Authenticated Encryption

Jeremy, Paul, Ken, and Mike

Objectives

- Examine three methods of authenticated encryption and determine the best solution considering performance and security

Basic Components

Message Authentication Code

+

Symmetric Encryption

Both of these components are used as black boxes

Generic Composition

SE - Symmetric encryption scheme

E - encryption algorithm

D - Decryption Algorithm

MA - Message authentication scheme

T - tagging algorithm

V - tag verifing algorithm

K - randomized key generation algorithm

κ - security parameter, length of the key

k - the key

- Note:
  - We separate the tagging and verification algorithm

Basic Components

Message Authentication Code (MAC)

- Integrity / Authenticity
  - Integrity of Plaintext (INT-PTXT)
  - Integrity of Ciphertext (INT-CTXT)

Symmetric Encryption

- Privacy
  - Indistinguishability
    - Chosen-plaintext attack (IND-CPA)
    - Chosen-ciphertext attack (IND-CCA)
  - Non-malleability
    - Chosen-plaintext attack (NM-CPA)
    - Chosen-ciphertext attack (NM-CPA)

Integrity

- Integrity of Plaintext (INT-PTXT)
  - Computationally infeasible to produce a ciphertext decrypting to a message which the sender has never encrypted
- Integrity of Ciphertext (INT-CTXT)
  - Computationally infeasible to produce a ciphertext not previously produced by the sender, regardless of whether or not the underlying plaintext is "new"

Integrity of symmetric encryption schemed

$SE = (E, K, D)$

$$\left.\begin{array}{l}\text{Algorithm } D^*_K(C) \\ \quad \text{If } D_K(C) \neq \perp \text{, then return 1} \\ \quad\quad \text{Else return 0}\end{array}\right\}$$

Verification algorithm
or
Verification oracle

E – Encryption Algorithm

K– Randomized key generation algorithm

D – Decryption Algorithm

Integrity of Authenticated encryption scheme

- The scheme SE is said to be INT-PTXT if the function $Adv_{SE,A_{ptxt}}^{\text{int}-ptxt}(\cdot)$ (the advantage of A_{ptxt}) is very small for any adversary whose time-complexity is polynomial in k.

- Likewise, the scheme SE is said to be INT-CTXT if the function $Adv_{SE,A_{ctxt}}^{\text{int}-ctxt}(\cdot)$ (the advantage of A_{ctxt}) is very small for any adversary whose time-complexity is polynomial in k.

Integrity of Authenticated encryption scheme

Experiment $Exp_{SE,A_{ptxt}}^{\text{int}-ptxt}(k)$

$K \xleftarrow{R} \mathrm{K}(\kappa)$

If $A_{ptxt}^{\mathrm{E}_K(\cdot),D^*(\cdot)}(\kappa)$ makes a query C to

the oracle $D_K^*(\cdot)$ such that

- $D_K^*(C)$ returns 1, and

- $\mathrm{M} \underline{\underline{def}} D_K(C)$ was never a query to $\mathrm{E}_\mathrm{K}(\cdot)$

then return 1 else return 0

Experiment $Exp_{SE,A_{ctxt}}^{\text{int}-ctxt}(k)$

$K \xleftarrow{R} \mathrm{K}(\kappa)$

If $A_{ctxt}^{\mathrm{E}_K(\cdot),D^*(\cdot)}(\kappa)$ makes a query C to

the oracle $D_K^*(\cdot)$ such that

- $D_K^*(C)$ returns 1, and

- C was never a response to $\mathrm{E}_\mathrm{K}(\cdot)$

then return 1 else return 0

$$\left.\begin{aligned} Adv_{SE,A_{ptxt}}^{\text{int}-ptxt}(k) &= \Pr[Exp_{SE,A_{ptxt}}^{\text{int}-ptxt}(k) = 1] \\ Adv_{SE,A_{ctxt}}^{\text{int}-ctxt}(k) &= \Pr[Exp_{SE,A_{ctxt}}^{\text{int}-ctxt}(k) = 1] \end{aligned}\right\} \text{Advantages of the adversaries}$$

$$\left.\begin{aligned} Adv_{SE}^{\text{int}-ptxt}(k,t,q_e,q_d,\mu_e,\mu_d) &= \max_{A_{ptxt}}\{Adv_{SE,A_{ptxt}}^{\text{int}-ptxt}(k)\} \\ Adv_{SE}^{\text{int}-ctxt}(k,t,q_e,q_d,\mu_e,\mu_d) &= \max_{A_{ctxt}}\{Adv_{SE,A_{ctxt}}^{\text{int}-ctxt}(k)\} \end{aligned}\right\} \text{Advantages of the scheme}$$

Indistinguishability

- Indistinguishability of Chosen Plaintext Attack (IND-CPA)
- Indistinguishability of Chosen Ciphertext Attack (IND-CCA)
- If M_0 and M_1 are encrypted, a 'reasonable' adversary should not be able to determine which message is sent.

