Security Notions for Symmetric Cryptography

15-859I
Spring 2003

Motivation

- Recall: A cryptosystem is said to satisfy IND-CPA if for every polynomial time adversary A,

  $\Pr[A^{LR(0,..,)}(1^k) = 1] - \Pr[A^{LR(1,..,)}(1^k) = 1]$

  is negligible. This definition implies that for any plaintext distribution, ciphertexts yield no efficiently computable information about their corresponding plaintexts.
- Is this always a sufficient notion of secrecy?

Other attacks

- Suppose Eve can get Alice or Bob to decrypt some made-up ciphertexts. Does IND-CPA imply that Eve still learns no information about other ciphertexts?
- Suppose Eve wants to modify Alice's message to Bob such that if the original plaintext says "ATTACK AT DAWN", the decryption of the modified ciphertext says "RETREAT AT DAWN." and vice-versa. Does IND-CPA prevent such an attack?

Katz and Yung's approach

- [KY99] consider 18 different security notions:
  - Two types of oracles: encryption (P) and decryption (C)
  - Three types of access: none (0), nonadaptive (1), adaptive (2)
  - Two goals: indistinguishability (IND), nonmalleability (NM)
- For each pair of notions (A,B) determine if:
- A is equivalent to B,
- A is strictly stronger (weaker) than B, or
- A and B are *incomparable*.

Equivalence between notions

- Security conditions A and B are equivalent if
- For every symmetric encryption scheme $\mathcal{SE}$ satisfying A, $\mathcal{SE}$ also satisfies B.
- For every symmetric encryption scheme $\mathcal{SE}$ satisfying B, $\mathcal{SE}$ also satisfies A.
- Established in the contrapositive, that is, assume $\mathcal{SE}$ satisfies A, and show how an attacker violating B violates A.

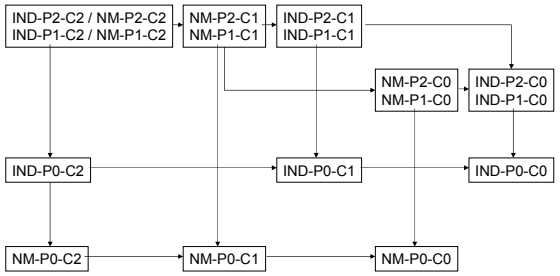
Strictly stronger relationship

- Notion A is said to be *strictly stronger* than notion B if:
- Every symmetric encryption scheme $\mathcal{SE}$ which satisfies A also satisfies B.
  - Also established in the contrapositive…
- BUT, if there is a scheme which satisfies B there exists a scheme which satisfies B but not A.

Incomparable notions

- A and B are said to be incomparable if:
- If there exists a symmetric scheme $\mathcal{SE}$ which satisifies A then there is one which satisfies A but not B, AND
- If there exists a scheme satisfying B, then there is one which satsifies B but not A.
- This is established by exhibiting a scheme which satisfies A and giving an attack in the sense of B.

Hierarchy



IND-PX-CY: Advantage

Let A = (A_1,A_2) be an adversary. Define

$$\mathrm{Adv}^{\text{IND-PX-CY}}_{A,\Pi}(k) =$$

$$2 \cdot \Pr\left[\begin{array}{c} A^{E_2,D_2}(x_0,x_1,s,y) = b : \\ sk \leftarrow K(1^k); (x_0,x_1,s) \leftarrow A_1^{E_1,D_1}(1^k); \\ b \leftarrow \{0,1\}; y \leftarrow E_{sk}(x_b) \end{array}\right] - 1$$

If X=0 then E_1="", and E_2="" If Y=0 then D_1="", and D_2=""

If X=1 then $E_1=E_{sk}$ and E_2="" If Y=1 then $D_1=D_{sk}$ and D_2=""

