CS 6260

Applied Cryptography

Message Authentication Codes (MACs).
New cryptographic goals

• Data privacy is not the only important cryptographic goal

• It is also important that a receiver is assured that the data it receives has come from the sender and has not been modified on the way (and detect if it is not the case)

• The goals are data authenticity and integrity
Encryption solves data privacy, not authenticity/integrity

- Recall OneTimePad: $E(K,M) = K \oplus M$

\[
\begin{align*}
  K & \quad C = K \oplus M \\
  \text{Sender } S & \quad \rightarrow \\
  A & \quad C' = K \oplus M' \\
  & \quad \leftarrow \\
  K & \quad \text{Receiver } R
\end{align*}
\]

R gets $M \oplus M'$ instead of $M$
Message Authentication Code (MAC)

- is the primitive for the goal of data authenticity in the symmetric-key setting

\[ \Pi = (K, MAC, VF) \]

\( \text{MsgSp}-\text{message space} \)

It is required that for every \( M \in \text{MsgSp} \) and every \( K \) that can be output by \( K \),

\[ VF(K, M, MAC(K, M)) = 1 \]
Message Authentication Code (MAC)

• If the key-generation algorithm simply picks a random string from some KeySp, then KeySp describes $\mathcal{K}$

• If the MAC algorithm is deterministic, then the verification algorithm $\mathcal{VF}$ does not have to be defined as it simply re-computes the MAC by invoking the MAC algorithm on the given message $M$ and accepts iff the result is equal to its input TAG.
Towards a security definition for MACs

• We imagine that an adversary can see some number of message plus tag pairs

• As usual, it is necessary but not sufficient to require that no adversary can compute the secret key

• Right now we will not be concerned with *replay attacks*

• We don’t want an adversary to be able to compute a new message and a tag such that the receiver accepts (outputs 1).
Security definition for MACs

Fix $\Pi=(K,\text{MAC},\text{VF})$
Run $K$ to get $K$

For an adversary $A$ consider an experiment $\text{Exp}_{\Pi}^{\text{uf-cma}}(A)$

Return 1 iff $\text{VF}(K,M,\text{Tag})=1$ and $M$ was not queried to the MAC oracle

The uf-cma advantage of $A$ is defined as

$$\text{Adv}_{\Pi}^{\text{uf-cma}}(A) = \Pr [\text{Exp}_{\Pi}^{\text{uf-cma}}(A) = 1]$$

UF-CMA security is defined the usual way.
Examples

We fix a PRF $F$: $\{0, 1\}^k \times \{0, 1\}^\ell \rightarrow \{0, 1\}^L$

$\Pi_1 = (\mathcal{K}, \text{MAC})$

**algorithm** $\text{MAC}_K(M)$

- if $(|M| \mod \ell \neq 0$ or $|M| = 0)$ then return ⊥
- Break $M$ into $\ell$ bit blocks $M = M[1] \ldots M[n]$
- for $i = 1, \ldots, n$ do $y_i \leftarrow F_K(M[i])$
- $\text{Tag} \leftarrow y_1 \oplus \cdots \oplus y_n$
- return $\text{Tag}$

It is easy to construct $A_1$ s.t. $\text{Adv}^{\text{uf-cma}}_{\Pi_1}(A_1) = 1$. 
Examples

We fix a PRF $F$: $\{0,1\}^k \times \{0,1\}^\ell \to \{0,1\}^L$

$\Pi_2 = (\mathcal{K},\text{MAC})$

**Algorithm** $\text{MAC}_K(M)$

1. $l \leftarrow \ell - m$
2. if $(|M| \mod l \neq 0$ or $|M| = 0$ or $|M|/l \geq 2^m)$ then return ⊥
3. Break $M$ into $l$ bit blocks $M = M[1] \ldots M[n]$
4. for $i = 1, \ldots, n$ do $y_i \leftarrow F_K([i]_m || M[i])$
5. $\text{Tag} \leftarrow y_1 \oplus \cdots \oplus y_n$
6. return $\text{Tag}$