Left-or-right

$\Sigma_K(LR(.,.,b))$, where b {0, 1}, to take input (M_0, M_1) $|M_0| = |M_1|$

if b = 0

  $C \leftarrow \Sigma_K(M_0)$

  return C

else

  $C \leftarrow \Sigma_K(M_1)$

  return C

- As was mentioned from Adam's lecture, we consider the encryption scheme to be "good" if a "reasonable" adversary cannot obtain "significant" advantage in distinguishing the cases b = 0 and b = 1 given access to the left-or-right oracle.

Non-malleability

- Prevents the generation of a ciphertext whose plaintexts are meaningful
- Requires that an attacker given a challenge ciphertext be unable to modify it into another, different ciphertext in such a way that the plaintexts underlying the two ciphertexts are "meaningful related" to each other.
- i.e.
  - Ptxt1: send a check of $100.00
  - Ptxt2: send a check of $1000.00

Non-malleability - Formally

Experiment $\mathrm{Exp}^{nm-cpa-b}_{\mathrm{SE,\,A_{cpa}}}(b)$

$k \xleftarrow{R} K(\kappa)$

$(\vec{c}, s) \leftarrow A^{E_k(LR(.,.,b))}_{cpa_1}(k)$

$\vec{p} \leftarrow \vec{D}_k(\vec{c})$

$x \leftarrow A_{cpa_2}(\vec{p}, \vec{c}, s)$

return x

Experiment $\mathrm{Exp}^{nm-cca-b}_{\mathrm{SE,\,A_{cca}}}(b)$

$k \xleftarrow{R} K(\kappa)$

$(\vec{c}, s) \leftarrow A^{E_k(LR(.,.,b))}_{cca_1}(k)$

$\vec{p} \leftarrow \vec{D}_k(\vec{c})$

$x \leftarrow A_{cca_2}(\vec{p}, \vec{c}, s)$

return x

SE = (K, E, D)

$b \in \{0,1\}$

$\kappa \in N$

$A_{cpa} = (A_{cpa1}, A_{cpa2})$, 1 oracle

$A_{cca} = (A_{cca1}, A_{cca2})$, 2 oracles

$$Adv^{nm-cpa}_{SE,A_{cpa}}(k) = \Pr[Exp^{nm-cpa-1}_{SE,A_{cpa}}(k) = 1] - \Pr[Exp^{nm-cpa-0}_{SE,A_{cpa}}(k) = 1]$$

$$Adv^{nm-cpa}_{SE,A_{cca}}(k) = \Pr[Exp^{nm-cca-1}_{SE,A_{cca}}(k) = 1] - \Pr[Exp^{nm-cca-0}_{SE,A_{cca}}(k) = 1]$$

$$Adv^{nm-cpa}_{SE}(k, t, q_e, \mu_e) = \max_{A_{cpa}}\{Adv^{nm-cpa}_{SE,A_{cpa}}(k)\}$$ If negligible, NM-CPA Secure

$$Adv^{nm-cca}_{SE}(k, t, q_e, \mu_e) = \max_{A_{cca}}\{Adv^{nm-cca}_{SE,A_{cca}}(k)\}$$ If negligible, NM-CCA Secure