If X=2 then $E_1=E_2=E_{sk}$ If Y=2 then $D_1=D_2=D_{sk}$

IND-PX-CY definition

- Say that Protocol Π is IND-PX-CY if for all polynomial time A, there exists a negligible function μ such that

$$\mathrm{Adv}^{\text{IND-PX-CY}}_{A,\Pi}(k) \le \mu(k)$$

NM-PX-CY: Success probability

Let A=(A_1,A_2) be an adversary, define

$$\mathrm{Succ}^{\text{NM-PX-CY}}_{A,\Pi}(k) =$$

$$\Pr\left[\begin{array}{c} y \notin \vec{y} \wedge \perp \notin \vec{x} \wedge R(x,\vec{x}) : \\ sk \leftarrow K(1^k); (M,s) \leftarrow A_1^{E_1,D_1}(1^k); x \leftarrow M; y \leftarrow E_{sk}(x) \\ (R,\vec{y}) \leftarrow A_2^{E_2,D_2}(M,s,y); \vec{x} = D_{sk}(\vec{y}) \end{array}\right]$$

Where E_1,E_2,D_1,D_2 depend on X,Y as previously

NM-PX-CY: Random probability

Let A=(A_1,A_2) be an adversary, define:

$$\mathrm{Succ}^{\text{NM-PX-CY}}_{A,\Pi,\$}(k) =$$

$$\Pr\left[\begin{array}{c} y \notin \vec{y} \wedge \perp \notin \vec{x} \wedge R(\tilde{x},\vec{x}) : \\ sk \leftarrow K(1^k); (M,s) \leftarrow A_1^{E_1,D_1}(1^k); x \leftarrow M; y \leftarrow E_{sk}(x) \\ (R,\vec{y}) \leftarrow A_2^{E_2,D_2}(M,s,y); \vec{x} = D_{sk}(\vec{y}); \tilde{x} \leftarrow M \end{array}\right]$$

Where E_1,E_2,D_1,D_2 depend on X,Y as previously

NM-PX-CY: Advantage and security

Define the advantage of A against Π by

$$\mathrm{Adv}^{\text{NM-PX-CY}}_{A,\Pi}(k) = \mathrm{Succ}^{\text{NM-PX-CY}}_{A,\Pi}(k) - \mathrm{Succ}^{\text{NM-PX-CY}}_{A,\Pi,\$}(k)$$

And say that Π is NM-PX-CY secure if for every A which produces a polynomial-time samplable M and a polynomial-time computable R, there is a negligible function μ such that

$$\mathrm{Adv}^{\text{NM-PX-CY}}_{A,\Pi}(k) \le \mu(k)$$

IND-P2-CY ⇔ IND-P1-CY

Suppose Π is IND-P1-CY. Let B be an IND-P2-CY adversary. Write B = (B_1,(B_2,B_3)) such that B_3 makes no encryption queries. WLOG suppose B_2 makes exactly n(k)=poly(k) encryption queries and never queries a decryption oracle with the result of an encryption query.

Key idea: B_2 can't distinguish between encryptions of its queries and encryptions of some other plaintext.

IND-P1-CY ⇒ IND-P2-CY

Define $p_k(d \mid b,i,c) =$

$\Pr[B_3^{D_2}(\vec{q},\vec{a},s'') = d :$

$sk \leftarrow K(1^k); (x_0,x_1,s) \leftarrow B_1^{E_{sk},D_1}(1^k); y \leftarrow E_{sk}(x_b);$

$((q_1,a_1),\ldots,(q_i,a_i),s') \leftarrow B_2^{E_{sk},D_2}(y,s;i);$

$a_{i+j} \leftarrow B_2^{E_{sk},D_2}(E_{sk}(x_c),s;i+j), \quad 1 \le j \le n-i;$

$(\vec{q},s'') \leftarrow B_2^{D_2}(y,s \mid q_1,\ldots,q_i,\vec{a},s')]$

$$\mathrm{Adv}^{\text{IND-P2-CY}}_{B,\Pi}(k) = p_k(1:b=1,i=n,c=0) - p_k(1:b=0,i=n,c=0)$$

