આપણા સૌજન્ય દાતાશ્રીઓ - ૨૦૨૧

- ઓરિએન્ટ ઓટો પાર્ટસ (ડીલર્સ ઇન ઇન્ડીકા - ઇન્ડીગો - ક્વોલીસ - મારૂતિ - લેન્સર - સુમો સ્પેર્સ) સહયોગ બિલ્ડીંગ દિનબાઈ ટાવર સામે, મીરજાપુર-અમદાવાદ-૧. કમલેશ ભીખુભાઈ સી. ચોક્સી - પૌલીન કે. ચોક્સી (સાંથલ) ફોન : ૨૫૫૦૨૬૬૭, ૨૫૫૦૪૪૪૯
- સ્વ. શ્રી પોપટલાલ સ્વયંદભાઈ શાહ (લિંબોડા)
- સ્વ. શ્રી ભોગીલાલ દોલચંદ શાહ (ખરણા) (માનદ્ મંત્રી-શ્રેયસ્કર મંડળ-અમદાવાદ)
- પ્રફુલભાઈ દોલચંદ શાહ (ખરણા) (માનદ્ મંત્રી-શ્રેયસ્કર મંડળ-અમદાવાદ) એ-૧૨, અભિનંદન એપાર્ટમેન્ટ, સંત કબીર સ્કૂલ સામે, નારણપુરા, અમદાવાદ-૧૩. મિલન કેટરર્સ - લગ્ન,શુભ પ્રસંગે તથા દરેક ધાર્મિક પ્રસંગે જમણવારનું કેટરીંગ કરી આપનાર ફોન : (રહે.) ૨૭૯૧૩૫૫ (મો.) ૯૮૨૫૦૧૪૫૪૭, (પ્રફુલ શાહ - આશા પી. શાહ)
- મહાવીર ઓપ્ટીશીયન એન્ડ કોન્ટેક લેન્સ ક્લીનીક પારસનગર સોલારોડ, નારણપુરા, અમદાવાદ-૬૩. (ફોન : ૨૭૪૮૩૪૩૫)
- રાજુભાઈ શાહ, પ્રફુલભાઈ વી. શાહ (દેકાવાડા) (મો. ૯૮૨૫૯૭૫૭૫૫)
- સુનિલભાઈ રસિકલાલ શાહ (ધામણવા) વિધિ માર્કેટીંગ (ટેક્ષટાઈલ એન્જીનીયર) ડૉ. પ્રિન્સી (B.P.T.) સ્તવન આઈ ક્લિનિક ડૉ. પાર્થ M.S. Opth. Gold Medalist લેબીક્સ ડાયગ્નોસ્ટીક - ડૉ. પ્રેક્ષા (મો. ૯૯૦૯૯ ૭૨૪૨૪)
- સ્વ. અમૃતલાલ કેશવલાલ શાહ પરિવાર (માણસા) - કીર્તિભાઈ - મંદાબેન, પૂજન - પૂજા - માનુષ. ફોન નં (રહે) ૨૬૭૫૦૧૮૬, ૪૮૦૦૬૦૬૫, (મો.) ૯૮૨૫૦૩૪૦૬૫
- શ્રીમતી ઈન્દુબેન ગુણવંતલાલ ચુનીલાલ શાહ (ઝુલાસણ) ફોન : (રહે.) ૨૭૪૭૦૬૧૬ (કાર્તિક-કલ્પના) (મલય-પિન્કી) (હેલી - સર્મપણ - કલગી) મલયભાઈ મો. ૯૮૨૫૦૬૬૨૧૦
- શ્રીમતી સરોજબેન ખુશાલભાઈ બબલદાસ શાહ (કૈયલ) ફોન : (રહે.) ૨૬૫૬૧૪૦૯ (વિપુલભાઈ - નયનાબેન-પુનિત-ખુશ્બુ-આંશી, ડૉ. મીરા) (નયનભાઈ - મોનાબેન - હર્ષ - હિમાની)
- શ્રીમતી ઈન્દુમતીબેન જશવંતલાલ ગીરધરલાલ શાહ (ઈન્દ્રોડા) ડી/૫૬, આર્યમાન બંગ્લોઝ, શીલજ રેલવે કોસીંગ પાસે, થલતેજ, અમદાવાદ-૫૯. જોલિત શાહ (રહે.) ૪૦૩૯૯૭૯૫, અતુલ શાહ (મો.) ૯૮૨૫૦૩૮૦૮૮, જશુભાઈ (મો.) ૯૭૨૭૭૪૯૭૯૫
- શ્રીમતી મંજુલાબેન અમૃતલાલ ચીમનલાલ શાહ (રાજપુર) ડૉ. દેવેન્દ્રભાઈ (M.D.PED.) મનીષભાઈ (ક્લોથ મરચન્ટ) મૃગેશભાઈ (બી.ઈ.સીવીલ) ફોન : (ઓ.) ૩૦૨૨૨૫૫૧ (રહે.) ૨૭૫૦૦૯૮૩
- શ્રીમતી મંજુલાબેન વિનોદચંદ્ર માણેકલાલ શાહ (જોટાણા) (શોધન - દેવીયાની) (સમીર - નિલમ) (શિવાની - ટ્વીંકલ) ૧૪, શાકુંતલ સોસાયટી, ઉસ્માનપુરા, અમદાવાદ-૧૪. રહે. ૨૭૫૫૧૯૮૯
- પ્રવિણકુમાર દોલચંદભાઈ શાહ (ખરણા) પકીનભાઈ (મો.) ૯૯૭૯૯૬૩૬૬૩ સૌમિલ પ્રવિણકુમાર શાહ (રહે.) ૨૬૪૪૪૩૨૯ (મો.) ૯૮૨૫૩૮૧૨૩૪ (એલ.આઈ.સી. તથા જનરલ ઈન્સ્યુરન્સ એજન્ટ) અરૂણા - કૃતિ - પ્રિયાંશી - અનુપ - નીમિતા - નીહલ (યુ.કે.)
- લીલાવતી બેન સેવંતીલાલ દેવચંદભાઈ શાહ પરિવાર (ગુંજાલા) બેચરાજી જગદીશભાઈ - સંધ્યાબેન, હર્ષ - પ્રેક્ષા (FRANCE), વિવાન, હિદાન, અમી (CFP) -ગૌરાંગ. ફોનઃ(રહે.) ૨૬૪૦૮૬૩૭, મો. ૯૯૭૮૪૦૭૬૦૬
- શ્રીમતી શીલાબેન કમલેશકુમાર શાહ (કટોસણ) મો. ૯૮૨૪૦૧૫૯૯૧, ફોન : ૨૬૪૬૩૧૨૫ એચ. આર. ઓટો પાર્ટસ (હેમલ કે. શાહ) મો. ૯૮૯૮૬૧૭૨૩૬ - શૈલેશભાઈ એફ. શાહ જી/૩, પ્રથમ માળે, મેઘાલય સોસાયટી, અંજલી સાડી સોરૂમ સામે, નવરંગપુરા, અમદાવાદ-૧૪.
- ગજરાબેન રતિલાલ શાહ પરિવાર (આખજ) (રહે.) ૨૫૫૦૮૧૫૨/૫૩ હસ્તે - જશવંતભાઈ, જયંતિભાઈ, રજનીભાઈ, અશોકભાઈ, વિનોદભાઈ, નવીનભાઈ
- શ્રીમતી હંસાબેન પંકજકુમાર શાહ (સાંથલ) (મો.) ૯૪૨૭૦૨૮૩૫૭ મિલન પંકજભાઈ શાહ (સી.એ) મીતા મિલન શાહ (સી.એ.), હેઝલ મિલન શાહ B/304, સીગ્નોર રેસીડન્સી, ધનંજય ટાવર પાસે, ૧૦૦ ફૂટ રોડ, સેટેલાઈટ, અમદાવાદ-૧૫

PAGE NO.
DATE

PRINCIPLES OF INFORMATION SECURITY - Kannan

→ Day 1 → missed (3rd Jan)

6th Jan

→ Past : Art of Secure Communication. } most of the course.
→ Present : Science of Information Securing
→ Future : Science of the impossible.

③ Art → Science. (Explicit transformation)
① One Problem → Everywhere. (Semi explicit)
② All tricks solving → One technique for (Not explicit) nothing solving anything at all)

① Captivates layman
② Captivates designers / engineers, etc.
③ Captivates Scientists.

⇒ Caesar Cipher :
→ Shift alphabet by 3.

a → D
b → E
z → C

crypto
⇓
fubswr
FUBSWR

(Satisfying) is mandatory)

⇒ Kerckhoff's Principle (19th century)
→ Security of a system mustn't depend on the obscurity of the algorithm, it must solely depend on the secrecy of the key.

① Secure memory is costly.
② Reverse Engineering
③ Updation / ~~Recreation~~ Revocation with ease.
④ Standardization (Internet Standards)
⑤ Scalability.
⑥ Ethical Hacking.

⇒ Shift Cipher:
Key $K \in [0, 25]$
Enc $c = m + K \pmod{26}$
If $K = 4$
↳ Crypto → GVCTXS.

⇒ Principle of Large Keyspace:
(people need to find large keyspace)
Else: Brute force attack works.

⇒ Mono alphabetic Substitution
a b c ... z
↓ ↓ ↓ ↓
X A D Y
Any permutation is a key

|Keyspace| = 26!