Adversary $A_2^{\text{MAC}_K(\cdot)}$

1. Let $a_1, b_1$ be distinct, $\ell - m$ bit strings
2. Let $a_2, b_2$ be distinct $\ell - m$ bit strings
3. $Tag_1 \leftarrow \text{MAC}_K(a_1a_2)$; $Tag_2 \leftarrow \text{MAC}_K(a_1b_2)$; $Tag_3 \leftarrow \text{MAC}_K(b_1a_2)$
4. $Tag \leftarrow Tag_1 \oplus Tag_2 \oplus Tag_3$
5. return($b_1b_2$,Tag)

$\text{Adv}_{\Pi_2}^{\text{uf-cma}}(A_2) = 1$
Note

• We broke the MAC schemes without breaking the underlying function families (they are secure PRFs).

• The weaknesses were in the schemes, not the tools
A PRF as a MAC

Fix a function family $F: \text{Keys} \times D \rightarrow \{0, 1\}^\tau$

Consider a MAC $\Pi = (K, \text{MAC})$

algorithm $K$

$K \leftarrow$ Keys

return $K$

algorithm $\text{MAC}_K (M)$

if $(M \not\in D)$ then return $\perp$

$Tag \leftarrow F_K (M)$

Return $Tag$

Theorem. Let $A$ be an adversary attacking $\Pi$ making $qma$ MAC oracle queries of total length $mma$ and $qva$ verification oracle queries of total length $mva$ and running time $ta$. Then there exists an adversary $B$ attacking $F$ as a PRF such that

$$\text{Adv}_{II}^{uf-cma} (A) \leq \text{Adv}_{F}^{prf} (B) + \frac{1}{2^\tau}$$

and $B$ makes $qma+qva+1$ queries and runs the time $ta+qva tc$, where $tc$ is the time to compare strings of the tag length. The total length of the queries is at most $mma+mva$ plus the largest length of strings in $D$. 
• **Proof.**

Adversary $B^f$

\[ d \leftarrow 0 ; \ S \leftarrow \emptyset \]

Run $A$

When $A$ asks its signing oracle some query $M$:

Answer $f(M)$ to $A$; $S \leftarrow S \cup \{M\}$

When $A$ asks its verification oracle some query $(M, \text{Tag})$:

\[ \text{if } f(M) = \text{Tag } \text{then} \]

\[ \text{answer 1 to } A \]

\[ \text{else answer 0 to } A \]

When $A$ outputs forgery $(M', t')$

If $f(m') = t'$ then return 1

otherwise return 0

\[
\Pr \left[ \text{Exp}_{F}^{\text{prf}-1} (B) = 1 \right] = \text{Adv}_{\Pi}^{\text{uf-cma}} (A)
\]

\[
\Pr \left[ \text{Exp}_{F}^{\text{prf}-0} (B) = 1 \right] \leq \frac{1}{2^\tau}
\]
• Any PRF makes a good MAC

• Are we done?

• Efficient PRFs (e.g. block ciphers) has short fixed input length

• We want it to work for arbitrary-length messages

• What if we hash a message first before applying the block cipher:

\[ H_{K1}(M) \]

\[ H_{K1}(M) \rightarrow 128 \text{ bit} \]

\[ E_{K2} \]

\[ \text{TAG} \]
What H will be good?

- **Definition.** [universal function family] Let $H: \text{KeySp}(H) \times \text{Dom}(H) \rightarrow \text{Ran}(H)$ be a function family. It is called universal if
  \[
  \forall X,Y \in \text{Dom}(H) \text{ s.t. } X \neq Y: \Pr[H_K(X) = H_K(Y)] = 1/|\text{Ran}(H)|
  \]

- **“Matrix” Construction.** Let $\text{KeySp}(H)$ be a set of all $n \times m$ matrices, where each element can be either 0 or 1. Let $\text{Dom}(H) = \{0,1\}^m$, $\text{Ran}(H) = \{0,1\}^n$. Define $H_K(X) = K \cdot X$ (where addition is mod 2)

- **Claim.** The above “matrix” function family is universal.
• The problem with the matrix construction is that the key is big.