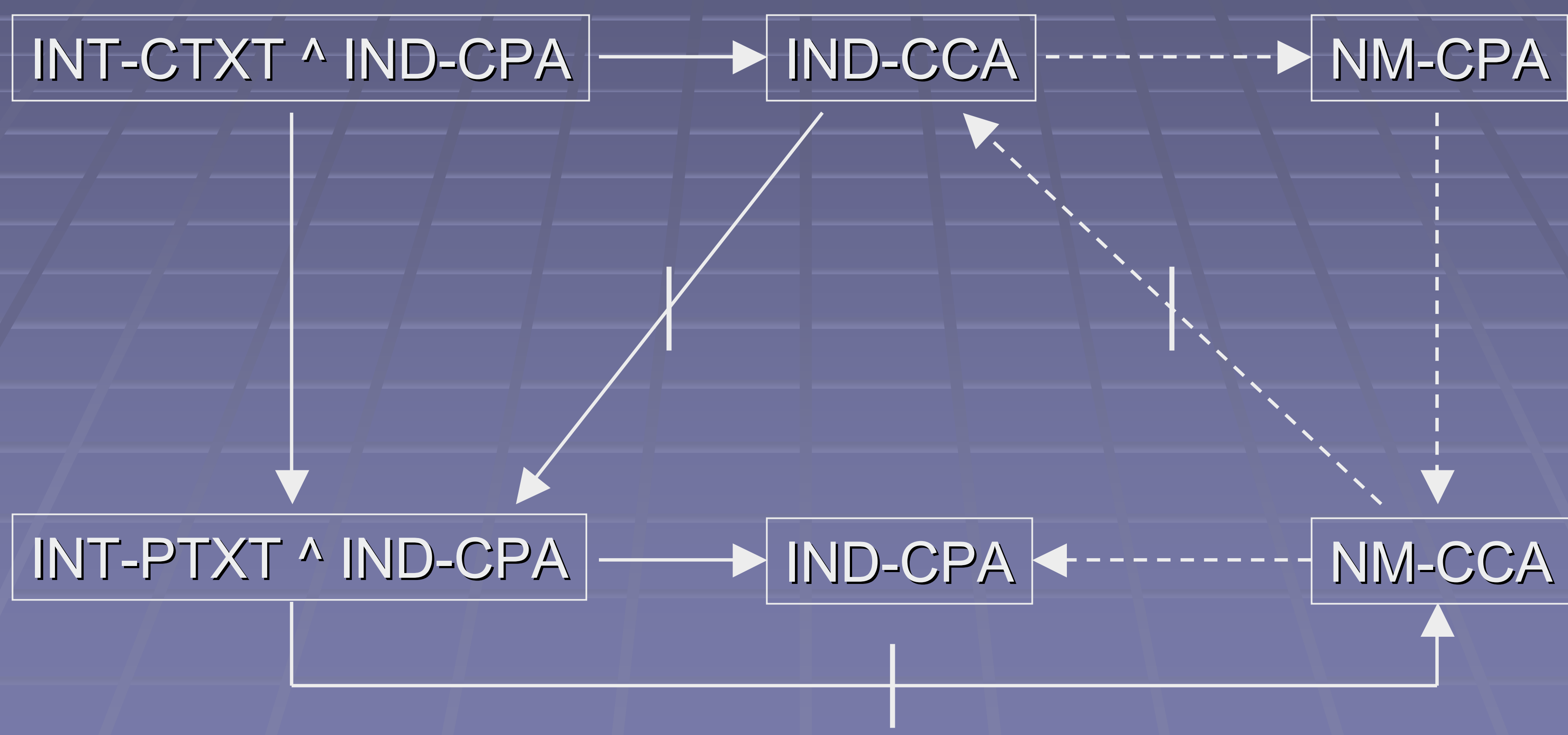
Unforgeability

- Weak Unforgeability against Chosen Message Attacks (WUF-CMA)
  - Adversary *F* can't create a new message and tag
- Strong Unforgeabililty against Chosen Message Attacks (SUF-CMA)
  - Adversary *F* can't create a new tag for an existing message

Difficulties

- The notions of authenticity are by themselves quite disjoint from the notions of privacy
  - i.e. Sending the message in the clear with an accompanying (strong) MAC achieves INT-CTXT but no kind of privacy

Relations among notions of symmetric encryption



Relations among notions of symmetric encryption
Theorem 3.1

$$INT-CTXT \rightarrow INT-PTXT$$

$$Adv_{SE}^{\mathrm{int}-ptxt}(k,t,q_e,q_d,\mu_e,\mu_d) \le Adv_{SE}^{\mathrm{int}-ctxt}(k,t,q_e,q_d,\mu_e,\mu_d)$$

- A – adversary mounting an attack against integrity of plaintexts of SE
- A’ – adversary mounting an attack against integrity of ciphertexts of SE
- A’ = A

Adversary A'(k)

return A(C)

C - is the winning query

$$Adv_{SE,A}^{\mathrm{int}-ptxt}(k) \le Adv_{SE,A'}^{\mathrm{int}-ctxt}(k)$$

It is initiative that if an adversary violates integrity of plaintexts of a scheme SE = (K,E,D) also violates integrity of ciphertexts of the same scheme

Proposition 3.3

- IND-CCA $\not\rightarrow$ INT-PTXT
- Given a symmetric encryption scheme *SE* which is IND-CCA secure, we can construct a symmetric encryption scheme $\overline{SE}$ which is also IND-CCA secure but is *not* INT-PTXT secure

IND-CCA $\not\rightarrow$ INT-PTXT

- Let SE = (K, E, D)
- We define a $\overline{SE}$ such that $\overline{SE}$ is IND-CCA secure but is not INT-PTXT secure
- Basically a certain known string (or strings) will be viewed by $\overline{D}$ as valid and decrypted to certain known messages, so that forgery is easy
- However these 'ciphertexts' will never be produced by the encryption algorithm, so privacy will not be affected

$$\overline{SE} = (K, \overline{E}, \overline{D})$$

Algorithm $\overline{\mathrm{E}}_k(M)$
$C' \leftarrow E_k(M)$
$C \leftarrow 0 \| C'$
Return C

Algorithm $\overline{\mathrm{D}}_k(C)$
Parse C as $b \| C'$ where b is a bit $\leftarrow E_k(M)$
if $b = 0$ then $M \leftarrow D_k(C')$; return M
Else return 0