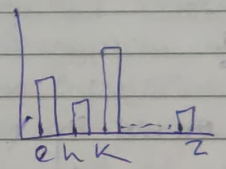
⇒ frequency of plain text & cipher text → preserved.
p_i → probability of i^{th} character in plain text.
q_i → probability of i^{th} character in ciphertext
in mono alphabetic cipher

$\therefore \forall i \exists j$ s.t. $p_i \cong q_j$
For MAS ↑

For shift cipher
$\forall i, \exists k$
$p_i \cong q_{i+k}$
(How to tell machine what is gibberish & what is not?)

$\sum_{i=0}^{25} p_i^2 \simeq 0.065$

$\sum_{i=0}^{25} p_i q_{i+k} \simeq 0.065$ For correct k



for cipher text.

→ Overlap 2 bar charts
→ Standard way:
(Not try out every permutation)
- what if ciphertext is too small?
→ Sort i & then overlap greedily.

⇒ Poly alphabetic Substitution: (Remain secured >100 years)
(Vegenere Cipher)

⇒ passkey: cat → if large enough ⇒ bruteforce won't work.
Cryptoc
+ cat cat
ER V
0 indexed
↳ How to break this?

⇒
(a) Assume that length of passkey is known & break.
(b) How to get the length of passkey?

(a) Vegenere cipher ⇒ collection of shift ciphers.
if len = 3:
1) $c_0\ c_3\ c_6$ ---- } shift by k_1 } Find k_i by prob. method 0.065
2) $c_1\ c_4\ c_7$ ---- } shift by k_2
3) $c_2\ c_5\ c_8$ ---- } shift by k_3

$c_0\ c_\ell\ c_{2\ell}\ c_{3\ell}$ ...
$c_1\ c_{\ell+1}\ c_{2\ell+1}\ c_{3\ell+1}$ ...

(b) → How to get length?
guess length ℓ
↳ check if correct or not.
$c_0\ c_\ell\ c_{2\ell}$ ....

$\sum_{i=0}^{25} q_i^2$ → ℓ is correct → ≈ 0.065
$\sum_{i=0}^{25} q_i^2$ → ℓ is incorrect → $\approx \frac{1}{26} \approx 0.038$

⟺ What is the definition of an "unbreakable" cipher?

↳ P.70.

10th Jan

Review:
Shift Cipher → monoalphabetic substitution cipher
↑ Principle of large keyspace (Bruteforce attack)
monoalphabetic substitution cipher ← frequency bound attacks
↓
Vegenere Cipher.
↓ ← Brute force + frequency sound attack.
↑ (perfect secrecy)
Shannon's Theory

⇒ Shannon's Perfect Secrecy:
→ Encryption scheme is a 4-tuple <Gen Enc Dec M>
→ An encryption scheme is said to be perfectly secret if for all probability distribution over M ∀ messages M and all possible ciphertext C.

leads to no new info & good/best for ciph
{ $P[\text{Message} = M \mid \text{Cipher} = C] = P[\text{Message } M]$ — [1]
(after looking at ciph) (before looking at ciph)

Theorem: One-time pad (Vernan Cipher) is perfectly secret.

One time pad:
$M = \{0,1\}^n$
3 algos & 1 message space.
[Gen]: Key generation algo.
(A) ← k — gen — k → (B) $k \in K$

[Enc] $(k, m) = c$

[$m \in M$] message space [$c \in C$] cipher space [$k \in K$] key space.

Dec : $(K, C) = M$

Gen : $K \xleftarrow{R} \{0, 25\}$

Enc : $c = m + k \pmod{26}$

Dec : $m = c - k \pmod{26}$

$M = [0, \ldots 25]^n$

$P[\text{message} = abc] = \alpha \rightarrow \left(\frac{1}{26}\right)^n$

$P[\text{message} = abc \mid \text{ciphertext} = bcd] = \beta$

↓ implies

message of form $(n, n+1, n+2), \ldots$ (26 poss).

$\therefore \boxed{\alpha \neq \beta}$

but if $n = 1 \rightarrow$ its a perfectly secret ; prob remains same .

One time pad : $|M| = \{0,1\}^n$

Gen : $K \xleftarrow{R} \{0,1\}^n$ Enc : $c = k \oplus m$

Dec : $m = k \oplus c$

$n = 3 \quad k = 101 \in_R$ 8 poss.

$m = 110$

$c = ?011 \quad R \xrightarrow{011} K$

(1) Alternate defⁿ :

Perfect secret encryption scheme

$\forall$ prob dist. over M, $\forall m \in M$, $\forall c \in C$

$P[\text{Ciphertext} = c \mid \text{message} = m] = P[\text{Ciphertext}]$ — 2

Baye's Theorem :

$$P[A|B] = \frac{P[B|A]\,P[A]}{P[B]}$$

$$P[m|c] = \frac{P[c|m] \cdot P[m]}{P[c]}$$

(equal implies)

(2) ⇒ An encryption scheme is p.s. iff $\forall$ prob dist over M, $\forall m_0, m_1 \in M$, $\forall c \in C$

$P[\text{Ciphertext} = c \mid \text{message} = m_0] = P[\text{Ciphertext} = c \mid \text{message} = m_1]$ — 3

2 ⇒ 3 (Trivial)

3 ⇒ 2 ?

Given $P[c|m_0] = P[c|m_1] \quad \forall m_0, m_1$

WKT $P[c] = \sum_{m \in M} P[c|m] \times P[m]$

$= P[c|m] \sum_{m \in M} P(m) \cdot 1$ (constant (from given))

$P[c] = z = P(c|m)$

Hence 3 ⇒ 2

One time pad (original def) : $P[c|m_0] = P[c = m_0 \oplus key]$

$= P[key = c \oplus m_0]$

$= \frac{1}{2^n}$ (key is chosen from uniform dist.)

$= P[key = c \oplus m_1]$

$= P[key = c \oplus m_2]$

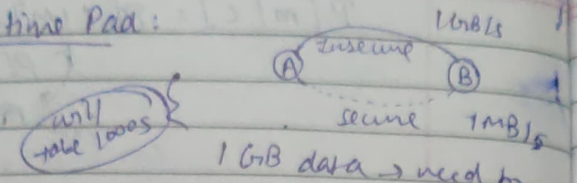
reveal proof for why not use same key in OTP

$\{ C_1 = m_1 \oplus K \text{ & same } K$
$C_2 = m_2 \oplus K \}$ → reveals something about msg
$C_1 \oplus C_2 = m_1 \oplus m_2$

given C_1, C_2 we can figure out only 2 m_1, m_2 s.t. $m_1 \oplus m_2 = C_1 \oplus C_2$

Limitations of One time Pad:

→



1 GB data → need to transfer securely from A → B

One bit key for one bit message

∴ Speed is limited by secure channel.

→ Could've just sent the entire message through secure channel.

Fast insecure channel
Slow secure channel

& same amt. of time (no need for Enc, dec)

⇒ One time pad → good to use if insecure & secure channels are not always available simultaneously.

↳ good to use if you want send message to yourself. (database to cloud & back)

⇒ OTP → can't get fast secure channel.

No secure channel < Slow secure channel < Fast secure channel

[No S.C. → Slow S.C. (Poor key Encryption) → never use this to send data]

[Slow S.C. → (symmetric key Encryption) Fast S.C.] → use combination

Thm: In any p.s. encryption scheme:

$|M| \leq |K| \Rightarrow H(M) \leq H(K)$

∴ ~~no. of bits of~~ Entropy of message is upper-bounded by entropy of key.

Proof: Suppose $|M| > |K|$,

Ciphertext → C. $\xrightarrow{k_1}$ $\xrightarrow{k_2}$ $\xrightarrow{k_3}$ $\left|\left(\begin{matrix} m_1 \\ m_2 \\ \vdots \end{matrix}\right)\right| \leq |M|$

implies

$\Rightarrow \exists\ m^* \in M$ s.t.

$P[m = m^* \mid C = c] = 0$

$\neq P[\text{message} = m^*]$

13th Jan

Review:

→ Breaking and patching - Historical Ciphers

→ Shannon's perfect secrecy - def^n

$\forall$ p.d.f M, $\forall m\ \forall c$

$P[\text{message} = m \mid \text{Ciphertext} = C] = P[\text{message} = m]$.

⇒ One time pad → perfect

⇒ Limitations of perfect schemes

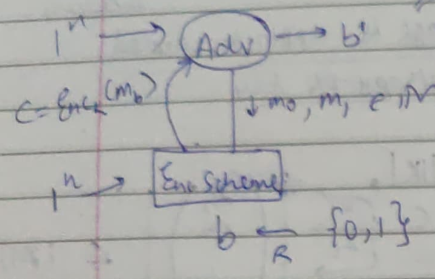
$|M| \leq |K|$

⇒ Impractical and/or impossible.

→ Two (Famous / Necessary) Relaxations to Shannon

∀ pdf over M
Adv → Adversary

⟹ Security against eavesdropping



b → bit of the message

∀ Adversaries A
$P(A(c) = b) = 1/2$

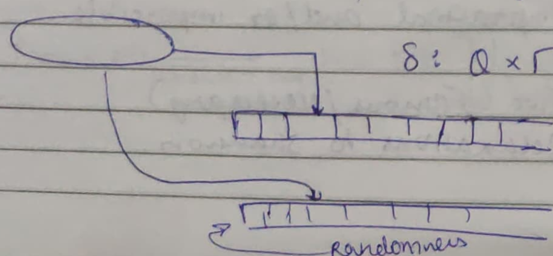
∀ Practical adversaries A
$P(A(c) = b) = 1/2 +$ tolerance

Security parameter ⟹ n (length of key)
Practical ≡ Efficient
≡ polynomial (n) resources time/space/energy etc.

