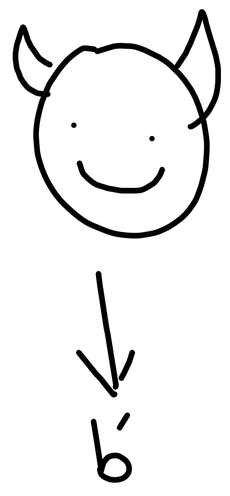
Chall

(i) Sample r_A, r_B, executes π on $(r_A, r_B, 1^n)$ if generates $Trans_\pi$

1^n →

(ii) $b \leftarrow \$ \{0,1\}$, if $b=0$ $k \leftarrow OUT_A(r_A, r_B, 1^n)$ if $b=1$, then $k \leftarrow \$ \mathcal{G}$

$(Trans_\pi, k)$ →



b'

Adversary A wins the game if $b' = b$

$KE^{eav}_{A,\pi}(n) = 1$ if $b' = b$.

Def: A key exchange protocol is secure in the presence of an eavesdropper if for every PPT adversary A, $\exists$ a negligible fn negl(), such that

$$P[KE^{eav}_{A,\pi}(n) = 1] \leq \frac{1}{2} + negl(n)$$

Diffie-Hellman Key-Exchange.

Common i/p: 1^n

- Alice runs $G(1^n)$ to obtain (G, q, g), where $G = \langle g \rangle$ and $|G| = q$.
- Alice chooses a random $x \leftarrow\$ \mathbb{Z}_q^*$ and computes $h_1 \leftarrow g^x$
- Alice sends (G, q, g, h_1) to Bob.
- Bob chooses a random $y \leftarrow\$ \mathbb{Z}_q^*$ and computes $h_2 \leftarrow g^y$ and sends h_2 back to Alice.

- Bob outputs $k_B \leftarrow h_1^y$
- Alice receives h_2 and outputs $k_A \leftarrow h_2^x$

$$OUT_A^{\Pi}(x, y, 1^n) = k_A = h_2^x = (g^y)^x$$

$$OUT_B^{\Pi}(x, y, 1^n) = k_B = h_1^y = (g^x)^y$$

$$h_1 = g^x, \quad h_2 = g^y.$$

$$Trans_{\Pi}(x, y, 1^n) = (G, q, g, h_1, h_2)$$

Adversary has $(Trans_{\Pi}(x, y, 1^n), k)$

Decisional Diffie-Hellman (DDH):

$$\left| \Pr_{x,y}\left[A\left(\underbrace{G, q, g}_{G(1^n)}, g^x, g^y, g^{xy}\right) = 1\right] - \Pr_{x,y,z}\left[A\left(\overbrace{G, q, g}^{G(1^n)}, g^x, g^y, g^z\right) = 1\right] \right| \leq negl(n)$$

Thm: If DDH is hard relative to G, then Diffie-Hellman key exchange protocol π is secure in the presence of eavesdropper.

Note that $G(1^n) \rightarrow (G, q, g)$

$$P_0\left[KE^{eav}_{A,\pi}(n) = 1\right] > \frac{1}{2} + \varepsilon(n)$$

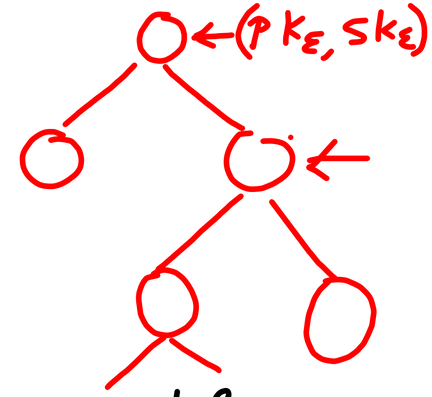
$$= P_0\left[KE^{eav}_{A,\pi}(n) = 1 \wedge b = 0\right] + P_1\left[KE^{eav}_{A,\pi}(n) = 1 \wedge b = 1\right]$$

$$= \frac{1}{2}\left(P_0\left[KE^{eav}_{A,\pi}(n) = 1 \mid b = 0\right] + P_1\left[KE^{eav}_{A,\pi}(n) = 1 \mid b = 1\right]\right)$$

$$= \frac{1}{2}\left(P_0\left[A(G, q, g, g^x, g^y, g^{xy}) = 0\right] + P_0\left[A(G, q, g, g^x, g^y, g^z) = 1\right]\right)$$

Security of Tree-based Signature:

m = 100
m' = 011



Construction:

Gen*: on i/p 1^n, compute $(pk_\varepsilon, sk_\varepsilon) \leftarrow Gen(1^n)$. output the public key pk_ε and the secret key and initial state sk_ε.

we denote $m|_i = m_1 \ldots m_i$ for $i = 0, \ldots, n$, where $m|_0 = \varepsilon$

Sign*: on i/p $m \in \{0,1\}^n$, carry out the following.

