CS 346 Class Notes

Mark Lindberg

Feb 10, 2016

Last Time:

PRF F gives a (strongly) secure Mac for n-bit messages. $\Pi'(\mathsf{Mac}', \mathsf{Vrfy}')$.

We can use the Mac above to produce a (strongly) secure Mac for arbitrary-length messages. $\Pi(\mathsf{Mac}, \mathsf{Vrfy})$.

The construction of Π was: Mac: Divide the message m into $\frac{n}{4}$-bit blocks. We have an n-bit key k. $m = m_1, m_2, \ldots, m_d$ when broken up. m_d, the last block, is padded with 0s if necessary. Let ℓ be the length of the message before padding. Then $t_i = \mathsf{Mac}'_k(r||\langle \ell \rangle||\langle i \rangle||m_i)$, where r is an uniformly chosen $\frac{n}{4}$-bit string, which is used for the entire message m, then sent with for verification. Tag $= (r, t_1, t_2, \ldots, t_d)$.

$\mathsf{Vrfy}(m, t)$, where t is of the form above. Construct each of the blocks as above, using the given r and m, and then run Mac' on each of them and check equality.

This Time:

Theorem: This scheme is secure as long as Π' is secure.

Recall: Our "experiment" for security is the $\mathsf{Mac\text{-}forge}$ experiment. $\Pr[\mathsf{Mac\text{-}forge}_{\mathcal{A},\Pi}(n) = 1]$ is `negl`.

$\mathsf{Mac\text{-}forge}$ means $\mathcal{A}$ gets a Mac_k oracle, $m_1 \to t_1$, $m_2 \to t_2$, etc., leading to an output of (m, t). A wins if for some "new" m it chooses $\mathsf{Vrfy}_k(m, t)$ passes.

Fix an arbitrary PPT A.

Proof: We have 3 events.

E_1: $\mathcal{A}$ succeeds.

E_2: "Repeat": Some r repeats.

E_3: "New Block": Some $(r||\langle \ell \rangle||\langle i \rangle||m_i)$ is passed to Mac'_k when checking $\mathcal{A}$'s output is "new".

$$\begin{aligned}
\Pr[E_1] &= \Pr[E_1 \wedge E_2] + \Pr[E_1 \wedge \overline{E_2} \wedge E_3] + \Pr[E_1 \wedge \overline{E_2} \wedge \overline{E_3}] \\
&\leq \Pr[E_2] + \Pr[E_1 \wedge E_3] + \Pr[E_1 \wedge \overline{E_2} \wedge \overline{E_3}]
\end{aligned}$$

Then $\Pr[E_2] = \mathcal{O}\left(\frac{q(n)}{2^n}\right)$ (I can't read that part of the board very well, so don't trust this.)

Also, $\Pr[[E_1 \wedge \overline{E_2} \wedge \overline{E_3}] = 0$, because $\overline{E_2} \wedge \overline{E_3}$ means we *have* to send an m we've already seen, thus breaking success. That is, $\overline{E_2} \wedge \overline{E_3} \Rightarrow \overline{E_1}$.

Finally, $\Pr[E_1 \wedge E_3]$. If this is not $\texttt{negl}$, then Π' is not secure. Assume we have an adversary $\mathcal{A}'$ for Π'. Then when $\mathcal{A}$ calls $\mathsf{Mac}_k(m)$, $\mathcal{A}'$ chooses a random r, forms blocks of the proper form, then uses oracle for Mac'_k to get t'_is. When $\mathcal{A}$ outputs (m, t), $\mathcal{A}'$ outputs...? It forms all d blocks, and does something. Need to reference the book. Conclusion: $\texttt{negl}(n) \geq \Pr[\mathcal{A}' \text{ succeeds}] \geq \Pr[E_1 \cap E_3]$.

Therefore, $\Pr[E_1] \leq \texttt{negl}(n)$.

Now we are going to examine a *practical* secure Mac for arbitrary length message. This is the CBC-MAC, Cipher Block Chaining Message Authentication Code.

Let $IV = 0^n$. Run CBC, and only output c_d, the last block of the cipher-text produced by the CBC. We assume that all messages m are of length $\ell(n)$-bits.

This doesn't work for arbitrary-length messages, so we can set some long length, and pad shorter messages. Or we could include some F'_k, and we output $F'_k(c_d)$. Either fix the arbitrary-length message.

<u>Authenticated Encryption</u>:

- Combines confidentiality and integrity.

An authenticated encryption scheme must be

1. Unforgeable.
2. CCA-secure.

What does it mean to be unforgeable? Let $\Pi = (\mathsf{Gen}, \mathsf{Enc}, \mathsf{Dec})$. Π is said to be unforgeable if it passes the $\mathsf{Enc\text{-}forge}_{\mathcal{A},\Pi}(n)$ experiment.

The experiment:

$\mathcal{A}$ gets access to Enc_k oracle, and $\mathcal{A}$ must produce a ciphertext c. $\mathcal{A}$ wins the game if $\mathsf{Dec}_k(c) \neq \perp$. (?) And it is "new" (Not the output of some oracle query).

"Encrypt-then-authenticate". Ingredients

1. CPA-secure encryption scheme Π_E.
2. Strongly secure Mac Π_M.

With different keys chosen uniformly at random.