Private Key Authentication

Integrity of message (nobody modified the message) and to know that the message is from a certain person. Encryption doesn't guarantee anything.

Message Authentication Codes (MACs)

Message authentication: Alice sends $(m, t = \text{Tag}_k(m))$ and Bob verifies whether $t = \text{Tag}_k(m)$. Eve can see $(m, t = \text{Tag}_k(m))$ but shouldn't be able to create a valid tag t' for any message $m' \neq m$. Tag is always the same for the same message. Message authentication does not guarantee that the message is from a specific person!

MAC A pair $\text{Vrfy}_k(m, t) \in \text{yes}, \text{no}$ and $\text{Tag}_k(m)$ where $\text{Vrfy}_k(m, \text{Tag}_k(m)) = \text{yes}$ should hold.

Security

An adversary breaks the MAC scheme if they output (m', t') such that $\text{Vrfy}_k(m', t') = \text{yes}$ and $m' \neq m_1 \ldots, m_w$ for all previous messages m_i received from an oracle with randomly chosen key k. This is the strongest possible adversary.

A MAC is secure if $\forall$ polynomial time adversaries A, $P(\text{A breaks MAC}) = \epsilon(n)$ (negligible in n).

Nothing prevents replay attacks though, Eve could always just resend (m, t) to Bob.

Construction of MACs block ciphers

For a block cipher $F : \{0,1\}^n \times \{0,1\}^n \to \{0,1\}^n$, a secure MAC for messages $m \in \{0,1\}^n$ is $\text{Tag}_k(m) = F(k, m)$. This way we send the message and the encrypted message and Bob just needs to reencrypt the message and check if it's the same.

Longer messages

Idea 1: Each block separately, does not work, because a permutation of message and tag $(m' = \text{perm}(m), t' = \text{perm}(t))$ is still valid.

Idea 2: Add counter to each message, does not work, prefix of message still valid.

Idea 3: Add length and counter before each block encryption, does not work, can use different parts of multiple messages, mix and match.

Idea 4: Add a (per message) fresh random value and length and counter to each block. Does work but very space inefficient, up to 4 times larger.

CBC-MAC

$\text{Tag}_k(m) = F_k(m_d \oplus (F_k(m_{d-1} \oplus \ldots F_k(|m|))))$

Hash functions

Definition

Constructions

From hash functions to MACs