IND-CCA $\not\rightarrow$ INT-PTXT Attack

Adversary $A^{\overline{E_K(\cdot)},\overline{D_k(\cdot)}}(k)$

Submit query 10 to oracle $\overline{D^*_k}(\cdot)$

•••

$\overline{D^*_k}(10) = 0$

10 →1010

(little Endian, LSB 1st)

$$Adv_{SE,A}^{\text{int}-ptxt}(k) = 1$$

- Query 10 is a valid ciphertext
- It decrypts to a msg (0) that the adversary never queried of its oracle

A makes zero queries to $\overline{E_K(\cdot)}$ and one query to $\overline{D_K(\cdot)}$ totaling 2 bits, and Is Certainly poly(k)-time

IND-CCA $\not\rightarrow$ INT-PTXT
IND-CCA Secure

- To prove that $\overline{SE}$ is IND-CCA secure, it suffices (enough) to associate with any poly(k)-time adversary B attacking SE in the IND-CCA sense such that $Adv^{ind-cca}_{\overline{SE},A}(k) \leq Adv^{ind-cca}_{SE,B}(k)$

$\text{Adversary } \mathrm{B}^{\mathrm{E}_{\mathrm{k}}(LR(.,.,b)),D_k(\cdot)}(k)$
$\quad \text{for i} = 1,\ldots.\mathrm{q}_{\mathrm{e}} + q_d \text{ do}$
$\quad\quad \text{when A makes a query } \mathrm{M}_{\mathrm{i},0}, M_{i,1} \text{ to its left-or-right encryption oracle do}$
$\quad\quad\quad \mathrm{A} \Leftarrow 0 \,||\, \mathrm{E}_{\mathrm{k}}(LR(\mathrm{M}_{\mathrm{i},0}, M_{i,1}, b))$
$\quad\quad \text{when A makes a query } \mathrm{C}_{\mathrm{i}} \text{ to its decryption oracle do}$
$\quad\quad\quad \text{Parse C as } \mathrm{b}_{\mathrm{i}} \,||\, C_i^{'} \text{ where } \mathrm{b}_{\mathrm{i}} \text{ is a bit}$
$\quad\quad\quad \text{if b} = 0 \text{ then A} \Leftarrow \mathrm{D}_{\mathrm{k}}(C_i^{'})$
$\quad\quad\quad \text{Else A} \Leftarrow 0$

B simulates A and Uses its oracles to Answer A's oracle queries

- It is easy for B to break the scheme if A can

Other Relations

- Theorem 3.2
  - INT-CTXT ^ IND-CPA → IND-CCA
- Proposition 3.4
  - INT-PTXT ^ IND-CPA (does not) →NM-CPA

Security of the Composite Schemes

- ***Secure***

  Proven to meet the security requirement, assuming component encryption scheme meets IND-CPA and message authentication scheme is unforgeable under CMA

- ***Insecure***

  Some IND-CPA secure symmetric encryption and some message authentication scheme unforgeable under CMA exist that doesn’t meet the security requirement

Generic Composition

Using both functions as black boxes

MAC

Symmetric Encryption

Encrypt-and-MAC

$$C = \boxed{Encrypt\ (M)} \,\|\, \boxed{MAC\ (M)}$$

Algorithm $\overline{\mathcal{K}}(k)$	Algorithm $\overline{\mathcal{E}}_{\langle K_e,K_m\rangle}(M)$	Algorithm $\overline{\mathcal{D}}_{\langle K_e,K_m\rangle}(C)$
$K_e \xleftarrow{R} \mathcal{K}_e(k)$	$C' \leftarrow \mathcal{E}_{K_e}(M)$	Parse C as $C' \\| \tau$
$K_m \xleftarrow{R} \mathcal{K}_m(k)$	$\tau \leftarrow \mathcal{T}_{K_m}(M)$	$M \leftarrow \mathcal{D}_{K_e}(C')$
`Return` $\langle K_e, K_m \rangle$	$C \leftarrow C' \\| \tau$	$v \leftarrow \mathcal{V}_{K_m}(M, \tau)$
	`Return` C	`If` $v = 1$, `return` M
		`else return` $\perp$.