(i) for $i = 0$ to $n-1$

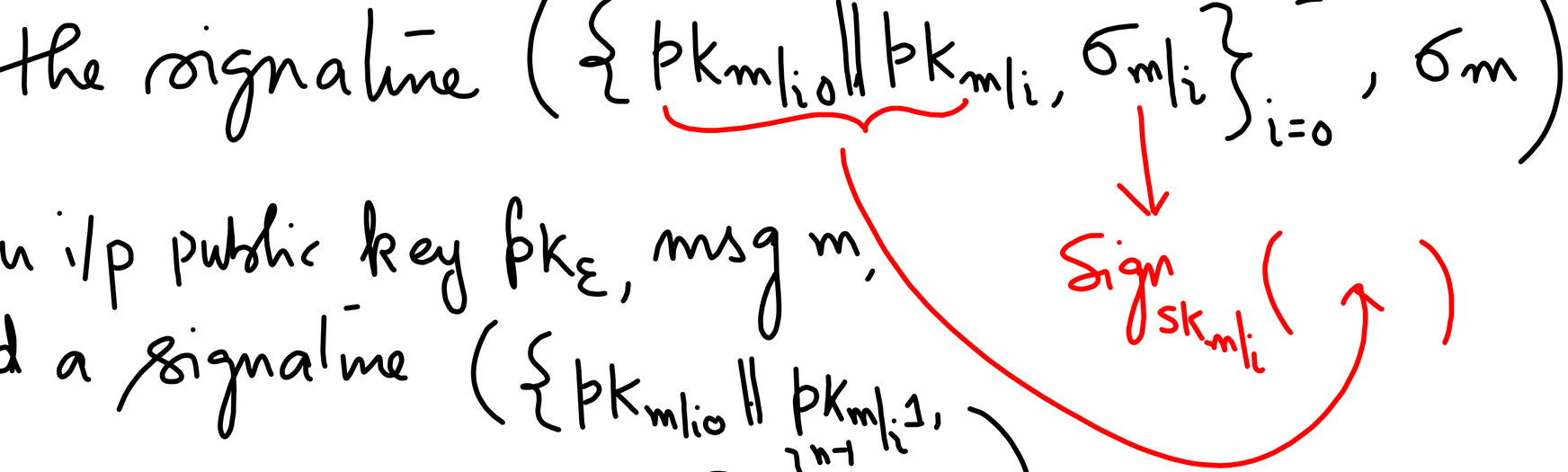
if $PK_{m|_i 0}, pk_{m|_i 1}$ are not in the state, then compute $(pk_{m|_i 0}, sk_{m|_i 0})$ by invoking $Gen(1^n)$. Similarly compute $(pk_{m|_i 1}, sk_{m|_i 1})$

and $\sigma_{m|i} = \text{Sign}_{sk_{m|i}}(pk_{m|i0} \| pk_{m|i1})$. We add all these values to the state.

- End for

2. If σ_m is not yet included in the state, compute $\sigma_m \leftarrow \text{Sign}_{sk_m}(m)$ and store it as a part of the state

3. Output the signature $\left(\{pk_{m|i0} \| pk_{m|i}, \sigma_{m|i}\}_{i=0}^{n-1}, \sigma_m\right)$



- Verify*: on i/p public key pk_ε, msg m, and a signature $\left(\{pk_{m|i0} \| pk_{m|i1}, \sigma_{m|i}\}_{i=0}^{n-1}, \sigma_m\right)$

output $\perp$ if and only if

$$\begin{cases} - \text{Verify}\left(pk_{m|i}, (pk_{m|i0} \,||\, pk_{m|i1}), \sigma_{m|i}\right) = 1 \quad \text{for } i = 0 \ldots, n-1 \\ - \text{Verify}\left(pk_m, m, \sigma_m\right) = 1 \end{cases}$$

↓ verifying the actual msg.

↑ to verify the authenticated public keys.

Thm: Let Π be a one-time signature scheme. Then Π_T is a existentially unforgeable signature scheme.