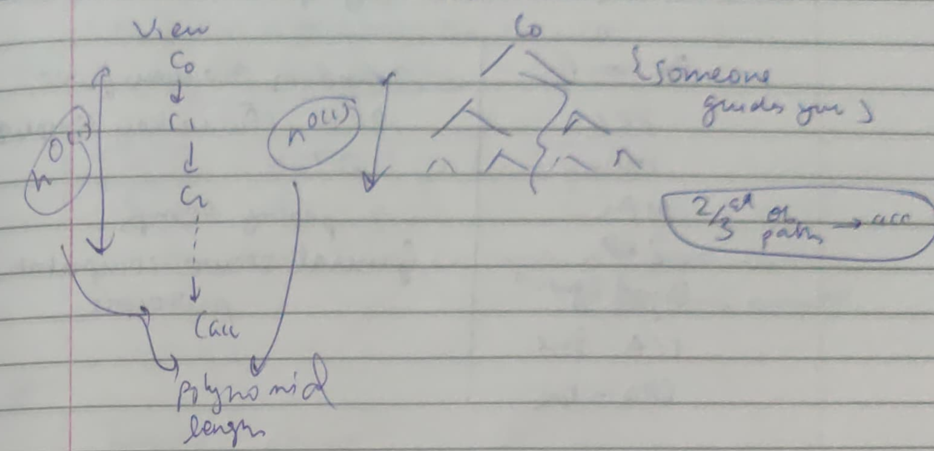
∀ PPTM A
(probabilistic polynomial turing machine)
$P(A(c) = b) = \frac{1}{2} +$ 'tolerance'

⟹ TM (turing machine)
$Q, \Gamma, \Sigma, \delta, q_{acc}, q_{rej}, q_{start}$

$\delta : Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$



Deterministic — Non deterministic



⟹ Negligible functions :

A function $\mu(n)$ is said to be negligible if ∀ polynomial p (asymptotically)
$\exists n_0 \; \forall n \geq n_0$

$\mu(n) < \frac{1}{p(n)}$

Sum summation of negligible func^n polynomial times is still negligible

1st relaxation

∀ PPTM A
$P(A(c) = b) \leq \frac{1}{2} + negl(n)$

→ 2nd relaxation

⟹ 2 (famous) necessary relaxations to Shannon
+ One way functions

+ (Trapdoor) One-way functions.

∃ One way fn.	∃ Trapdoor One way func
PRG	PKC (Public key cryptography)
Hashing	OT
MAC	2-party comp
ZKP	General secure computation protocols
Digital Signature	
CCA - Enc	
CPA - Enc	

⇒ Ciphertext only attack.

⇒ Known Plaintext attack (KPA)
↳ previous pair of (c & m) (prev message)

c^*

p_i, c_i

⇒ Chosen ~~Plaintext attack~~ Plaintext attack (CPA)

⇒ Chosen Ciphertext attack (CCA)

AES → secure/standard encryption

4 diff. types of security

① ⇒ Heuristic security
⇒ DES, AES, ~~RC4~~, RC6, ~~MD5~~

② ⇒ Proven security.
→ One time pad
→ Secret ~~sh~~ Share

③ ⇒ Provable security.
⇒ if one way func then PRNG

④ ⇒ none of the above.

[m_1] [m_2] [m_3] [m_4]
↓ ↓ ↓ ↓
c_1 c_2 c_3 c_4

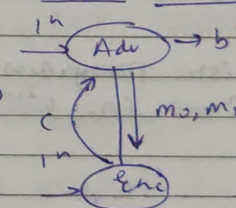
concatenate c_1, c_2, c_3, c_4 ⟶ will give ciphertext for entire message.

17th Jan

Review
→ Shannon's pessimistic Thm.
→ 2 relaxations to Shannon
→ PPTM Adversary.
→ Negligible error probability.

"Magic"
Two relaxations + ∃ "Special Mathematical Object"

Perfect Security.



C = Enc(m_b)

$b \in_R \{0,1\}$.

∀ Adv A
$P[b=b'] = \frac{1}{2}$

⇒ Computational Security

∀ PPTM A $P[b=b'] \leq \frac{1}{2} + negl(n)$

• Candidates for SMO that suffices:

① OWF (One Way function) ⤷ exists iff other exists

② PRG (Pseudo Random generator)

10^{-6} → not negligible (we can find a polynomial whose inverse is smaller than this)

$\mu(n) < \frac{1}{p(n)}$ — negligible func

$\frac{1}{n}$ → not negligible $\quad \frac{1}{n} \nless \frac{1}{n^2}$

$\frac{1}{n^k}$ → not negligible $\quad \frac{1}{n^k} \nless \frac{1}{n^{k+1}}$

e^{-n} → negligible → grows smaller at rate than any polynomial

2^{-n} → negligible

1.1^{-n} → negligible

$\frac{1}{(\log\log n)!}$ → not negligible $\quad \frac{1}{(\log n)!}$ → negligible

⇒ **PRG Defⁿ** : A deterministic program (or function)
$G : \{0,1\}^n \to \{0,1\}^{\ell(n)}$ is a PRG if:

Properties

① Expansion: $\ell(n) > n$

② Pseudo randomness : ∀ PPTM D

$$P[D(U_{\ell(n)}) = 1] - P[D(G(U_n)) = 1] \leq negl(n)$$

Discriminator etc.

$U_{\ell(n)}$ → uniformly chosen string of length ℓ

$G(U_n)$ → string generated by PRG given a string uniformly chosen of length n.

D should not be able to distinguish both of them → (expl. of math eq.)

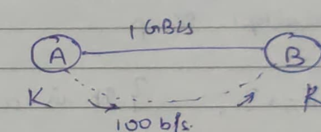
n = 100 bits

$\ell(n)$ = 1000 bits.

2^{1000} strings — TRG

2^{100} only possible outputs — PRG

$2^{100} < 2^{1000}$



One Time Pad

K → G(K) — No adversary can distinguish efficiently b/w G(K) & a randomly generated string of length same as G(K).

$Enc_K(m) = G(K) \oplus m$

$Dec_K(c) = G(K) \oplus c$

⇒ **One Way Functions (OWF):**

A function $f : \{0,1\}^n \to \{0,1\}^n$ is said to be a one way function if:

(a) Easy to compute : ∃ a polynomial-time DTM M, ∀ x $M(x) = f(x)$.

(b) Hard to Invert : ∀ PPTM M

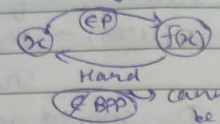
↳ P.T.O.

$P[M(f(x)) = y, f(x) = f(y)] \leq negl(n)$

polynomial time

$n = |x|$

Do OWF exist?



$NP \neq BPP \Leftrightarrow NP \neq P$

OWF exists or not } similar to P=NP

⇒ Discrete Logarithm Problem (DLP)

- Cyclic group G

$\mathbb{Z}_p^*$ p is prime

↳ Operation is multiplication % p

$\mathbb{Z}_3^* \rightarrow \{1,2\}$

$\mathbb{Z}_p^* \rightarrow \{1, \ldots, p-1\}$ } It is a Group.

Generator also exist

$\mathbb{Z}_5^* \rightarrow \{1,2,3,4\}$ (2, 4, 3, 1) 2 is generator for ⇒

p → prime
g → generator

$g^x \bmod p = y$ [given g, p, y, what is x?]

↓ One-to-One

input → $\log(p)$ bits (Brute force won't be polynomial)

$\mathbb{Z}_7^*$ $g = 3$

$3^x \bmod 7 = 4$

Find x. What if p → 1024 bit prime?

$x \rightarrow 4$

2, 4
6, 4.

→ Hard-core Predicates

$x \rightarrow f(x)$ (Easy)

$f(x) \rightarrow x$ (Hard)

If it is difficult to get x from f(x) → 1 bit which is hardest to determine → hard core predicate.

Given p, g, y, LSB(x) = ? → can be determined efficiently

Hence not a hardcore predicate.

→ For a family of OWFs $f : \{0,1\}^n \rightarrow \{0,1\}^n$, the function $h : \{0,1\}^n \rightarrow \{0,1\}$ is a hard core predicate (hcp) of f if

(a) $x \xrightarrow{\text{easy}} h(x)$

(b) $\forall$ PPTM M

$P[M(1^n, f(x)) = h(x)] \leq \frac{1}{2} + negl(n)$

Compute $y^{\frac{p-1}{2}} \bmod p$

WKT. $g^x \bmod p = y$

$\Rightarrow g^{x \cdot (\frac{p-1}{2})} \bmod p$

$\Rightarrow$ If $x = 2k$, $\rightarrow g^{k \cdot (p-1)} \bmod p$ = 1

$\Rightarrow$ If $x = 2k+1 \rightarrow g^{2k(\frac{p-1}{2}) + \frac{p-1}{2}} \bmod p$

$\rightarrow g^{\frac{p-1}{2}} \bmod p$

$y^{\frac{p-1}{2}} \bmod p \rightarrow +1$ or -1

-1 Congruent to prime 2 sqrts of each

$a^{p-1} \bmod p = 1$ Fermat's if a & p are co-prime

g → generator

↳ Number Theory

There is a polytime reduction from finding MSB(x) to finding all bits of x. ⇒ MSB(x) → hard core predicate

(*) With the help of hcp → we'll find equivalence of OWF and PRG.