• There are other efficient constructions of universal hash functions

• But will combining a universal hash and a PRF will really give us a secure MAC?

• Yes. And let’s prove it.
"Hash-and-PRF" MAC

- **Construction.** Let $H: \text{KeySp}(H) \times \text{Dom}(H) \rightarrow \text{Ran}(H)$ and $F: \text{KeySp}(F) \times \text{Ran}(H) \rightarrow \text{Ran}(F)$ be function families. Define a MAC $\text{HPRF}=(K,\text{MAC},\text{VF})$ with $\text{MsgSp}=\text{Dom}(H)$ as follows:

  - $K$: $K_1 \leftarrow \text{KeySp}(H)$, $K_2 \leftarrow \text{KeySp}(F)$, Return $K_1||K_2$
  
  - $\text{MAC}(K_1||K_2,M)$: $\text{Tag} \leftarrow F_{K_2}(H_{K_1}(M))$, Return $\text{Tag}$
  
  - $\text{VF}(K_1||K_2,M,\text{Tag})$: If $\text{Tag}=F_{K_2}(H_{K_1}(M))$ then return 1, otherwise return 0
• **Theorem.** If $F$ is PRF and $H$ is universal, then $\text{HPRF}$ is a secure MAC.

• **Lemma.** If $F$ is PRF and $H$ is universal then $\text{HPRF}$ is PRF.

• **Proof of the Theorem.** Follows from the Lemma and the fact that any PRF is a secure MAC.

• **Proof of the Lemma.** We will prove that for any $A$ there exists $B$ with $t_B = O(t_A)$, $q_B = q_B$ s.t.

\[
\text{Adv}_{\text{HPRF}}^\text{prf}(A) \leq \text{Adv}_F^\text{prf}(B) + \frac{q_A(q_A - 1)}{2 \cdot |\text{Ran}(H)|}
\]
Adversary $B^f$

\[ K1 \leftarrow KeySp(H) \]

Answer $B$’s queries $M$ with $f(HK_1(M))$

Output the same bit $B$ outputs

Let $g$ be a random function with domain $\text{Ran}(H)$ and range $\text{Ran}(F)$

Let $g'$ be a random function with domain $\text{Dom}(H)$ and range $\text{Ran}(F)$

Let $\text{coll}$ be an event when $HK_1(M) = HK_1(M')$ for any two queries $M, M'$ made by $A$

\[
\text{Adv}^\text{prf}_F(B) = \Pr\left[ \text{Exp}^{prf-1}_F(B) \right] - \Pr\left[ \text{Exp}^{prf-0}_F(B) \right]
\]

\[
= \Pr\left[ \text{Exp}^{prf-1}_{HPRF(H \circ F)}(A) \right] - \Pr\left[ \text{Exp}^{prf-0}_{HPRF(H \circ F)}(A) \right]
\]

\[
= \Pr\left[ \text{Exp}^{prf-1}_{HPRF(H \circ F)}(A) \right] - \Pr\left[ \text{Exp}^{prf-1}_{HPRF(H \circ F)}(A) \right] + \Pr\left[ \text{Exp}^{prf-1}_{g'}(A) \right] - \Pr\left[ \text{Exp}^{prf-1}_{HPRF(H \circ F)}(A) \right] - \Pr\left[ \text{Exp}^{prf-1}_{HPRF(H \circ F)}(A) \right]
\]

\[
= \text{Adv}^\text{prf}_{HPRF}(A) + \Pr\left[ \text{Exp}^{prf-1}_{g'}(A) \right] - \Pr\left[ \text{Exp}^{prf-1}_{HPRF(H \circ F)}(A) \right]
\]