Hybrid argument for IND-P1-CY⇒IND-P2-CY

- We will give three types of Hybrid adversaries:
- A_r, 1≤r≤n: A_r will have advantage $p_k(1:1,r,0) - p_k(1:1,r-1,0)$
- D_r, 1≤r≤n: D_r will have advantage $p_k(1:0,r-1,0) - p_k(1:0,r,0)$
- C: C will have advantage $p_k(1:1,0,0) - p_k(1:0,0,0)$
- So we will have B's IND-P2-CY advantage equal to the sum of the Hybrids' IND-P1-CY advantages.
- So if B's advantage is non-negligible, then some hybrid has non-negligible IND-P1-CY advantage, a contradiction, QED.

Hybrids: A_r

- A_r in stage 1:
  - set $(x_0,x_1,s) \leftarrow B_1^{E,D}(1^k)$
  - Draw $((q_1,a_1),\ldots,(q_r,a_r),s') \leftarrow B_2^{E,D}(y_1=E(x_1),s;r)$
  - set $x_1' = q_r$
  - draw $x_0' \leftarrow B_2^{E,D}(E(x_0),s;r)$
  - Draw $(q_i, a_i) \leftarrow B_2^{E,D}(E(x_0),s;i)$, $r < i \le n$
  - Return $(x_0',x_1',s'') = (s',a_1,\ldots,a_n,q_1,\ldots,q_r))$
- A_r in stage 2:
  - a_r=y (Either $E(x_0')$ or $E(x_1')$)
  - $(q_1,\ldots q_n,s''') \leftarrow B_2^D(y_1,s \mid q_1 \ldots q_r,a_1,\ldots,a_n,s')$
  - Return $B_3^D(q,a,s''')$

Hybrids: D_r

- D_r in stage 1:
  - draw $(x_0, x_1, s) \leftarrow B_1^{E,D}(1^k)$; draw $y_0 \leftarrow E(x_0)$
  - draw $((q_1,a_1),\ldots,(q_r,a_r),s') \leftarrow B_2^{E,D}(y_0,s;r)$
  - set $x_0' = q_r$; draw $x_1' \leftarrow B_2^{E,D}(E(x_0),s;r)$
  - draw $(q_i,a_i) \leftarrow B_2^{E,D}(E(x_0),s;i)$, $r < i \le n$
  - Return $(x_0',x_1',s'') = (q_1,\ldots,q_r,a_1,\ldots,a_n,s'))$
- D_r in stage 2:
  - a_r=y
  - $(q_1,\ldots,q_n,s''') \leftarrow B_2^D(y_0,s:q_1 \ldots q_r,a_1,\ldots,a_n,s')$
  - Return $B_3^D(q,a,s''')$

Hybrids: C

- C in stage 1:
  - $(x_0,x_1,s) \leftarrow B_1^{E,D}(1^k)$
  - $(q_i,a_i) \leftarrow B_2^{E,D}(E(x_0),s;i)$, $1 \le i \le n$
  - Return $(x_0,x_1,s' = (s,a_1,\ldots a_n)$
- C in stage 2:
  - $(q,s'') \leftarrow B_2^{D}(y,s|a_1,\ldots,a_n)$
  - Return $B_3^{D}(q,a,s'')$

NM-P2-CY ⇔ NM-P1-CY

- We define a slightly different type of decryption oracle, C'Y.
- We will show that NM-P1-CY ⇒ IND-P1-C'Y
- And IND-P2-C'Y ⇒ NM-P2-CY
- The results for IND-PX-CY carry over to IND-PX-C'Y, giving IND-P1-C'Y⇒IND-P2-C'Y
- Thus we get NM-P1-CY ⇒ NM-P2-CY