⇒ Fermat's little theorem :

⊙ If p is prime

$\forall a \in [1, p-1] \Rightarrow a^{p-1} \equiv 1 \pmod p$

Proof

if $ai \bmod p = aj \bmod p$ } (not to prove)

then $i = j$

$\Rightarrow a(i-j) \equiv 0 \bmod p$.

a is not divisible by p

i-j has to be " " p

but $1 \le i$ & $j \le p-1$

$\therefore i = j$

$\Pi(a \quad 2a \quad 3a \quad \dots \quad (p-1)a)$ ⇒ both are same

$\Pi(1 \quad 2 \quad 3 \quad \dots \quad (p-1))$

$(p-1)!\, a^{p-1} = (p-1)!$

$\therefore a^{p-1} = 1$

24th Jan

DLP : Discrete Log Problem

Given $y = g^x \bmod p$ where g, p are known

Find $x \in [1, \dots, p-1]$

MSB(x)

⇒ Given $y = g^x \bmod p$ find if $x < \frac{p-1}{2}$?

PRG from OWF

Today

① Reduce DLP to MSB(x)

② Design a provably secure pseudorandom generator.

↳ If DLP problem has efficient algo, we can have PRG

→ Having TRG → does not solve problem but PRG does

⇒ LSB(x) ∈ P for DLP

↳ to know $x \to$ ∈ or ∉

→ If we can find MSB(x) ⟶ we can solve DLP.

LSB(x)

x is even — x is odd

$y' = \sqrt{y} \bmod p$.

$= g^{\frac{x}{2}} \bmod p$

$y' = \frac{y}{g} \bmod p$.

$= g^{x-1} \bmod p$

$\pm g^{\frac{x}{2}} \bmod p$

↳ 2 poss → ∴ can't solve DLP

How to find sqrt in Z_p^*?

Is DLP but google it algo

Sqrt Algo

$x^2 \equiv z \pmod p$ → solve for x

Assume $p \equiv 3 \bmod 4$

$z^{\frac{p+1}{4}} \bmod p = x$ ⇒ given z

$z^{\frac{p+1}{2}} \bmod p = x^2$

$= z$

(we know $z^{\frac{p-1}{2}} \to +1$ / -1)

Ex. p = 19

z = 9

$9^5 \bmod 19 = \pm 16$

$x^2 \equiv 1 \pmod p$

$p \mid x^2 - 1$

$p \mid (x-1)(x+1)$

1 & -1 ↳ only 2 possible for

If we can answer this given we know we want $g^{x/2} \% p$ else

$g^{\frac{x}{2}} \bmod p$

$g^{\frac{x}{2} + \frac{p-1}{2}} \bmod p$.

If $\frac{x}{2} < \frac{p-1}{2}$ $\quad \frac{x}{2} + \frac{p-1}{2} > \frac{p-1}{2}$

New Ques : Given $y = g^x \% p$ where g, p are known, find $x \in [1 \ldots p-1]$
(Hardcore predicate for DLP)

⇒ Given a ~~OWF~~ one way permutation f, and its hcp h, ⇒ find PRG

$G : \{0,1\}^n \to \{0,1\}^{n+1}$

$$G(s) = f(s) \,\|\, h(s)$$

(‖ : concatenation)

Suppose ∃ PPTM D,

$$P[D(U_{n+1}) = 1] - P[D(G(U_n)) = 1] \geq \epsilon(n)$$

(not negligible)

- U_{n+1} : last bit is uniformly distributed.
- $G(U_n)$: first n bits are identically distributed
- last bit : $h(s)$

~~$D(h(s), f(s))$~~ :

$$P[D(f(s)) = h(s)] \geq \frac{1}{2} + \text{negl}(n)$$

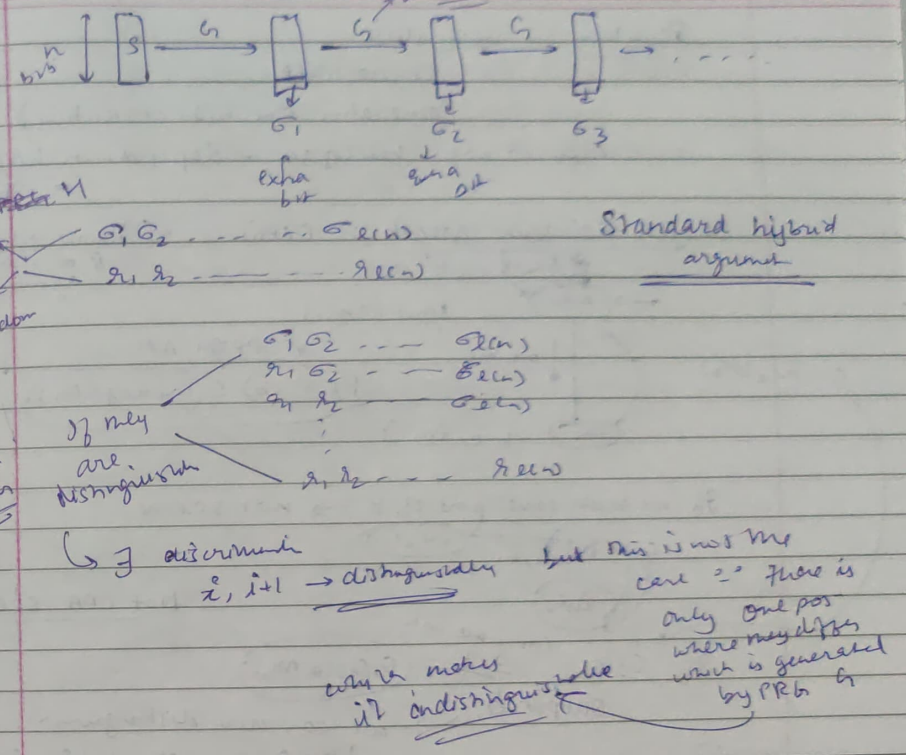
If h → hardcore predicate, acc to def^n,

$\forall$ PPTM M : $P(M(f(x)) = h(x)) \leq \frac{1}{2} + \text{negl}(n)$

$\rightarrow$ ⇒ contradiction

$\leq \epsilon(n)$

⇒ Given a PRG $G : \{0,1\}^n \to \{0,1\}^{n+1}$, how to construct a PRG $H : \{0,1\}^n \to \{0,1\}^{l(n)}$ for any polynomial $l(\cdot)$.

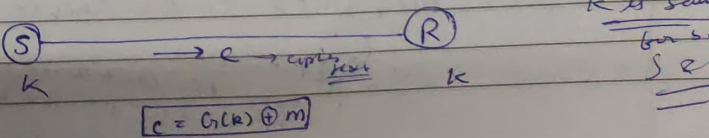


① Assignment (PRG)

WAP for probably secure PRG, assuming DLP is a one-way permutation.

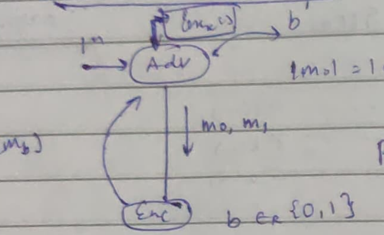
② Assignment (Secure Encryption against eavesdropper)

$Enc(m) = G(k) \oplus m$.



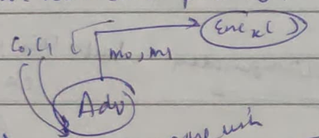
Problem : very slow
∵ sequential
(generating one bit at a time)
Algo → not stateless → depends on history

Chosen Plaintext Attack : (CPA) [Above scheme is not secure against this]



∀ PPTM Adv
$P(b' = b) \le \frac{1}{2} + negl(n)$ — not happening

If we secure some part of K → not secure.



Not CPA secure

If $c = c_b$, compare with $G(k) \oplus m_b$ ($b = 0, 1$)
output b. → we can now distinguish b/w m_0, m_1 msgs. ⇒ with prob (1).

Th^m } No deterministic encryption scheme is CPA-secure

⇒ Probabilistic Encryption :
define encryption scheme with key (k) msg (m)
$Enc_k(m)$ → random noise
$= \langle r, Enc_k(r) \oplus m \rangle$
$Dec_k(r, c) = c \oplus Enc_k(c)$
↳ CPA secure
(∵ now Adv can't distinguish b/w c_0 & c_1 i.e. m_0 & m_1 ∵ of random noise)

27^th Jan

Review :
→ Hard core predicates of OWF (Eg. DLP)
→ PRG from one way permutations
→ Secure-encryption against Eavesdropper
$c = G(k) \oplus m$
→ Need for probabilistic Encryption
→ CPA - Security Definition

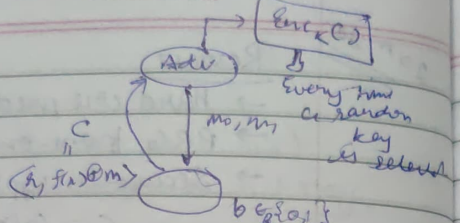
Today :
→ Pseudo-Random Functions
→ CPA - Secure encryption
→ mode of operation

(*) Pseudo Random Functions (PRF)
⇒ Computationally indistinguishable from a Truly random functions

Claim
f → TRF $f : \{0,1\}^n \to \{0,1\}^n$
↳ i.e. a string is randomly chosen.
How many functions possible? ⇒ ~~2^{2n}~~ 2^{n2^n}
2^n strings.
(Want to pick a truly random fun. ↳ $n2^n$ addressing bits required)

∴ Cannot pick a TRF.
If we have TRF, CPA - security → straightforward.