$A_\Pi \rightarrow$ adv. for Π

$A_{\Pi_T} \rightarrow$ adv for Π_T

A_Π will simulate the role of the challenger for A_{Π_T}

A_Π has i/p (pk). Let us assume that $l(n)$ be the no. of signing queries that A_{Π_T} makes. Let $l^*(n) = (2nl(n)+1) \rightarrow$ denotes the max. no. of public keys required.

Choose $i^* \leftarrow \{1, \ldots, t^*(n)\}$

set $pk^{i^*} \leftarrow pk$.

for every $i \neq i^*$, compute $(pk^i, sk^i) \leftarrow Gen(1^n)$

Run A_{π_T} on public key $pk_\varepsilon = pk^1$. when A_{π_T} requests a signature for a message m, to the following:

for $i = 0$ to $n-1$

if the values $pk_{m|_i 0}, pk_{m|_i 1}, \sigma_{m|_i}$ have not been defined,

set $pk_{m|_i 0}$ and $pk_{m|_i 1}$ equals to the next two unused public keys pk^j, pk^{j+1} respectively and compute the signature on $(pk_{m|_i 0} \| pk_{m|_i 1})$

as $\sigma_{m|i} \leftarrow Sign_{sk_{m|i}}(pk_{m|i0} \| pk_{m|i1})$

Add $(pk_{m|i0} \| pk_{m|i1}, \sigma_{m|i})$ to the slate.

- End for

2. If σ_m is not yet defined, then we compute $\sigma_m \leftarrow Sign_{sk_m}(m)$

3. Give $\left(\left\{ (pk_{m|i0} \| pk_{m|i1}), \sigma_{m|i} \right\}_{i=0}^{n-1}, \sigma_m \right)$ signaling to A_{Π_T}

- Say A_{TH} has output a forgery on a msg m, with signature

$$\left(\left\{ (pk'_{m|_i 0} \,||\, pk'_{m|_i 1}),\ \sigma'_{m|_i} \right\}_{i=0}^{n-1},\ \sigma'_m \right)$$

$$\begin{cases} \text{for every } i = 0, \ldots n-1 \\ \quad \text{Verify}_{pk_{m|_i}} \left(pk'_{m|_i 0} \,||\, pk'_{m|_i 1},\ \sigma'_{m|_i} \right) = 1 \\ \quad \text{Verify}_{pk_m} \left(m, \sigma'_m \right) = 1 \end{cases}$$

$n = 110$

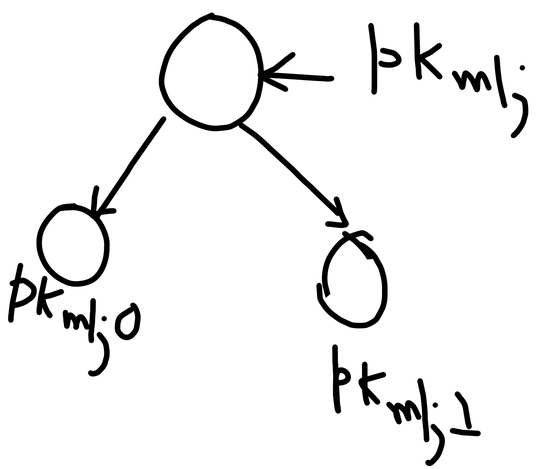
Case 1:

$\exists j \in \{0, \dots, n-1\}$: $pk'_{m|j0} \neq pk_{m|j0}$ or $pk'_{m|j1} \neq pk_{m|j1}$

take the min such j and let i be such that

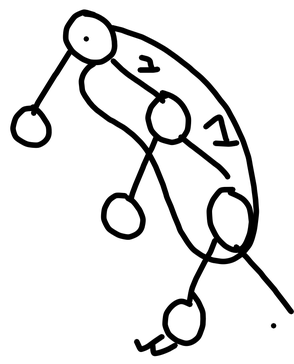
$pk^i = pk_{m|j} = pk'_{m|j}$. If $i = i^*$, then forge with

$(pk'_{m|j0} \| pk'_{m|j1}, \sigma'_{m|j})$



$pk'_m = pk_m$

$\text{Verify}_{pk_{m|j}}(\quad \downarrow \quad) = 1$



Case 2: If (1) does not hold, then $pk'_m = pk_m$. Let i be such that $pk^i = pk_m$. If $i = i^*$, forge with (m, σ'_m)

with prob. at least $\varepsilon(n)$

Conclusion: If A_{Π_T} succesfully forges, then A_Π succesfully forge with prob. at least $\frac{\varepsilon(n)}{\ell^*(n)}$.