Encrypt-and-MAC Security

Security		Weak MAC	Strong MAC
Privacy	IND-CPA	*Insecure*	*Insecure*
	IND-CCA	*Insecure*	*Insecure*
	NM-CPA	*Insecure*	*Insecure*
Integrity	INT-PTXT	*Secure*	*Secure*
	INT-CTXT	*Insecure*	*Insecure*

MAC-then-Encrypt

C = $\boxed{\text{Encrypt}\ (M\ \|\ \boxed{\text{MAC}\ (M)}\ )}$

Algorithm $\overline{\mathcal{K}}(k)$ $K_e \xleftarrow{R} \mathcal{K}_e(k)$ $K_m \xleftarrow{R} \mathcal{K}_m(k)$ Return $\langle K_e, K_m \rangle$	Algorithm $\overline{\mathcal{E}}_{\langle K_e, K_m \rangle}(M)$ $\tau \leftarrow \mathcal{T}_{K_m}(M)$ $C \leftarrow \mathcal{E}_{K_e}(M \| \tau)$ Return C	Algorithm $\overline{\mathcal{D}}_{\langle K_e, K_m \rangle}(C)$ $M' \leftarrow \mathcal{D}_{K_e}(C)$ Parse M' as $M \| \tau$ $v \leftarrow \mathcal{V}_{K_m}(M, \tau)$ If $v = 1$, return M else return $\perp$.

MAC-then-Encrypt Security

Security		Weak MAC	Strong MAC
Privacy	IND-CPA	*Secure*	*Secure*
	IND-CCA	*Insecure*	*Insecure*
	NM-CPA	*Insecure*	*Insecure*
Integrity	INT-PTXT	*Secure*	*Secure*
	INT-CTXT	*Insecure*	*Insecure*

Encrypt-then-MAC

$$C = \boxed{\textit{Encrypt}\ (M)} \;\|\; \boxed{\textit{MAC}\ (\ \boxed{\textit{Encrypt}\ (M)}\ )}$$

Algorithm $\overline{\mathcal{K}}(k)$	Algorithm $\overline{\mathcal{E}}_{\langle K_e, K_m \rangle}(M)$	Algorithm $\overline{\mathcal{D}}_{\langle K_e, K_m \rangle}(C)$
$K_e \stackrel{R}{\leftarrow} \mathcal{K}_e(k)$	$C' \leftarrow \mathcal{E}_{K_e}(M)$	Parse C as $C' \| \tau'$
$K_m \stackrel{R}{\leftarrow} \mathcal{K}_m(k)$	$\tau' \leftarrow \mathcal{T}_{K_m}(C')$	$M \leftarrow \mathcal{D}_{K_e}(C')$
`Return` $\langle K_e, K_m \rangle$	$C \leftarrow C' \| \tau'$	$v \leftarrow \mathcal{V}_{K_m}(C', \tau')$
	`Return` C	`If` $v = 1$, `return` M
		`else return` $\perp$.

Encrypt-then-MAC Security

Security		Weak MAC	Strong MAC
Privacy	IND-CPA	*Secure*	*Secure*
	IND-CCA	*Insecure*	*Secure*
	NM-CPA	*Insecure*	*Secure*
Integrity	INT-PTXT	*Secure*	*Secure*
	INT-CTXT	*Insecure*	*Secure*

Summary of Methods

Weakly Unforgeable

Composition Method	Privacy			Integrity	
	IND-CPA	IND-CCA	NM-CPA	INT-PTXT	INT-CTXT
Encrypt-and-MAC	Insecure	Insecure	Insecure	Secure	Insecure
MAC-then-Encrypt	Secure	Insecure	Insecure	Secure	Insecure
Encrypt-then-MAC	Secure	Insecure	Insecure	Secure	Insecure

Strongly Unforgeable

Composition Method	Privacy			Integrity	
	IND-CPA	IND-CCA	NM-CPA	INT-PTXT	INT-CTXT
Encrypt-and-MAC	Insecure	Insecure	Insecure	Secure	Insecure
MAC-then-Encrypt	Secure	Insecure	Insecure	Secure	Insecure
Encrypt-then-MAC	Secure	Secure	Secure	Secure	Secure

Theorem 4.7

- Encrypt-then-MAC method is IND-CPA and INT-PTXT

- SE be a symmetric scheme
- MA be message authentication scheme

$$Adv_{\overline{SE}}^{ind-cpa}(k,t,q,\mu) \leq Adv_{SE}^{ind-cpa}(k,t,q,\mu)$$

$$Adv_{\overline{SE}}^{\mathrm{int}-ptxt}(k,t,q_e,q_d,\mu_e,\mu_d) \leq Adv_{MA}^{wuf-cma}(k,t,q_e,q_d,\mu_e,\mu_d)$$

Theorem 4.7 - IND-CPA

$$Adv_{\overline{SE}}^{ind-cpa}(k) \leq Adv_{SE,A_p}^{ind-cpa}(k,t,q,\mu)$$

$Adversary\ \mathrm{A}_{\mathrm{p}}^{\mathrm{E}_{\mathrm{Ke}}(LR(.,.,b))}(\kappa)$

$\mathrm{k}_{\mathrm{m}} \xleftarrow{R} K_m(\kappa)$

$\mathrm{For\ i = 1,....,\ q\ do}$

$\text{When A makes a query } (\mathrm{M}_{\mathrm{i,o}}, M_{i,1})\ \textit{to its left-or-right} \text{ encryption oracle do}$

