Today: Symmetric Encryption (cont.)

1. Encryption scheme for long messages

2. Authenticated encryption: GCM AES (Galois Counter Mode Advanced Ecryption Standard)

Last time:

1. Defined Encryption scheme: CPA (Chosen Plaintext Attack) security.

2. One-time pad

3. Many-time security (CPA security) from PRF

Recall: An encryption scheme with msg space M and key space K, and ciphertext space C, consists of two probabilistic poly-time algorithms:

$Enc: \mathcal{K}\times\mathcal{M}\longrightarrow\mathcal{C}$ and $Dec: \mathcal{K}\times\mathcal{C}\longrightarrow\mathcal{M}$ such that for every k in K and m in M,

$Pr[Dec(k,Enc(k,m))=m]=1$

Recall: Motivated security definition: for every m1, m2 in M:

$Enc(K,m1) \underset{\uparrow}{\cong} Enc(K,m2)$

computationally indistinguishable

This definition is sufficient if no information is given about k.

However, the adv does obtain auxiliary information about k, of the form

Enc(k,m1),...Enc(k,mt), possibly for m1,...,mt in M of his choice!

Motivated the definition of CPA security:

Enc(k,m1) and Enc(k,m2) are computationally indistinguishable even given encryptions of any messages of the adversary's choice. Moreover, these msgs can be chosen adaptively, and m1,m2 can be chosen by the adv after obtaining all these ciphertexts.

Formalized as a game-based definition

One-Time Pad: $\mathrm{Enc}(k,m)=k \oplus m$, $\mathrm{Dec}(k,c)=k \oplus c$

Has only one time security: For every m1,m2 in M: $\mathrm{Enc}(k,m1) \underset{\substack{\uparrow \\ =}}{\cong} \mathrm{Enc}(k,m2)$

perfect security

Idea: Make the one-time pad CPA secure by using a PRF!

Instead of using k as the pad, use F(k,r) as the pad!

For any r, as long as we don't reuse the same r!

Ex. select r at random, then whp we won't reuse the same r

(assuming we don't encrypted too many msgs, enough to hit the birthday paradox).

If we use AES as the PRF, then it allows us to encrypt msgs of length 128.

How do we encrypt longer msgs??

Partition the msg into blocks of length 128, and encrypt each block separately.

$$Enc(K, m1 \| m2 \| \ldots \| mt) = Enc(k, m1) \| Enc(k, m2) \| \ldots \| Enc(k, mt)$$

CPA secure!

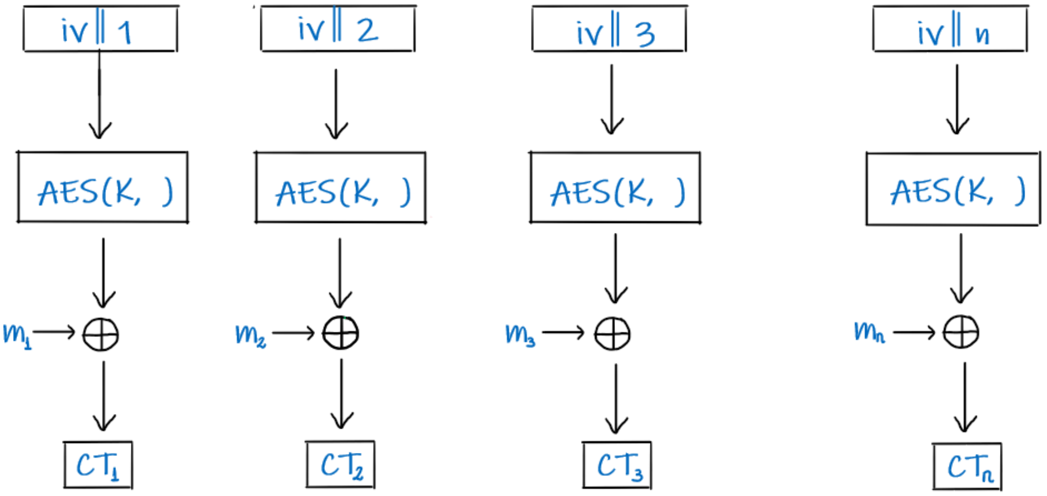
Note: If we use the encryption scheme from last lecture with AES as the PRF:

$$Enc(K, m1 \| \ldots \| mt) = r1 \| AES(k, r1) \oplus m1 \| \ldots \| rt \| AES(k, rt) \oplus mt$$

More efficient instantiation: Reuse the same randomness with a counter!

$$Enc(K, m1 \| \ldots \| mt) = r \| AES(k, r \| 1) \oplus m1 \| \ldots \| AES(k, r \| t) \oplus mt$$

counter mode



This is CPA secure and efficient! as long as the none of the nonces are reused.

This is CPA secure and efficient! as long as the none of the nonces are reused.

Is CPA security sufficient? What about integrity?

Integrity is also important for confidentiality!

If an adv can change Enc(k,m) to Enc(k,m') this can help him find m!

Ex, m' can append to m: "if your output is yes, then send a long string of 0s",

and as we said the ciphertext does not hide the length of msg.

Goal: Construct authenticated CPA encryption scheme.

CPA encryption + MAC

Enc'((k,k'), m) = (ct, MAC(k', ct)), where ct = Enc(k,m)

Note: 1. We are using different keys: k for Enc and k' for MAC.

2. Notice the order: Enc then MAC, and opposed to MAC then Enc.

Both these choices are important for security!

For 1: Recall that in the CPA def, we assume that the only info the adv has about k is ciphertexts of his choice. If the same key is used for other applications then more information about the key is leeked, which may compromise security.