⇒ $c = \langle r, f(r) \oplus m \rangle$ (f ↳ TRF) Key size : $n \cdot 2^n$ bits



$r \rightarrow$ fully random string
$f \rightarrow$ truly random fun
$\therefore f(x) \rightarrow$ truly random

$\forall$ PTM A

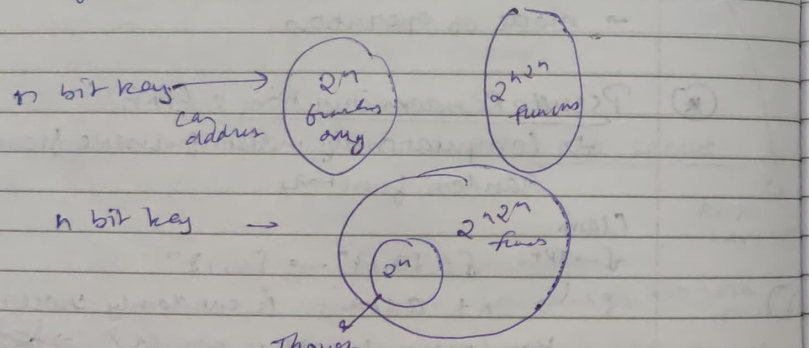
$P[b' = b] = 1/2$.

2 relaxation

$\forall$ PPTM A

$P[b' = b] \leq \frac{1}{2} + negl(n)$

If PRF is used $\rightarrow$ a key of n bit only required for it to be CPA secure.



$\Rightarrow$ $F: \{0,1\}^n \times \{0,1\}^n \rightarrow \{0,1\}^n$ is a PRF if it is easy to compute $F(k,n)$ and

$\forall$ PPTM D

$|P[D^f(1^n) = 1] - P[D^{f_n}(1^n) = 1]| \leq negl(n)$

$\because$ Turing machine only accepts strings & not functions.

PRF $\rightarrow$ more complex than PRG.

$|P[D^f(1^n) = 1] - P[D^F(1^n) = 1]| \leq negl(n)$

$f \rightarrow$ fully random; $F \rightarrow$ Pseudo Random.

PRG $\xrightarrow{to}$ PRF

PRG $G: \{0,1\}^n \rightarrow \{0,1\}^{2n}$

$G_0(x) = Left(G(x))$ $\rightarrow$ left half
$G_1(x) = Right(G(x))$ $\rightarrow$ right half
} length preserving function

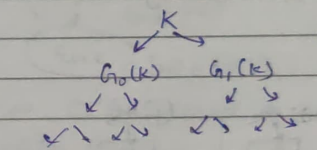
$\therefore$ $G(x) = G_0(x) \,|\, G_1(x)$

$x = x_0 x_1 x_2 \ldots x_{n-1}$ $\rightarrow$ randomly chosen string of 0 & 1.

$F_K(x) = G_{x_{n-1}}(\ldots(G_{x_1}(G_{x_0}(k)))\ldots)$

$\downarrow$ n bit string output

PRF



2^n possible outputs

If G $\rightarrow$ is PRG then $\Rightarrow$ F $\rightarrow$ PRF.

$\Rightarrow$ CPA Secure Encryption :

$Enc_K(m)$ $\quad \langle r, F_K(r) \oplus m \rangle$

$Dec_K(r, c) = F_K(r) \oplus c$.

$\hookrightarrow$ Pr. 2

CPA secure → till now → large doubling n → 2n

Assignment 1 (CO3)

WAP to design a CPA-secure encryption scheme that is provably secure.

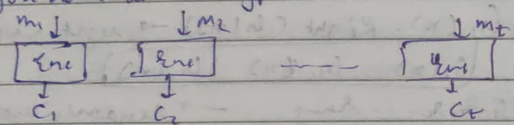
⇒ **Modes of Operation:**

$m = m_1 m_2 \ldots m_t$ ⇒ nt bits.
(t blocks)

① Electronic Code **ECB**

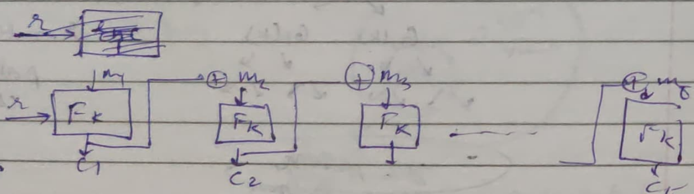
If you have an encryption scheme:

Use same r for encryption



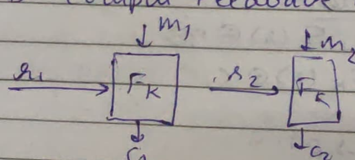
Enc → $\langle r, F_k(r) \oplus m \rangle$;

② **CBC** → Cipher Block chaining.



Only need n bit string (r).

③ OFB (Output Feedback mode)



$r_i = F_k^{(i)}(r)$

(r_i → dependent on r)

∴ Instead sending all r → to every block we can calc i^{th} r thing
(Problem → sequential)

④ Randomized Counter mode. (RCM)

$r_i = F_k(r + i)$

Enc: $m_1 \ldots m_t$

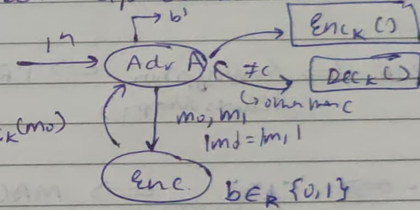
$\langle r, c_1, \ldots, c_t \rangle$

$c_i = F_k(r_i) \oplus m_i$

AES → Conjectured PRF

$c = F_k(m)$ → Deterministic & hence not CPA secure

★ Chosen-Ciphertext Attack (CCA)



$c = Enc_k(m_b)$ $\quad b \in_R \{0,1\}$

$c = \langle r, F_k(r) \oplus m \rangle$

Enc() is CCA secure if

∀ PPTM A

$P(b = b') \leq \frac{1}{2} + negl(n)$

$m_0 = 0^n$

$m_1 = 1^n$

$c_b = \langle r_b, F_k(r_b) \oplus m_b \rangle$

$c_b' = c_b$ with LSB toggled.

If I decrypt c_b':

$Dec(c_b') =$ if b=0: $0^{n-1}1$; if b=1: $1^{n-1}0$

This is CPA-secure but not CCA-secure.

Assignment-1 Q4:

⇒ CPA secure encryption is VARIOUS MODES OF OPERATION

(1 way fn ⇌ PRG ⇌ PRF ⇌ PRP) — slower it gets — working

RCG, AES is PRF assumption, DES

31st Jan

Review

→ OWF ⟺ PRG ⟺ Encryption (eavesdropping) ⟹ PRF ⟺ CPA secure encryption ⟺ PRP ⟺ MAC ⟸ CCA encryption ⟸ …

⇒ PRF F_k

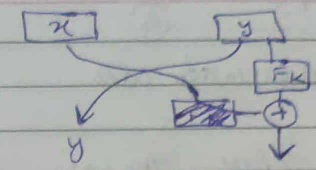
we don't require it to be one-to-one (∵ Probabilistic)

$\langle r, F_k(r) \oplus m \rangle$

↳ CBC
OFB
Rand Ctr.

⇒ Fiestel Network

Func takes input in 2 parts.



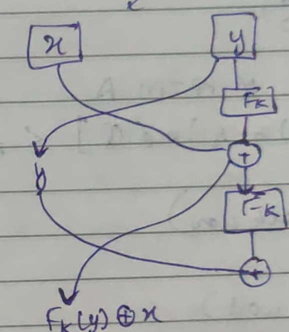
$(x, y) \rightarrow (y, F_k(y) \oplus x)$

Invertible func if not PRF.

$(x, y) \rightarrow (y, F_k(y) \oplus x)$

$(F_k(y) \oplus x,\ y \oplus F_k(F_k(y) \oplus x))$ } keep on repeating this

↓



$(y \oplus F_k(F_k(y) \oplus x),$
$F_k(y) \oplus x \oplus F_k(y \oplus F_k(F_k(y) \oplus x)))$

If you start with PRF, randomness percolates as you increase iterations.

↳ 3 iters → necessary
↳ 4 iters → sufficient.

$\langle r, F_k(r) \oplus m \rangle$

↳ CPA secure but NOT CCA secure

★ Data Integrity:

(A) —→ m —(Adv)— (B)

m

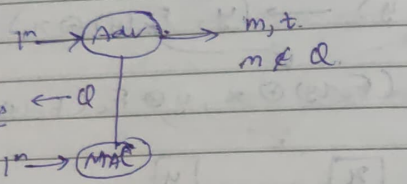
(Adv can mix something with m^0 → make it m', how does B know if m received is same as what A sent m^0 or some other m'?)

(A) → ⟨m, t⟩ (B)

MAC : message authentication code.