$\mathrm{C}_{\mathrm{i}} \leftarrow E_{K_e}(LR(\mathrm{M}_{\mathrm{i,o}}, M_{i,1}, b)); \tau_i \leftarrow \mathrm{T}_{K_m}(C_i); A \Leftarrow C_i \parallel \tau_i$

$\mathrm{A} \Rightarrow \mathrm{b'}$

$\mathrm{Return\ b'}$

Theorem 4.7 - INT-PTXT

$$Adv_{\overline{SE},A}^{\mathrm{int}-ptxt}(k) \le Adv_{M,A_p}^{wuf-cma}(k)$$

$$Adversary\ \mathrm{F}_{\mathrm{p}}^{\mathrm{T}_{K_m}(\cdot),V_{K_m}(.,.)}(\kappa)$$

$$\mathrm{k_e} \xleftarrow{R} K_e(\kappa)$$

$$\text{For i} = 1,\ldots.,\mathrm{q_e}+\mathrm{q_d}\ \text{do}$$

When A makes a query $\mathrm{M_i}$ to its encryption oracle do

$$\mathrm{C_i} \leftarrow E_{K_e}(\mathrm{M_i}); \tau_i \leftarrow \mathrm{T}_{K_m}(C_i^{'}); A \Leftarrow C_i^i \,\|\, \tau_i$$

When A makes a query $\mathrm{C_i}$ to its verification oracle do

$$\text{Parse } \mathrm{C_i} \text{ as } C_i^i \,\|\, \tau_i^i; v_i \leftarrow V_{K_m}(C_i^{'},\tau_i^{'}); A \Leftarrow v_i$$

Proposition 4.9

- Encrypt-then-MAC method with a SUF-CMA-secure MAC is INT-CTXT, IND-CPA, and IND-CCA

$$Adv_{\overline{SE},A}^{\mathrm{int}-ctxt}(k) \leq Adv_{MA,F}^{suf-cma}(k)$$

$$Adv_{\overline{SE}}^{ind-cpa}(k,t,q,\mu) \leq Adv_{SE}^{ind-cpa}(k,t,q,\mu)$$

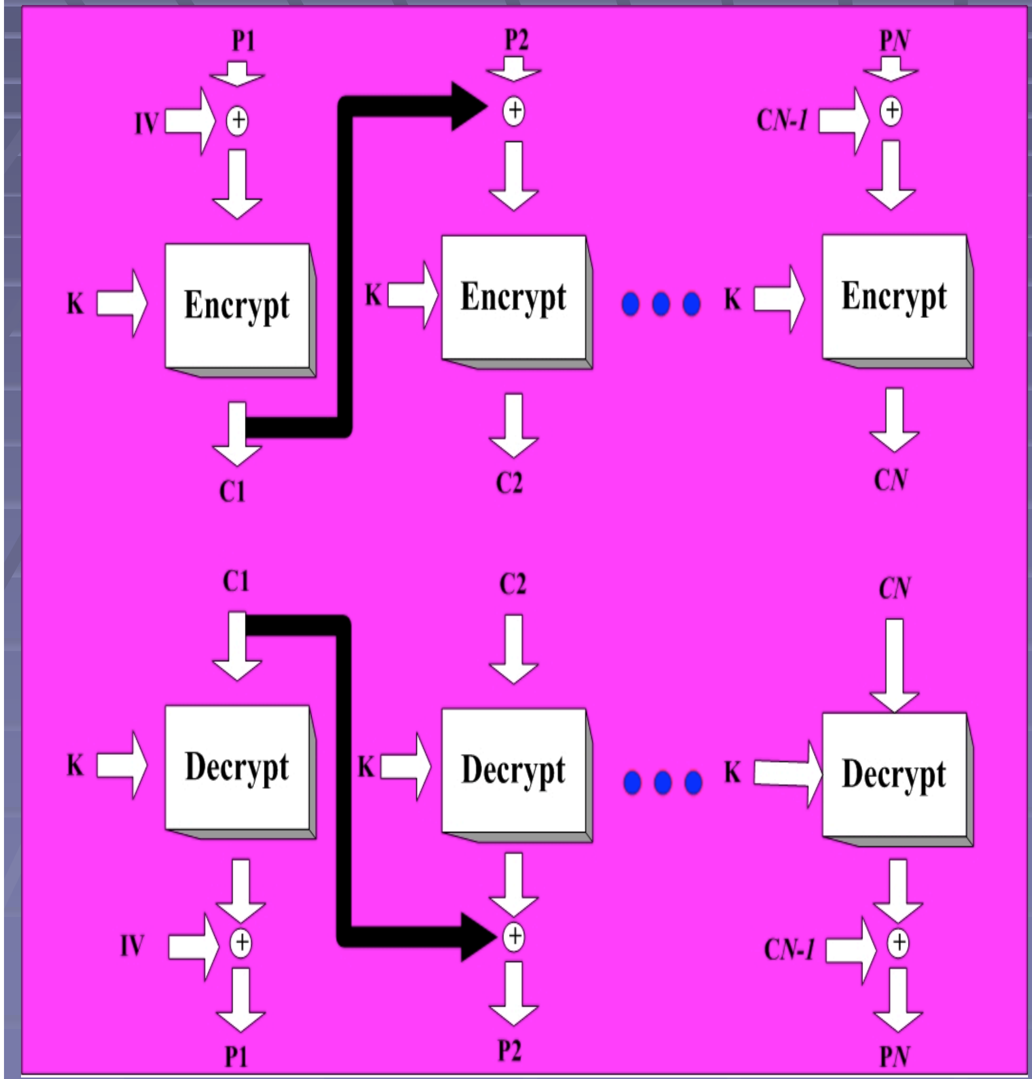
$$Adv_{\overline{SE}}^{\mathrm{int}-ctxt}(k,t,q_e,q_d,\mu_e,\mu_d) \leq Adv_{MA}^{suf-cma}(k,t,q_e,q_d,\mu_e+q_e l,\mu_d)$$

