Black Board for IBE / IBS lecture

November 28, 2016

1 Security generic IDS.

I guess I fell a bit short on this one, so here is the full proof for completeness. In contrast to the lecture, I will give the complete proof, reducing the (standard) EU-CMA security of the used signature scheme to EU-ID-CMA of the IBS.

First we have to define EU-ID-CMA. Let IBS be an identity-based signature scheme, as defined in the lecture. Consider the following game that uses two oracles

- $\mathcal{O}_{\mathsf{Sign}}(\mathsf{id}, M)$, which returns a signature σ of m, signed using $\mathsf{sk}_{\mathsf{id}}$, and
- $\mathcal{O}_{\mathsf{KeyEx}}(\mathsf{id})$, which returns $\mathsf{sk}_{\mathsf{id}}$.

Experiment $\mathsf{Exp}^{\text{EU-ID-CMA}}_{\text{IBS}}(\mathcal{A})$
$(\mathsf{MSK}, \mathsf{PP}) \longleftarrow \mathsf{mkg}(1^n)$
$(M^\star, \sigma^\star, \mathsf{id}^*) \longleftarrow \mathcal{A}^{\mathcal{O}_{\mathsf{Sign}}(\mathsf{id},M),\mathcal{O}_{\mathsf{KeyEx}}(\mathsf{id})}(\mathsf{PP})$
Let $\{(\mathsf{id}_i, M_i)\}_1^{q_s}$ be the queries to $\mathsf{sign}(\mathsf{sk}, \cdot)$ and
$\{(\mathsf{id})\}_1^{q_e}$ those to $\mathcal{O}_{\mathsf{KeyEx}}(\mathsf{id})$.
Return 1 iff $\mathsf{vf}(\mathsf{id}^*, M^\star, \sigma^\star) = 1$ AND $(\mathsf{id}^*, M^\star) \notin \{\mathsf{id}_i, M_i\}_1^{q_s}$ AND $\mathsf{id}^* \notin \{(\mathsf{id})\}_1^{q_e}$.

For the success probability of an adversary $\mathcal{A}$ in the above experiment we write

$$\mathrm{Succ}^{\text{EU-ID-CMA}}_{\text{IBS}}(\mathcal{A}) = \mathsf{Pr}\left[\mathsf{Exp}^{\text{EU-ID-CMA}}_{\text{IBS}}(\mathcal{A}) = 1\right].$$

Then we call an identity-based signature scheme EU-ID-CMA secure if for all PPT algorithms $\mathcal{A}$, $\mathrm{Succ}^{\text{EU-ID-CMA}}_{\text{IBS}}(\mathcal{A})$ is negligible in the security parameter.

We now want to prove that if Dss is an EU-CMA-secure signature scheme, the certificate-based IBS, described on slide 12 of the lecture is EU-ID-CMA secure. The idea is that the reduction randomly replaces one of the used instances of Dss with the instance for which it has to break EU-CMA security. Then we show that a forgery for the IBS has to include a forgery for at least one Dss instance. We conclude, showing that we replaced exactly this instance with non-negligible probability, meaning we learned a forgery for the target Dss instance with non-negligible probability.

Proof. Towards a contradiction, assume there exists a PPT adversary $\mathcal{A}$ making q_s signing and q_e extraction queries, breaking the EU-ID-CMA security of IBS. Then we can build an oracle machine $\mathcal{M}^{\mathcal{A}}$ that, using $\mathcal{A}$, breaks the EU-CMA-security of Dss as follows. Remember that $\mathcal{M}^{\mathcal{A}}$ takes as an input a public key pk_c for Dss and access to a signing oracle that allows it to obtain valid signatures

of arbitrary messages under pk as it plays in the EU-CMA game. Now assume that $\mathcal{M}^{\mathcal{A}}$ knows for how many different id's $\mathcal{A}$ will make some queries (of any type). Call this number q. Obviously, $q \leq q_s + q_e$ and it must be polynomially bounded as $\mathcal{A}$ runs in polynomial time (and each query takes at least time 1).

First, $\mathcal{M}^{\mathcal{A}}$ samples a random index $i \in [0, q]$. $\mathcal{M}^{\mathcal{A}}$ will use pk_c and the signing oracle

- for the master key pair, if $i = 0$, or
- for the ith identity that the adversary makes any query for.