IND-PX-C'Y

Let A =(A_1,A_2,A_3), and define

$$\mathrm{Adv}_{A,\Pi}^{\text{IND-PX-C'Y}}(k) = \left|\mathrm{Succ}_{A,\Pi}^{\text{IND-PX-C'Y}}(k) - \mathrm{Fail}_{A,\Pi}^{\text{IND-PX-C'Y}}(k)\right| ,$$

$$\mathrm{Succ}_{A,\Pi}^{\text{IND-PX-C'Y}}(k) = \Pr[A_3(\vec{p},s'') = b \wedge \perp \notin \vec{p} : sk \leftarrow K(1^k);$$
$$(x_0,x_1,s) \leftarrow A^{E_1,D_1}(1^k); b \leftarrow \{0,1\}; y \leftarrow E_{sk}(x_b);$$
$$(\vec{c},s'') \leftarrow A_2(x_0,x_1,y,s); \vec{p} = D_{sk}(\vec{c})]$$

$$\mathrm{Fail}_{A,\Pi}^{\text{IND-PX-C'Y}}(k) = \Pr[A_3(\vec{p},s'') = 1 - b \wedge \perp \notin \vec{p} : sk \leftarrow K(1^k);$$
$$(x_0,x_1,s) \leftarrow A^{E_1,D_1}(1^k); b \leftarrow \{0,1\}; y \leftarrow E_{sk}(x_b);$$
$$(\vec{c},s'') \leftarrow A_2(x_0,x_1,y,s); \vec{p} = D_{sk}(\vec{c})]$$

NM-P1-CY⇒IND-P1-C'Y

- Let $B=(B_1,B_2,B_3)$ be a IND-P1-C'Y adversary. Construct NM-P1-CY adversary $A=(A_1,A_2)$:
- $A_1^{E,D}(1^k)$:
  - $(x_0,x_1,s) \leftarrow B_1^{E,D}(1^k)$; $M = \{x_0,x_1\}$
  - Choose $x' \notin M$; $y' \leftarrow E_{sk}(x')$
  - Return $(M,s'=(y',s))$
- $A_2^{D}(M,s',y)$:
  - $(\mathbf{c},s'') \leftarrow B_2^{D}(x_0,x_1,s,y)$, choose $\sigma \leftarrow \{0,1\}^k$
  - Return $(R_{M,s'',\sigma},(y',\mathbf{c}))$,
    Where $R_{M,s'',\sigma}(x,\mathbf{x})$ =
    parse $(x',\mathbf{p}) = \mathbf{x}$,
    return 1 iff $x = x_b$ and $B_3(\mathbf{p},s'';\sigma)=b$

IND-P2-C'Y⇒NM-P2-CY

- Let $B=(B_1,B_2)$ be NM-P2-CY adversary.
- $A_1^{E,D}(1^k)$:
  - $(M,s) \leftarrow B_1^{E,D}(1^k)$; $x_0,x_1 \leftarrow M$
  - Return $(x_0,x_1,s'=(M,s))$
- $A_2^{E,D}(x_0,x_1,s',y)$:
  - $(R,\mathbf{y}) \leftarrow B_2^{E,D}(M,s,y)$
  - Return $(\mathbf{y},s''=(R,y,\mathbf{y}))$
- $A_3(\mathbf{p},s'')$:
  - if $y \notin \mathbf{y}$ and $\perp \notin \mathbf{p}$ and $R(x_0,\mathbf{p})$ then b = 0; Else $b \leftarrow \{0,1\}$
  - Return b

IND-P0-C2 ⇏ IND-P1-C0

- Let $\Pi=(K,E,D)$ satisfy IND-P0-C2
- Define P=(K',E',D') where
  - $K'(1^k) = sk \leftarrow K(1^k); v \leftarrow \{0,1\}^k; y \leftarrow E_{sk}(v)$; return (sk,v,y)
  - $E'_{sk,v,y}(x) =$ if x=v then (y,v) else $(E_{sk}(x),v)$
  - $D'_{sk,v,y}(y',v') =$ if v=v' then $D_{sk}(y')$ else $\perp$
- Claim 1: P is not IND-P1-C0 secure.
  - A_1: Choose any x, request E(x) to get v, request E(v) to get y. Set $x_0=v, x_1 \leftarrow \{0,1\}^k$, s=y
  - $A_2(y',v',s=y)$ = if y'=y then 0, else 1