$MAC_K(m) = t$

$\forall m, f_K(m,t) =$ ~~Reject~~ Yes / NO



MAC is secure if $\forall$ PPTM A

$P[\forall r\, t_r(m,t) = r \mid m \notin Q] \leq negl(n)$

(REPLAY is not an attack now)

Example (Doesn't work)

(1) $|m_1| = n$

$m = m_1 \ldots m_t \qquad t = \bigoplus_{i=1}^{t} F_K(m_i)$

works for $t=1$ → PRF

$F_K(m_i)$ → indistinguishable TRF

∴ For each new msg → new random no.

predicting Out come is negligible

⇒ Free to get tags for any msg you want

tag of 1^n & tags of end ⇒ ⊕ them & get a tag for new msg that was never queried.

Sequence Numbers :

$1 \| m_1 \quad 2 \| m_2 \quad (3 \| m_3) \quad \ldots \quad t \| m_t$

we get attack for this

Permutation attack

attack fail here

attack fail here

(2) $t = \bigoplus_{i=1}^{t} F_K(i \| m_i)$ → what will be the attack for this?

Does not work ~~wrong~~ → prefix

(3) $\langle t_1, \ldots t_K \rangle$

$t_i = F_K(i \| m_i)$ ⇒ Still doesn't work

(simple prefix & xor attack)

$m = m_1 \ldots m_K \qquad m' = m_1' \ldots m_K'$

$t_1 \ldots t_K \qquad t_1' \ldots t_K'$

take tags from both of them & combine them.

↳ get a tag for new msg.

⇒ use message ids

$t_i = F_K(r \| i \| m_i)$ This works

$t_i = F_K(r \| l \| i \| m_i)$ secure & robust to all vb them

(Permutation, Prefix & Inter linear attacks

↳ Apart from this no attack is possible

(message ids + length + sequence numbers

↳ meets the security def -

but now tag is as big as msg)

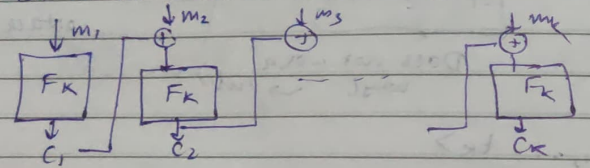
Practitioners do following:

① CBCMAC (used to do min) (Current trend)

base → HMAC (Industry standard) → Hashing

tag for $m = m_1 \dots m_k$?

Using F_K, encrypt m in CBC mode and t = last output



⇒ Basic CBCMAC is secure for fixed length inputs.

If we allow queries of variable length → we can break basic CBCMAC)

Breaking CBCMAC: (adversary)

m_1, t_1 (m_1 → get t_1)

$m_2 = t_1 \oplus m$ $m_1 \| t_1 \oplus m$, t_2

t_2 is valid tag of m which I never queried

Secure CBCMAC:

ⓐ Prepend length to msg.

append length → won't help ∵ our attack is prefix attack)

ⓑ ~~Apply $F_K(1)$ to obtain a new key K' & continue~~

Use 2 keys k_1, k_2.

k_1 obtains t_1

$t_2 = F_{k_2}(t_1)$

ⓒ Apply $F_K(2)$

Assignment 1 Q5

WAP for secure CBCMAC (using the PRF, already available).

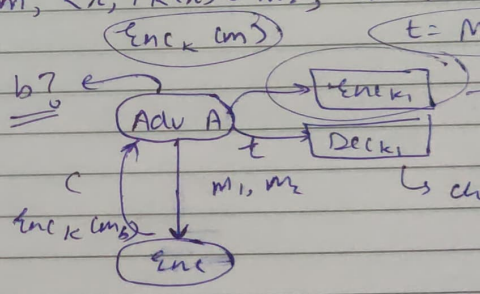
Assignment 1 Q6

WAP for CCA-secure encryption

Given CPA secure encryption & secure MAC } ⇒ CCA-secure encryption

Encrypt then authenticate.

m, $\langle r, F_K(r) \oplus m \rangle$, $\langle\langle r, F_K(r) \oplus m \rangle, t \rangle\rangle$



∴ you can authenticate after encryption → CCA secure

DLP is hard → SOTA CCA secure encryption scheme

Assume AES → PRF → F_K

RC4 → PRG

$c = AES_K(m)$ ✗ → won't even be CPA secure

entire thing now becomes CCA secure

AES_K in CBC, ... mode → then it becomes CPA secure

↳ tag it with $CBCMAC_K$ (ciphertext)

7th Feb

Review

→ Shannon's Pessimistic Yuck

2 Relaxations & 1 Assumption.

Symmetric Key Crypto

Public key Crypto

Ciphertext only attack (PRG based)

CPA security (Probabilistic func. PRF based)

MAC [CBCMAC, HMAC, ...]

CCA security [CPA Sec + Secure MAC]

(Encrypt then Authenticate).

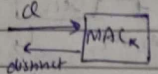
⊙ MAC scheme :

$m = m_1 m_2 \ldots m_n$

$t_i = F_k(r \| i \| l \| m_i)$

$t = t_1 t_2 \ldots t_n$

Proof of security



→ prob. that r will be repeated → negligible

Case #1 r is re-used. (unique m'e Q)

Case #1.1 $|m| = |m'|$ (length of new msg is equal)

To $m \neq m'$ & $|m| = |m'|$

$\Rightarrow \exists\ i\ m_i \neq m_i'$

$\therefore t_i = F_k(r \| i \| m_i')$

$\therefore r \| i \| m_i'$ } new msg entry

$\in F_k(\ldots)$ → completely random

∴ We now have a new msg entry.

Case #1.2 $|m| \neq |m'|$

$F_k(r \| i \| \ldots)$

$F_k(r \| i \| \ldots) = ?$ ✗ length → problem

~~You've done~~ ~~get a new msg~~.

Case #2 r is new

$F_k(r \| \ldots) = ?$ ✗ r → problem

Assignment II Q1

WAP a provably secure MAC scheme (using your PRF)

Collision-Resistant Hashing

Hash function?

$\{H^1, H^2, \ldots\}$ → family of many func of

~~& key~~

F_k → k → secret key $n = |k|$

F^s → known key (indexing)

$\therefore H^s: \{0,1\}^* \to \{0,1\}^{\ell(n)}$

∀ PPTM A, A and H.

~~$Pr[A^{H^s}]$~~

$$Pr[A(H^s, 1^n) = (x, y) \text{ s.t. } x \neq y \wedge H^s(x) = H^s(y)] \leq negl(n)$$

CRHF ⇔ PRF ⇔ PRG ⇔ hcp (OWP)

→ collision free func ⇒ $\{0,1\}^{l(n)} \to \{0,1\}^{l(n)}$

~~→ There are easy to build compression func~~
~~→ There are easy to build infinite compression func~~
~~but tough to build~~

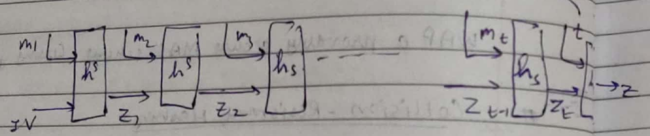
★ Merkle Damgard Transform : (If we are able to compress by very few bits ⇒ we can have ∞ compression as well)

50% compression — Input : $h^s : \{0,1\}^{2n} \to \{0,1\}^n$
↳ collision resistant hash func (assumption)

Output : $H^s : \{0,1\}^* \to \{0,1\}^n$

works for n+1 → {0,1} as well — Proof

$m = m_1 \dots m_t$ (length of msg)



Case 1 : If $|x| \neq |y|$, ~~$H^s(x) = H^s(y)$~~
↳ we have a collision for h^s (Z is same for both)
(last instant $t_x \neq t_y$) ↳ not possible

Case 2 : If $|x| = |y|$, but $x \neq y$
⇒ $\exists\, i,\ x_i \neq y_i$
⇒ collision for h_s for some i from last instance of h_s.
∴ again contradiction.

⇒ n bit output ⇒ 2^n outputs. (bruteforce)

⇒ Birthday attack (generic bruteforce attack for hash func)

Birthday probability → same room more than 0.5 prob → 2 people have same birthday (Ans → 23).

1, 2, … N indexed objects
Pick q times independently.
What's the chance of picking the same no. more than once? ↳ collision — $q = O(\sqrt{N})$

Thm : Pr [collision] ?

$$\frac{q(q-1)}{4N} \leq \Pr[\text{collision}] \leq \frac{q(q-1)}{2N}$$

Proof :

$\Pr[\text{collision}] = 1 - \Pr[\text{No collision}]$

$\Pr[\text{No collision}_q] = \Pr[NC_1] \cdot \Pr[NC_2 \mid NC_1] \cdot \Pr[NC_3 \mid NC_2] \dots \Pr[NC_q \mid NC_{q-1}]$ (in q^{th} trial)

$\Pr[NC_1] = 1$ (for first trial)

$\Pr[NC_2 \mid NC_1] = 1 - \frac{1}{N}$ $\quad \Pr[NC_q] = \prod_{i=1}^{q-1} \left(1 - \frac{i}{N}\right)$

$\Pr[NC_3 \mid NC_2] = 1 - \frac{2}{N}$

⋮

$\Pr[NC_{i+1} \mid NC_i] = 1 - \frac{i}{N}$.

∴ $\Pr[NC_q \mid NC_{q-1}] = 1 - \frac{q-1}{N}$

$0 \leq x \leq 1$

$\Rightarrow 1 - x \leq e^{-x} \leq 1 - \frac{x}{2}$ $\quad$ ∴ (Assume $\frac{i}{N} \to x$)

$$P_r[NC_q] \le \prod_{i=1}^{q-1} e^{-\frac{i}{N}}$$

$$P_r[NC_q] \le e^{-\frac{1}{N}\sum_{i=1}^{q-1} i} = e^{-\frac{1}{N}\frac{q(q-1)}{2}}$$

$$\le 1 - \frac{q(q-1)}{2N}$$

$h^s : \{0,1\}^{2n} \to \{0,1\}^n$ → is not collision resistant, DLP should not be hard anymore.