For all other identities, $\mathcal{M}^{\mathcal{A}}$ just behaves like the key generation center and generates a new key pair. Note that $\mathcal{M}^{\mathcal{A}}$ can answer all signing queries. For all identities but the ith one, $\mathcal{M}^{\mathcal{A}}$ knows the secret key. For the ith identity, $\mathcal{M}^{\mathcal{A}}$ forwards any signing query to the signing oracle for pk_c. Also note that $\mathcal{M}^{\mathcal{A}}$ can answer all key extraction queries besides the one for the ith identity. If $\mathcal{A}$ makes a key extraction query for the ith identity, $\mathcal{M}^{\mathcal{A}}$ aborts. Otherwise, when $\mathcal{A}$ returns a valid forgery $(M^\star, \sigma^\star, \mathsf{id}^*)$, there are two mutually exclusive cases:

- If id^* did not appear in any query before, then there exists no certificate for id^*. This means, $\mathcal{A}$ has forged a signature cert on message $(\mathsf{id} \| pk)$, where pk is the public key $\mathcal{A}$ used for id^*. In that case, if $i = 0$, $\mathcal{M}^{\mathcal{A}}$ returns this forgery, else, $\mathcal{M}^{\mathcal{A}}$ aborts.
- If id^* did appear in a query, there are two more cases.
  - Either, the public key pk that $\mathcal{A}$ uses for id^* is the same as the one used in the queries. In this case, $\mathcal{A}$ must have forged the user signature. So, if id^* is the ith identity, $\mathcal{M}^{\mathcal{A}}$ extracts the user signature from the IBS signature $\sigma^\star$ and outputs it, or aborts otherwise.
  - Or, if pk is different from the one that $\mathcal{M}^{\mathcal{A}}$ used in previous queries, obviously $\mathcal{A}$ must have forged a certificate, i.e., a signature under the master key pair. In this case $\mathcal{M}^{\mathcal{A}}$ extracts the forgery as above if $i = 0$, or aborts if $i \neq 0$.

Summing up, $\mathcal{A}$ always either forges a certificate or a user signature. The former is useful for $\mathcal{M}^{\mathcal{A}}$ if $i = 0$, the latter if id^* is the ith identity.

We can now analyze the success probability, categorizing $\mathcal{A}$ depending on the type of forgery she does. If $\mathcal{A}$ forges a certificate, $\mathcal{M}^{\mathcal{A}}$ succeeds with $1/(q+1)$ times the probability that $\mathcal{A}$ succeeds as $i = 0$ with probability $1/(q+1)$ and $\mathcal{M}^{\mathcal{A}}$ will not abort in this case as it can answer all extraction queries. If $\mathcal{A}$ forges a user signature, $\mathcal{M}^{\mathcal{A}}$ also succeeds with $1/(q+1)$ times the probability that $\mathcal{A}$ succeeds as id is the ith identity with probability $1/(q+1)$. Under this condition, $\mathcal{M}^{\mathcal{A}}$ also never aborts as $\mathcal{A}$ is not allowed to ask a key extraction query for id^*. Applying a union bound, we see that in total $\mathcal{M}^{\mathcal{A}}$ succeeds with $1/(q+1)$ times the probability that $\mathcal{A}$ succeeds.

□

2 Factoring given $n, d(, e)$

- No nice, simple way!

- just prob. algorithm that succeeds with probability 1/2:
- Since $ed \equiv 1 \mod \phi(n)$ there exists $k \in \mathbb{Z}$ such that $ed - 1 = k\phi(n)$.
- Consequently, $a^{ed-1} \equiv 1 \mod n \qquad (\forall a \in \mathbb{Z}_n^*)$.
- Let $ed - 1 = 2^s t$ for odd t.
- Then there exists $i \in [1, s]$ such that

  $$a^{2^{i-1}t} \not\equiv \pm 1 \mod n \text{ and } a^{2^i t} \equiv 1$$

  for at least half of all $a \in \mathbb{Z}_n^*$. (We need a non-trivial root of 1. As 1 has four roots, two of which are ± 1, prob. is at least 0.5.)
- For such a, i it follows that $\gcd(a^{2^{i-1}t} - 1, n)$ is a non-trivial factor of n. (A non-trivial root is $x \equiv 1 \mod q$ and $x \equiv -1 \mod p$, or the other way around. Hence $x - 1 \equiv 0 \mod q$ and $x - 1 \equiv -2 \mod p$, or the other way around... which means, it is a multiple of q.)
- Just select a until you find such a, i.

3 IBE proof

- Board pic for interplay of adversaries (I did not tex this, but you can extract the details from the security notion definitions).
- A_1 is adversary against scheme, A_2 against PKE, B against BDH.
- Interface A_1: key extraction queries, challenge query, access to H_1, H_2.
- Interface A_2: Challenge query, access to H_2.
- Interface B: Given $P \xleftarrow{\$} \mathbb{G}^*$ and aP, bP, cP for $a, b, c \in \mathbb{Z}_p$, compute $\hat{e}(P, P)^{abc}$.