IND-P0-C2 ⇏ IND-P1-C0

- Let Π=(K,E,D) satisfy IND-P0-C2
- Define P=(K',E',D') where
  - $K'(1^k)$ = sk←$K(1^k)$;v←$\{0,1\}^k$;y←$E_{sk}(v)$; return (sk,v,y)
  - $E'_{sk,v,y}(x)$ = if x=v then (y,v) else $(E_{sk}(x),v)$
  - $D'_{sk,v,y}(y',v')$ = if v=v' then $D_{sk}(y')$ else $\perp$
- Claim 2: P is IND-P0-C2 secure.
  - Given A decryption oracle for Π, we can simulate a decryption oracle for P by choosing v ←$\{0,1\}^k$.
  - Since there is no encryption oracle, the probability of making a decryption query with v is at most $1/2^k$

NM-P0-C2 ⇏ IND-P0-C0

- Let Π=(K,E,D) satisfy NM-P0-C2
- If there's no IND-P0-C0 scheme, we're good.
- Else there is, and then there's a PRF F.
- Define P=(K',E',D') where
  - $K'(1^k)$ = sk←$\{0,1\}^k$
  - $E'_{sk}(x)$ = x,$F_{sk}(x)$
  - $D'_{sk}(y,z)$ = if $F_{sk}(y)$=z then y else $\perp$
- Claim 1: P is not IND-P0-C0 secure.
- Claim 2: P is NM-P0-C2 secure.

NM-P0-C2 ⇏ IND-P0-C0

- Define P=(K',E',D') where
  - $K'(1^k)$ = sk←$\{0,1\}^k$
  - $E'_{sk}(x)$ = x,$F_{sk}(x)$
  - $D'_{sk}(y,z)$ = if $F_{sk}(y)$=z then y else $\perp$
- Claim 2: P is NM-P0-C2 secure.
  - Suppose (B_1,B_2) is a NM-P0-C2 adversary
  - $D^f(1^k)$:
    - (M,s) ← $B_1^D(1^k)$; x←M; y = x,f(x)
    - (R,z)←B_2^D(M,s,y); (x',y') = z_1;
    - if y' = f(x') and x≠x' return 1 else return 0

NM-P0-C2 ⇏ IND-P0-C0

- Define P=(K',E',D') where
  - $K'(1^k)$ = sk←$\{0,1\}^k$; $E'_{sk}(x)$ = x,$F_{sk}(x)$; $D'_{sk}(y,z)$ = if $F_{sk}(y)$=z then y else $\perp$
- Claim 2: P is NM-P0-C2 secure.
  - $D^f(1^k)$:
    - (M,s) ← $B_1^D(1^k)$; x←M; y = x,f(x)
    - (R,z)←B_2^D(M,s,y); (x',y') = z_1;
    - if y' = f(x') and x≠x' return 1 else return 0
  - $Adv_{B,\Pi}^{NM\text{-}P0\text{-}C2}(k) \le Pr[D^F(1^k)=1]$
  - $Pr[D^{rf}(1^k)=1] \le 1/(2^k\text{-}q(k))$
  - $Adv_{B,\Pi}^{NM\text{-}P0\text{-}C2}(k) \le Insec_F^{prf}(t,q(k),k) + 1/(2^k\text{-}q(k))$