$h^s(x_1, x_2) = g^{x_1} b^{x_2}$ → random

g is a generator of G_q

$b \in G_q$

x_1, x_2 : 2 n bit numbers

g : 1 element in group G_q

b : n bit

$2n \to n$

2 integers → 1 integer (50% compression)

$g^z = b$; $z = ?$

If we find a collision for h^s → we can solve DLP

$h^s(x_1, x_2) = h^s(y_1, y_2)$ where $x_1, x_2 \neq y_1, y_2$

$g^{x_1} b^{x_2} = g^{y_1} b^{y_2}$

$g^{x_1} b^{x_2} = b^{y_2} b^{x_2}$

$\Rightarrow g^{x_1 - y_1} = g^{z(y_2 - x_2)}$

$\Rightarrow z = \frac{x_1 - y_1}{y_2 - x_2} \mod$ (group's order)

(ie mod $(p-1)$ bc Z_p^*)

$\therefore z = 1 \ldots p-1$

but DLP is hard to solve $\Rightarrow z$ → hard to find.

$\Rightarrow h^s(x_1, x_2) = g^{x_1} b^{x_2}$ → provably secure collision resistant hash function

Assignment II

WAP for secure collision resistant hashing (using DLP)

Assignment II Q3

WAP for provably secure HMAC

10th Feb

Story so far:

- → OWFs
- → PRGs
- → Stream ciphers ($G(k) \oplus m$ style of encryption)
- → PRFs
- → CPA secure encryption
- → Block ciphers (& their modes of op).
- → Pseudorandom permutations
- → MACs
- → CBCMAC

→ Hashing (with collision resistance)

→ HMAC.

→ CCA secure encryption (CPA + MAC)

⇒ HMAC (MAC based on Hash)

Message m

Key k

owasp

ipad (inner pad) → constant

$H^s(k \oplus opad \| H^{iv}((k \oplus ipad) \| m)) = Tag.$ } HMAC with key k.

opad = 3636... (Hexadecimal)

ipad = 5c5c... (Hexadecimal)

All of the above points are equivalent
(all or nothing situation)
∴ ↳ They are called **Minicrypt**

Trapdoor OWF ⇒ OWF
OWF ⇏ Trapdoor OWF
(we don't know yet (most likely possible but not proved))

They are called **Cryptomania**

(margin: list of equivalent scenarios)

(a) Algorithmica ($P = NP$)
(b) Heuristica ($P \neq NP$ (worst case), $P_{avg} = NP_{avg}$ → avg domain) dist $P \neq$ dist NP
(c) Pessiland (dist $P \neq$ dist NP (avg difficult), OWFs don't exist)

(margin: Anti crypto)

(d) Minicrypt (OWFs exist, but Public key crypto doesn't)

(margin: done already)

(e) Cryptomania (PKC exists)

(margin: will do this later; conjecture that we are here)

but we don't know ⇒ All exclusive, if in 1 world then not others

→ Bit commitment
→ Zero knowledge proofs
→ Digital signatures
} Top 3 primitives

↳ are in minicrypt (but all practical signatures ↳ used for PKC)

★ Bit commitment :

(A) → c → (B)
Bit b → b
A wants to commit bit b to B.
(c doesn't reveal b)
(b is not modified)

Commit phase
Reveal phase

During commit → no one knows what I committed
During reveal → everyone knows what I did & it was not modified as well.

Binding
Blinding

Theoretically impossible to know whether msg was modified after commit or not.

ZKP → exists iff B.C. exists ⇒ iff OWF exist.

Commit phase : B doesn't know bit b.
Reveal phase : B gets b; an assurance that b is unchanged.

⇒ Perfect hiding : c should not have any info about b.
⇒ Impossible.
∴ then output doesn't depend on input

Full Information about b vs No knowledge of b (→ info that can be extracted efficiently.)

One way permutation (F)
$g^x \bmod p$ → full info about x ↳ but no knowledge about x

$c = Enc_k(b)$ → hiding is perfectly done.
↳ but reveal is sus.

∴ $c = Enc_{k'}(b')$ → with some other key k' & some other bit b'

i.e. if I give k' to decrypt c during reveal → will get a diff msg
∴ No idea / assurance that msg is changed or not.

PAGE NO.
DATE

Bitcommitment protocol:

Let f be a one way permutation. (one to one func)

$h \rightarrow f$'s hard core predicate. (unique input ⇒ unique output)

$f: \{0,1\}^n \rightarrow \{0,1\}^n$

$h: \{0,1\}^n \rightarrow \{0,1\}$.

One way property:
$\forall$ PPTM A
$P[A(f(x), 1^n) = y \text{ s.t. } f(x) = f(y)] \leq negl(n)$

Hard core predicate for OWF property:
$\forall$ PPTM A
$P[A(f(x), 1^n) = h(x)] \leq \frac{1}{2} + negl(n)$.

⇒ To commit to bit b:
① Choose an n-bit string s uniformly at random
② Publish $\langle f(s), h(s) \oplus b \rangle$
[Given $f(s)$, → won't be able to predict hcp of $f(s)$ & then one time pad of b]

⇒ To reveal b:
① Publish $\langle s, b \rangle$.

⇒ To check the validity of b:
① Compute $f(s)$ & compare ⇒ s is not modified
② Compute $h(s)$ ⇒ $h(s)$ is not modified
③ Compute $b = h(s) \oplus [h(s) \oplus b]$ (you got during commit phase)
④ Compute $b \stackrel{?}{=} b'$ ⇒ b is not modified.

PAGE NO.
DATE

Example (using DLP)
→ To commit to bit b
Send $g^s \bmod p$; $MSB(s) \oplus b$
→ Reveal phase
Send s, b.

⇒ What is efficiency? ⇒ What is intelligence?
⇒ What is negligible? → What is a fore/back ground process?
⇒ What is fault?
⇒ What is fairness?
⇒ What is trust?

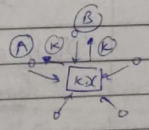
Hw for

Public Key Cryptography

Main limitations of Symmetric Key Crypto:
→ Secure backchannel for key establishment.
→ Too many keys. (key management issues)
$n \rightarrow {}^nC_2$ pairs $\rightarrow {}^nC_2$ keys

(Trusted)
⇒ Key distribution centres (KDC)

n nodes ⇒ 1 single KDC
${}^nC_2$ keys
$O(\frac{n^2}{2})$ keys



O(n) keys now
Securely communicate with KDC only

✗ Open Systems ? (you don't know users)

✓ ⇒ KDC → works in closed system
↳ it knows everyone in closed system

✗ Single point of failure

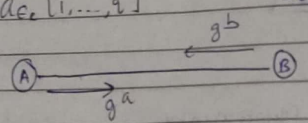
⇒ Diffie Hellman Protocol

DLP : $y = g^x$ mod p → working in $\mathbb{Z}_p^*$

$\therefore q = p-1$ ← order

Choose integer b/w $1 \dots q$.

$a \in_R [1, \dots, q]$ $b \in_R [1 \dots q]$



A: $a; g^b \Rightarrow (g^b)^a = g^{ab}$ (key)

B: $b; g^a \Rightarrow (g^a)^b = g^{ab}$ (key)

Eve obtains g^a, g^b

Key = g^{ab} ? Looks secure but no proof yet

↳ Secure based on assumption that "it is secure"

DDH assumption (Decisional Diffie Hellman assumption)

Given $g^a, g^b, \langle G, q, g \rangle$

∀ PPTM A

$P[A(g^{ab}, g^a, g^b) = 1] - P[A(R, g^a, g^b) = 1] \leq negl$

Not known

DLP ⇒ OWF ⇔ PRG ⇒ PRF ⇔ SKE (CPA) ⇒ Block ciphers ⇒ MAC

(1) Not known

DDH ⇒ (SKE) Bit Commitment ⇔ Hash map

↳ Secure key exchange ⇔ Trapdoor OWF

⇔ OT ⇔ SMPC ⇔ PKC ⇔ ...

" Universal Assumption"

∴ DDH protocol is secure → I can get everything impossible like OWF etc

(not sure if this statement is correct or not)

Assume P ≠ NP proof is not discovered,

SKE $\not\Rightarrow_{BB}$ OWF

(No black box reduction).

DDH ⇒ DLP.

DLP ⇏ DDH.

↳ necessary for DDH but not sufficient

∴ DLP → not hard, DDH is broken

∴ DDH is secure ⇒ DLP is hard

"RSA assumption"

↳ also a universal assumption.