\[
= \text{Adv}^\text{prf}_{HPRF}(A) + \Pr\left[ \text{Exp}^{prf-1}_{g'}(A) \right] - \Pr\left[ \text{Exp}^{prf-1}_{HPRF(H \circ F)}(A) \right]
\]

\[
\leq \text{Adv}^\text{prf}_{HPRF}(A) - \Pr\left[ \text{coll} \right] = \text{Adv}^\text{prf}_{HPRF}(A) - \frac{q_A \cdot (q_A - 1)}{2 \cdot \text{Ran}(H)}
\]
Let $E: \{0,1\}^k \times \{0,1\}^n \rightarrow \{0,1\}^n$ be a block cipher. CBC-MAC = ($\{0,1\}^k$, MAC):

$\text{MsgSp} = \{0,1\}^{nm}$ for some $m \geq 1$.

**Theorem.** For any adversary $A$ there exists an adversary $B$ such that

$$
\text{Adv}_{\text{uf-cma}}^{\text{CBC-MAC}} \leq \text{Adv}_{E}^{\text{prp-cpa}}(B) + \frac{m^2 q_A^2}{2^{n-1}}
$$

where $q_B = q_A + 1, t_B = t_A$
Can we use a hash function as a building block?

- SHA1: \( \{0,1\}^{<2^{64}} \rightarrow \{0,1\}^{160} \)

- Collision-resistant: hard to find \( M, M' \) s.t. \( \text{SHA1}(M) = \text{SHA1}(M') \)

- Is it a good idea to use SHA1 as a MAC?

- What about:
  - \( \text{MAC}_K(M) = \text{SHA1}(M || K) \)?
  - \( \text{MAC}_K(M) = \text{SHA1}(K || M) \)?
  - \( \text{MAC}_K(M) = \text{SHA1}(K || M || K) \)?

- Cannot prove security for these constructions.

- Secure construction: HMAC
  - \( \text{HMAC}_K(M) = \text{SHA1}(K \oplus c || \text{SHA1}(K \oplus d || M)) \), where \( c, d \) are some constants
Can we get it all?

- We know how to achieve data privacy (IND-CPA security) and data authenticity/integrity (UF-CMA security) separately.

- Can we achieve the both goals at the same time (can we send messages securely s.t. a sender is assured in their authenticity/integrity)?

- Can we use the existing primitives: encryption schemes and MACs?
Recall: symmetric encryption scheme

A scheme SE is specified a key generation algorithm $K$, an encryption algorithm $E$, and a decryption algorithm $D$.

$SE = (K, E, D)$

MsgSp-message space

It is required that for every $M \in \text{MsgSp}$ and every $K$ that can be output by $K$, $D(K, E(K, M)) = M$.
Recall: IND-CPA security

Fix $SE=(\text{KeySp}, E, D)$

$K \leftarrow \text{KeySp}$

For an adversary $A$ consider an experiment $\text{Exp}_{SE}^{\text{ind-cpa-b}}(A)$

The IND-CPA advantage of $A$ is:

$\text{Adv}_{SE}^{\text{ind-cpa}}(A) = \Pr[\text{Exp}_{SE}^{\text{ind-cpa-1}}(A) = 1] - \Pr[\text{Exp}_{SE}^{\text{ind-cpa-0}}(A) = 1]$

A symmetric encryption scheme $SE$ is indistinguishable under chosen-plaintext attacks if for any adversary $A$ with “reasonable” resources $\text{Adv}_{SE}^{\text{ind-cpa}}(A)$ is “small” (close to 0).
Recall: IND-CCA security

Fix $SE=(KeySp, E, D)$

$K \leftarrow KeySp$

For an adversary $A$ and a bit $b$ consider an experiment $\text{Exp}^{\text{ind-cca}-b}_{SE}(A)$