NM-PX-CY ⇒IND-PX-CY, X∈{1,2}

- Let (B_1,B_2) be a IND-PX-CY adversary. Construct NM-PX-CY adversary (A_1,A_2):
- $A_1^{E,D}(1^k)$:
  - (x_0,x_1,s) ←$B_1^{E,D}(1^k)$; M = $\{x_0,x_1\}$;y_i = $E(x_i)$
  - Return (M,s'=(s,y_0,y_1))
- $A_2^{E,D}(y,s')$:
  - c ←$B_2^{E,D}(y,s)$
  - Return (R,y_c),
    where $R(x_a,x_b)$ = 1 iff a=1-b

IND-PX-C2 ⇒ NM-PX-C2

- Notice that the proof that IND-P2-C'Y ⇒ NM-P2-CY also works if we have a C2 Oracle, regardless of the PX oracle required by the NM-PX-C2 adversary.
- So we have that IND-PX-C2 ⇒ NM-PX-C2

IND-P2-C1 ⇏ NM-P0-C0

- Let Π=(K,E,D) by IND-P2-C1 secure.
- Define P = (K',E',D'):
  - $K'(1^k) = K(1^k)$
  - $E'_{sk}(x) = y \leftarrow E_{sk}(x)$, $b \leftarrow \{0,1\}$, return y,b
  - $D'_{sk}(y_1,b) = D_{sk}(y_1)$
- Note that P is malleable: given $y=E_{sk}(x)$ we can produce $y' \neq y$ such that $D_{sk}(y') = D_{sk}(y)$.
- P is still IND-P2-C1 secure: we can simulate a E' oracle given an E oracle.

NM-P2-C0 ⇏ IND-P0-C1 ∨ NM-P0-C1

- Let Π=(K,E,D) by NM-P2-C0 secure.
- Define P = (K',E',D'):
  - $K'(1^k) = sk \leftarrow K(1^k)$, $v \leftarrow \{0,1\}^k$; return (sk,v)
  - $E'_{sk,v}(x) = (0, E_{sk}(x))$
  - $D'_{sk,v}(b,y) =$ if (b=0) then $D_{sk}(y)$
    Else if $y = 1^k$ then v
    Else if y=v then sk; else ⊥
- Claim 1: P is NM-P2-C0 secure.
- Proof: Can emulate a E'_{sk} oracle given an E oracle. Don't have a D oracle to learn v, so at most $1/2^k$ increase in advantage.

NM-P2-C0 ⇏ IND-P0-C1 ∨ NM-P0-C1

- Let Π=(K,E,D) by NM-P2-C0 secure.
- Define P = (K',E',D'):
  - $K'(1^k) = sk \leftarrow K(1^k)$, $v \leftarrow \{0,1\}^k$; return (sk,v)
  - $E'_{sk,v}(x) = (0, E_{sk}(x))$
  - $D'_{sk,v}(b,y) =$ if (b=0) then $D_{sk}(y)$
    Else if $y = 1^k$ then v
    Else if y=v then sk; else ⊥
- Claim 2: P is {IND,NM}-P0-C1 insecure.
- Proof: Can get sk with 2 queries to D'.

NM-P2-C1 ⇏ NM-P0-C2

- Π=(K,E,D) is NM-P2-C1 secure, and F a PRF.
- Define P = (K',E',D'):
  - $K'(1^k) = sk \leftarrow K(1^k)$, $K \leftarrow \{0,1\}^k$; return (sk,K)
  - $E'_{sk}(x) = y \leftarrow E_{sk}(x)$; return (0,y,"")
  - $D'_{sk,K}(b,y,z) =$ if (b=0 AND z="") then $D_{sk}(y)$;
    Else if (b=1) and (z="") then $F_K(y)$
    Else if (b=1) and ($z=F_K(y)$) then $D_{sk}(y)$
    Else ⊥
- P is malleable given C2 access: given ciphertext 0,E(x),"" we can produce ciphertext 1,E(x),F_K(E(x))
- But a C1 adversary can't predict $F_K(E(x))$. (roughly)

Hierarchy