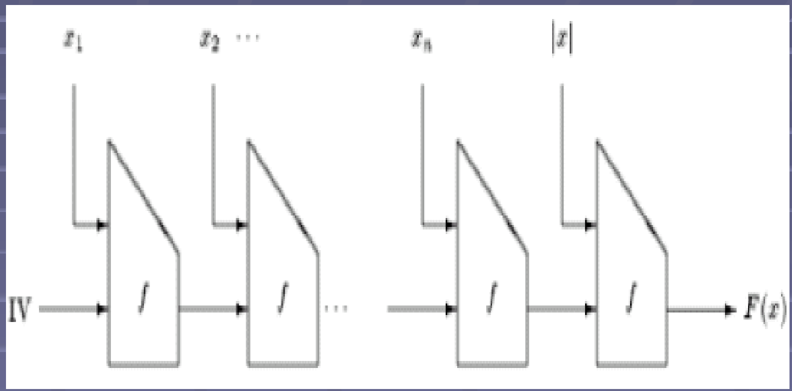
$$Adv_{\overline{SE}}^{ind-cca}(k,t,q_e,q_d,\mu_e,\mu_d) \leq 2\times Adv_{MA}^{suf-cma}(k,t,q_e,q_d,\mu_e+q_e l,\mu_d) + Adv_{SE}^{ind-cpa}(k,t,q_e,\mu_e)$$

Conclusion

Encrypt-then-MAC provides the most secure solution for authenticated encryption

CBC – Cipher Block Chain





•If IV is different then instances of same msg (or block) will be encrypted differently

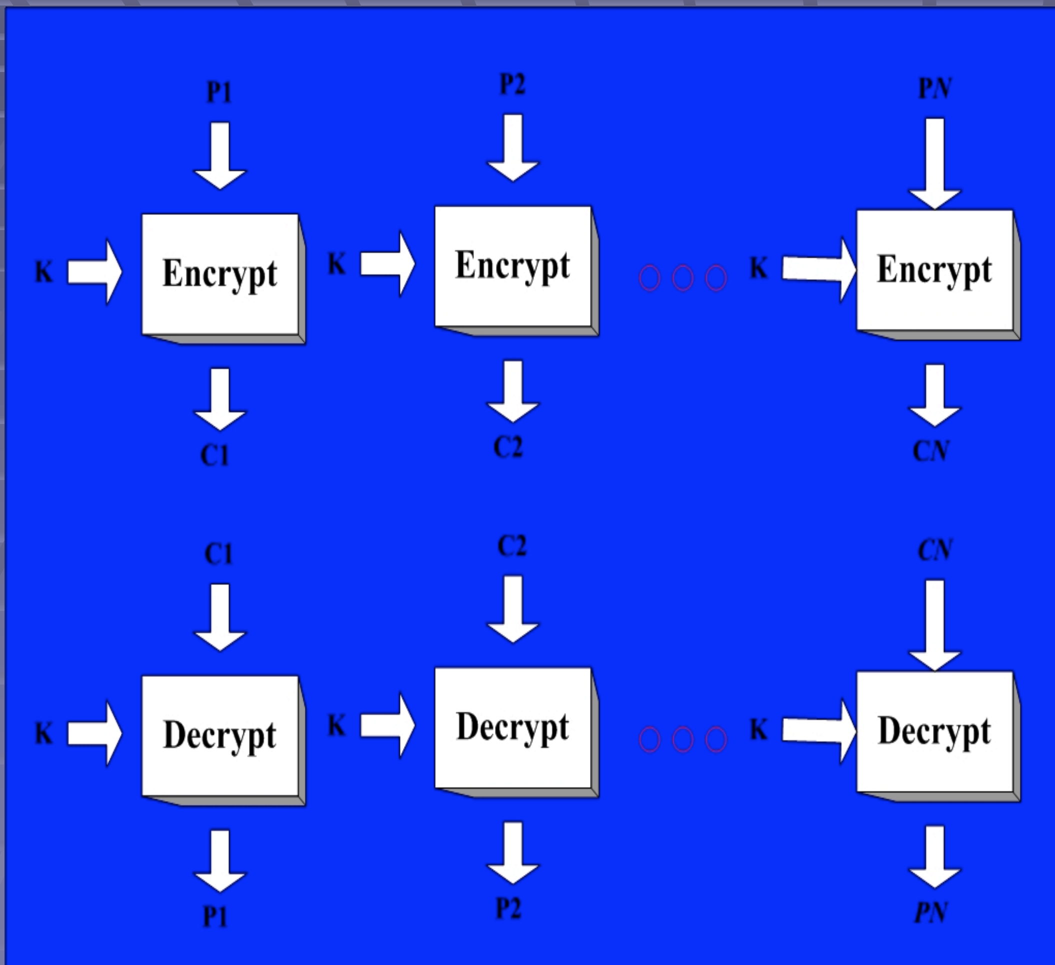
•If K'th cipher block C_k gets corrupted in transmission – only blocks P_k and P_{k+1} are affected