One can use a single key k* and let k = F(k*,0) and k' = F(k*,1), where F is a PRF.

Indeed this is used in GCM AES

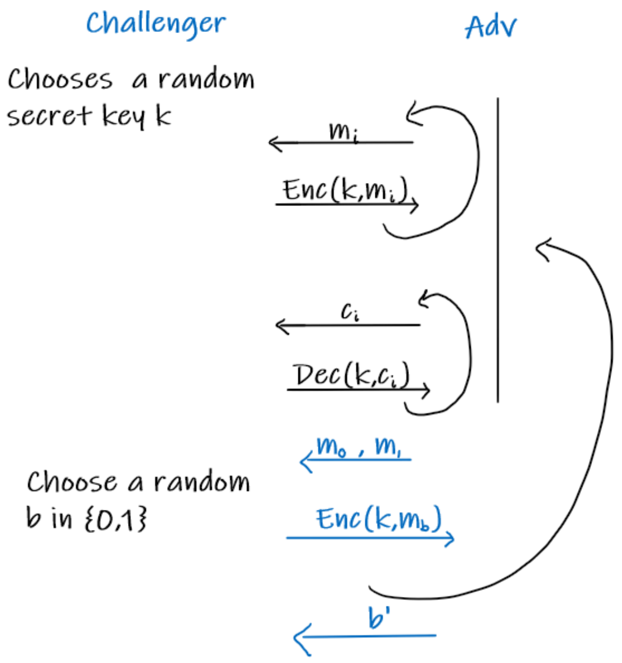
For 2: What goes wrong if we first MAC then Enc?

Desired security definition: Security against Chosen Ciphertext Attacks (CCA)

Similar to CPA security, but the adversary is also allowed to ask to see decryptions of msgs of its choice!

Formalized via the following game-based definition:

Def: An encryption scheme (Enc,Dec) is CCA secure if every efficient adv wins in the following game with prob at most 1/2+negl



Adv wins iff b'=b.

Thm: Let (Enc,Dec) be CPA secure encryption, and let MAC be a secure MAC, then (Enc',Dec') is CCA secure encryption, where

$$Enc'((k,k'), m) = (ct, MAC(k', ct)), \text{ where } ct = Enc(k,m)$$

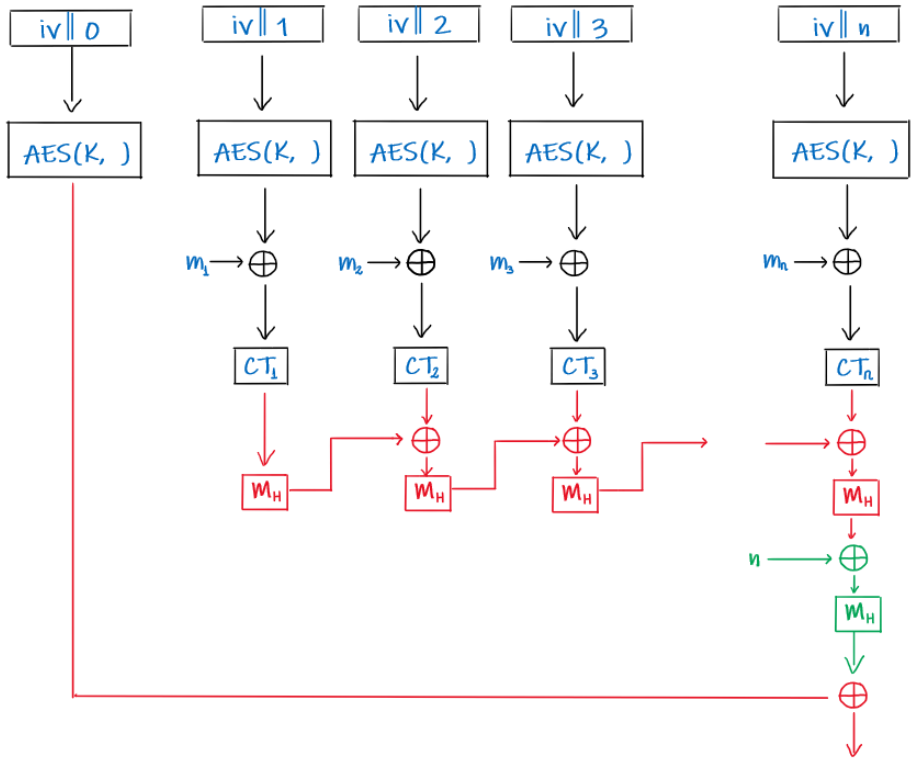
$$Dec'((k,k'), (ct,\sigma)): \text{ If } MAC(k',\sigma)=0 \text{ then output fail. Otherwise, output } Dec(K,ct)$$

Hw: Show that if we first MAC then Enc, the resulting scheme may not be CCA secure!

AES GCM: authenticated encryption scheme used in practice.

CPA secure encryption, using AES in counter mode (as eplained above),

together with GMAC (Galois MAC).



M_H is not a secure MAC!

$M_H(x) = H \cdot x$

where H = AES(k,0) is a string of length 128,

$M_H : \{0,1\}^{128} \longrightarrow \{0,1\}^{128}$ is multiplication by H,

where multiplication is in a finite field of size 2^{128}

known as Galois Field ($GF[2^{128}]$)

M_H is a one-time secure MAC!

where H is the secret key.

We cannot reveal even a single tag!

Also provides an effcient way to authenticate auxiliary unencrypted data, such as IP addresses...

(essentially by adding the data to the GMAC, need to authenticate the length of the auxiliary data).