CS 127/CSCI E-127: Introduction to Cryptography

Prof. Salil Vadhan — Fall 2013

Lecture Notes 16:

Message Authentication Codes

Reading.

- Katz-Lindell 4.1–4.5

1 The Problem

- *Data authenticity:* How do you know whether a message you receive actually came from who you think it did? And was not tampered with?
- Telephone: their voice, Written letters: handwriting, signature, Electronic communications? E-mail headers?
- Not provided by encryption! The authenticity problem is different from the privacy problem. Here we want data integrity, but many encryption schemes are "malleable" (i.e. the plaintext can easily be modified by modifying the ciphertext). For example:

- Idea: attach a "tag" or "signature" to every message that "authenticates" it as coming from a particular party.
  - Message Authentication Codes: private-key version. The two parties, sender and receiver, share a private key to verify that the message comes from the person whom the key is shared with.
  - Digital Signatures: public-key version. anyone can verify.

2 The Definition

Definition 1 *A* message authentication code *consists of three algorithms* $(\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ *such that:*

- *The* key generation *algorithm* Gen *is a randomized algorithm that returns a* key k*; we write* $k \xleftarrow{R} \mathsf{Gen}(1^n)$.
- *The* tagging *algorithm* Mac *is a (possibly) randomized algorithm that takes a key* k *and a* message m *and outputs a tag* t*; we write* $t \xleftarrow{R} \mathsf{Mac}_k(m)$.
- *The* verification *algorithm* Vrfy *is a deterministic algorithm that takes a key* k*, a message* m*, and a tag* t*, and outputs* $\mathsf{Vrfy}_k(m,t) \in \{\texttt{accept}, \texttt{reject}\}$.

Associated with the scheme is a message space $\mathcal{M}$ *from which* m *is allowed to be drawn. We require* $\mathsf{Vrfy}_k(m, \mathsf{Mac}_k(m)) = \mathtt{accept}$ *for all* $m \in \mathcal{M}$, $k \xleftarrow{R} \mathsf{Gen}(1^n)$.

May allow randomized or stateful tagging algorithms, but (unlike) encryption, deterministic stateless schemes are possible.

Defining security:

- The adversary's goal is to produce a forgery, i.e. produce any pair (m, t) such that $\mathsf{Vrfy}_k(m, t) = \mathtt{accept}$. We will not make any assumptions on the formatting of messages, so even if m is nonsensical, it still counts as a forgery.
- Attack model: chosen message attack. The adversary selects messages m_i and gets to see their tags t_i before trying to produce a forgery. We allow an adaptive attack, i.e. the adversary can select m_{i+1} based on $(m_1, t_1), \ldots, (m_i, t_i)$.
- Unavoidable attacks: we will not protect against replay attacks in our definition (though there are various ways of accomplishing this, through a stateful verification algorithm). We will require that the forgery (m, t) is not one of the adversary's queries.

Definition 2 (existential unforgeability under adaptive chosen message attack) *A message authentication scheme* $(\mathsf{Gen}, \mathsf{Mac}, \mathsf{Vrfy})$ *is* secure *if for every PPT* $\mathcal{A}$*, there is a negligible function* ε *such that*

$$\Pr\left[\mathcal{A}^{\mathsf{Mac}_k(\cdot)}(1^n) \text{ forges}\right] \leq \varepsilon(n) \qquad \forall n,$$

"$\mathcal{A}$ *forges"* $\equiv$ *A produces a pair* (m, t) *for which (a)* $\mathsf{Vrfy}_k(m, t) = \mathtt{accept}$*, and (b)* m *is different from all of* $\mathcal{A}$*'s queries to the* Mac_k*-oracle.*

Equivalently, $\mathcal{A}$ *succeeds with negligible probability in the* message authentication game*:*

- Preventing Replay Attacks: time stamps, counters, unique identifiers.
- As usual, definition is conservative, errs on safe side.

3 MACs for Fixed Length

Simple construction: $\mathsf{Mac}_k(m) \stackrel{\text{def}}{=} f_k(m)$ where $\mathcal{F}_n = \{f_k : \{0,1\}^n \to \{0,1\}^n\}$ is a pseudorandom function family.

$$\mathsf{Vrfy}_k(m, t) = \begin{cases} \mathtt{accept} & \text{iff } f_k(m) = t \\ \mathtt{reject} & \text{otherwise} \end{cases}$$

Note that the construction is deterministic and stateless.

Theorem 3 *If* $\mathcal{F} = \bigcup_n \mathcal{F}_n$ *is a pseudorandom function family, then the MAC defined above is secure.*

Proof: Let A be any PPT.

Claim 4 *The probability that A forges when a truly random function is used (i.e., in the "Ideal MAC") is $\leq 2^{-n}$.*

Proof of claim: Use the dynamic view of the truly random function f: the values of f are generated "on the fly" (a special case of the "principle of deferred decisions").

Claim 5 *The probability that A forges when a pseudorandom function is used is at most $2^{-n} + \mathrm{neg}(n)$.*

Proof of claim:

■

Thus, under the assumption that a block cipher like AES is a family of pseudorandom permutations, we directly obtain very efficient MACs for fixed message length.