The IND-CCA advantage of $A$ is:

$$\text{Adv}^{\text{ind-cca}}_{SE}(A) = \Pr[\text{Exp}^{\text{ind-cca}-1}_{SE}(A) = 1] - \Pr[\text{Exp}^{\text{ind-cca}-0}_{SE}(A) = 1]$$

A symmetric encryption scheme $SE$ is indistinguishable under chosen-ciphertext attacks (IND-CCA secure) if for any adversary $A$ with “reasonable” resources $\text{Adv}^{\text{ind-cca}}_{SE}(A)$ is “small” (close to 0).
Integrity (INT-CTXT) of symmetric encryption schemes

Fix $SE=(\text{KeySp}, E, D)$

$K \xleftarrow{\$} \text{KeySp}$

For an adversary $A$ consider an experiment $\mathsf{Exp}_{SE}^{\text{int-ctxt}}(A)$

For an adversary $A$ consider an experiment $\mathsf{Exp}_{SE}^{\text{int-ctxt}}(A)$

\[
\text{Adv}_{SE}^{\text{int-ctxt}}(A) = \Pr \left[ \mathsf{Exp}_{SE}^{\text{int-ctxt}}(A) = 1 \right]
\]
• **Theorem.** \([\text{IND-CPA} \land \text{INT-CTXT} \Rightarrow \text{IND-CCA}]\) For any SE and an adversary \(A\) there exist adversaries \(A_c, A_p\) s.t.

\[
\text{Adv}_{SE}^{\text{ind-cca}}(A) \leq 2 \cdot \text{Adv}_{SE}^{\text{int-ctxt}}(A_c) + \text{Adv}_{SE}^{\text{ind-cpa}}(A_p)
\]

s.t. the adversaries’ resources are about the same

• **Proof.** Let \(E\) denote the event that \(A\) makes at least one valid decryption oracle query \(C\), i.e. \(D_K(C)\neq \perp\)
Adversary $A_c^{\mathcal{E}_K(\cdot),\mathcal{D}_K^*(\cdot)}$

$$b' \leftarrow \{0, 1\}$$

When $A$ makes a query $M_{i,0}, M_{i,1}$
to its left-or-right encryption oracle do
$$A \leftarrow \mathcal{E}_K(M_{i,b'}).$$

When $A$ makes a query $C_i$
to its decryption oracle do
$$v \leftarrow \mathcal{D}_K^*(C_i)$$
If $v = 0$,
then $A \leftarrow \bot$,
else stop.

$$\Pr[b' = b \land E] \leq \Pr[E]$$
$$= \Pr_c[A_c \text{ succeeds}]$$
$$= \text{Adv}_{SE}^{\text{int-ctxt}}(A_c)$$
Adversary $A_p^{E_K(\mathcal{L}\mathcal{R}(\cdot,\cdot,b))}$

When $A$ makes a query $M_{i,0}, M_{i,1}$ to its left-or-right encryption oracle do

$A \leftarrow E_K(\mathcal{L}\mathcal{R}(M_{i,0}, M_{i,1}, b))$

When $A$ makes a query $C_i$ to its decryption oracle do

$A \leftarrow \bot$

$A \Rightarrow b'$

Return $b'$

\[
\Pr \left[ b' = b \land \neg E \right] \leq \Pr_p \left[ b' = b \right]
\]

\[
= \frac{1}{2} \cdot \text{Adv}_{SE}^{\text{int-cpa}}(A_p) + \frac{1}{2}
\]
\[
\frac{1}{2} \cdot \text{Adv}_{SE}^{int-cca}(A) + \frac{1}{2} \\
= \Pr\left[ b' = b \right] \\
= \Pr\left[ b' = b \land E \right] + \Pr\left[ b' = b \land \neg E \right] \\
\leq \frac{1}{2} \cdot \text{Adv}_{SE}^{int-cpa}(A_p) + \text{Adv}_{SE}^{int-ctxt}(A_c) + \frac{1}{2}
\]