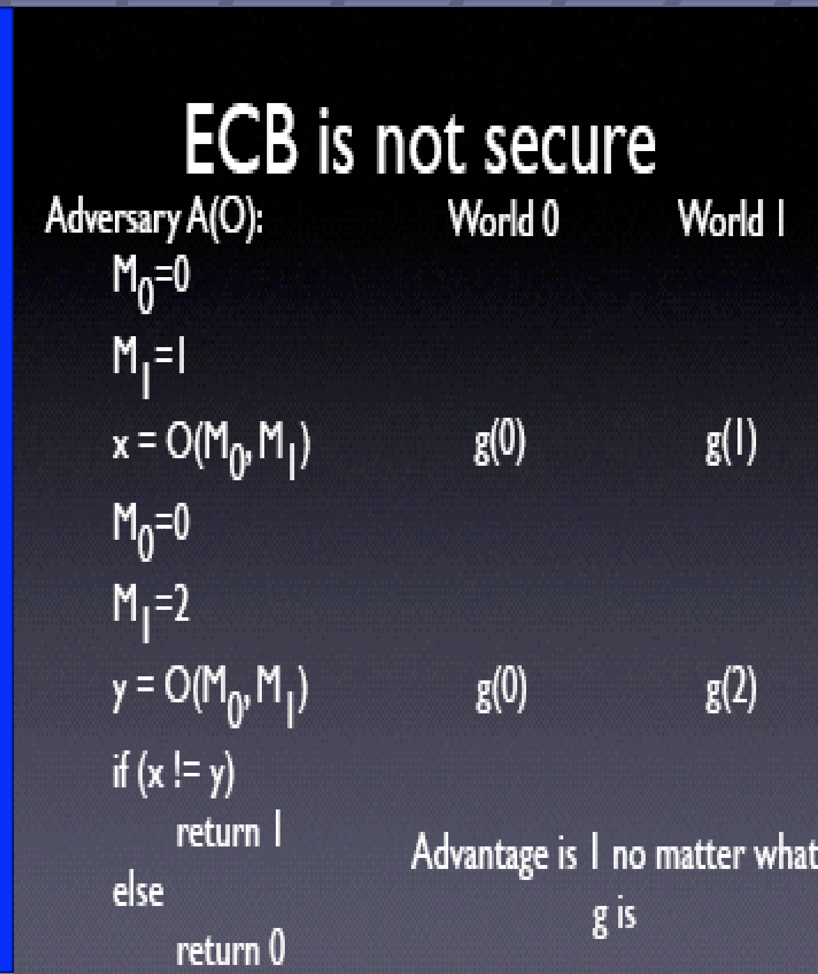
  •This can also allow some msg tampering

•If one plaintext block P_k is changed – All subsequent ciphertext blocks will be affected

  •This leads to an effective MAC

ECB – Electronic Code Book





If the same key is used then identical plaintext blocks map to identical ciphertext

Proposition 4.1

- Encrypt-and MAC method is not IND-CPA

Proposition 4.2

- Encrypt-and MAC method is IND-CPA insecure for any deterministic MAC)

Theorem 4.3

- Encrypt-and-MAC is INT-PTXT secure

Proposition 4.4

- Encrypt-and-MAC method is not INT-CTXT secure

Theorem 4.5

- MAC-then-encrypt method is both INT-PTXT an IND-CPA secure

Proposition 4.6

- MAC-then-encrypt method is not NM-CPA secure

Proposition 4.8

- Encrypt-then-MAC method with a WUF-CMA-secure MAC is not NM-CPA secure