(pk, sk)

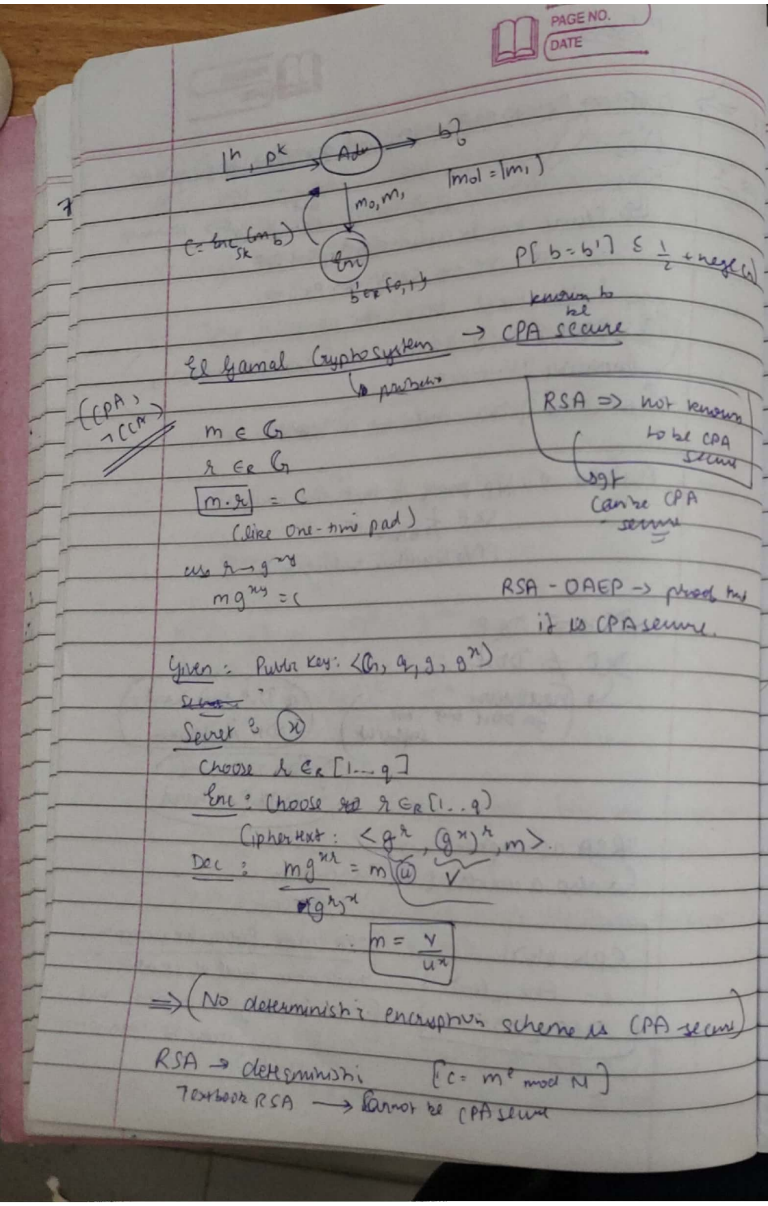
$c = Enc_{pk}(m)$

$Dec_{pk}(c) = m$

CPA secure Public key crypto system

↳ minimum level of security

(∵ mandatory assumption that adv has access to encryption server)



$P[b = b'] \leq \frac{1}{2} + negl(\lambda)$

ElGamal Cryptosystem → CPA secure

(CPA, ¬CCA)

$m \in G_q$

$r \in_R G_q$

$m \cdot r = c$

(like One-time pad)

use $r \to g^{xy}$

$m g^{xy} = c$

RSA ⇒ not known to be CPA secure

RSA - OAEP → proved that it is CPA secure.

Given: Public key: $\langle G, q, g, g^x \rangle$

Secret: x

Choose $r \in_R [1 \ldots q]$

Enc: Choose $r \in_R [1 \ldots q)$

Ciphertext: $\langle g^r, (g^x)^r, m \rangle$.

Dec: $m g^{xr} = m \cdot V$

$g^{ry x}$

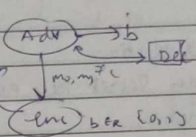
$m = \frac{V}{u^x}$

⇒ (No deterministic encryption scheme is CPA secure)

RSA → deterministic $[c = m^e \bmod N]$

Textbook RSA → cannot be CPA secure

CCA secure → Not possible for ElGamal



$c_0 = \langle g^{r_0}, g^{x r_0} m_0 \rangle$

$c_1 = \langle g^{r_1}, g^{x r_1} m_1 \rangle$

$c_b = \langle g^{r_b}, g^{x r_b} m_b \rangle$

$c_b' = \langle g^{r'} \cdot g^{r_0}, g^{r_1 x} \cdot g^{r_0 x} m' m_b \rangle$

$g^{r'}, g^{r'x} m'$ for some msg $g m'$

$c_b' = \langle g^{r'+r_0}, g^{r_1 x + r_0 x} m' m_b \rangle$

From Dec server, we get $m' m_b$

m' → gives my message

∴ we can get m_b from output of dec server.

Hence, not CCA secure ElGamal

How to make ElGamal CCA secure?

Anything Homomorphic or homomorphism in Enc scheme → cannot be CCA secure → but can be CPA secure

17th Feb

Review :
→ Public-Key revolution
→ Diffie Hellman Secret key establishment
→ DDH Assumption
→ CPA secure PKC
→ El Gamal CPA secure PKC.
→ CCA-security
→ El Gamal is not CPA secure.

Today : RSA Assumption
RSA PKC.

⇒ RSA Cryptosystem :

Key Generation :

Public key $N = pq$; e $(\gcd(e, \phi(N)))$
if $N = pq$
$\phi(N) = (p-1)(q-1)$

$\phi(N)$ = No. of posibie integers $< N$ and coprime to N
$\phi(N) = |Z_N^*|$

$\langle N, e \rangle$

Private/Secret key : d s.t. $ed = 1 \pmod{\phi(N)}$

$c = Enc_{PK}(m) = m^e \bmod N$
$Dec_{SK}(c) = c^d \bmod N$

Correctness :

$c = m^e \bmod N$
$c^d \equiv m^{ed} \bmod N \stackrel{?}{=} m$

Euler's extension of FLT

Corollary : $m^{\phi(N)} \bmod N = 1$

Thm In any group, G, $\forall g \in G$, $g^{|G|} = 1$

Proof

$Z_p^* \Rightarrow \phi(p) = p-1$
(∵ p is prime)
$\Rightarrow a^{p-1} \equiv 1 \bmod p$ (FLT)

This would imply $m^{ed} \bmod N = m^{ed \bmod \phi(N)} \bmod N = m$
($\because ed \bmod \phi(N) = 1 \bmod \phi(N)$)

$g_1, g_2, \ldots, g_{|G|}$ are elements of G
If $g \cdot g_i = g \cdot g_j \Rightarrow g_i = g_j$
↳ How to prove this?

There is a unique inverse of every element in G
This would imply
$\therefore g^{-1} g \cdot g_i = g^{-1} g g_j \Rightarrow g_i = g_j$
Hence proved.
(∵ gg_i is a unique element)

$g_1 g_2 \cdots g_{|G|} = (g g_1) \cdots (g g_{|G|})$
$g_1 \cdots g_{|G|} = g^{|G|} g_1 \cdots g_{|G|}$
$\Rightarrow g^{|G|} = 1$

$N = \prod_{i=1}^{k} p_i^{\alpha_i}$

$\phi(N) = \prod_{i=1}^{k} p_i^{\alpha_i - 1}(p_i - 1)$

This was 'Textbook' RSA !

→ There is internet version of it as well

⇒ Attacks on Textbook RSA :

(*) Small exponent attack ($e = 3$)

$m^3 \bmod N$ (with small message)

if $m < \sqrt[3]{N}$

$\therefore m \to \sqrt[3]{c}$

How to find cube root of no. ?

(Newton's method or smthg like that)

RSA assumption :

$\forall$ PPTM A,

$P\left[A(N, e, y) = x \text{ s.t. } y = x^e \bmod N\right] \leq \text{negl}(\log N)$

a large p & q randomly of equal length

(*) Common Modulus Attack

Same N, different e_i's → dangerous (not secure)

(e, d) for N (& $\therefore$ diff. d_i's)

$e = 3$ (small exponent for same message but different moduli).

$c_1 = m^3 \bmod N_1$

$c_2 = m^3 \bmod N_2$

$c_3 = m^3 \bmod N_3$

} we can get m from them

(*) Chinese Remainder Theorem (CRT)

"$x \equiv a_1 \pmod{p_1}$ (p_i, p_j → coprime)

$x \equiv a_2 \pmod{p_2}$

$x \equiv a_3 \pmod{p_3}$

→ Unique x upto $\prod_{i=1}^{k} p_i$ satisfies the above ($x \in [0 \ldots \prod p_i - 1]$)"

∴ we can calculate $m^3 \bmod (N_1 N_2 N_3) = t$ ($m < \sqrt[3]{N}$)

↳ m is ~~many~~ easy ($\sqrt[3]{t}$)

Proof :

$x_0 \equiv 0 \pmod{p_1}$

$x_0 \equiv 0 \pmod{p_2}$

$\vdots$

$x_0 \equiv 0 \pmod{p_k}$

} ⇒ $x = p_1 p_2 \cdots p_k$ but it does not belong to $[0 \ldots \prod p_i - 1]$

special care

$\therefore x = 0$

[∵ If $\forall i, a_i = 0 \Rightarrow x_0 \equiv 0$.

↳ in general $x_0 = m \cdot \prod_{i=1}^{k} p_i$]

→ If $x_1 \equiv 1 \pmod{p_1}$, $x_1 \equiv 0 \pmod{p_2}$, $x_1 \equiv 0 \pmod{p_k}$

↳ $p_1 \nmid x_1$.

$\Rightarrow x_1 = \left[\prod_{i=2}^{k} p_i^{e}\right] \cdot M$

$\therefore$ Find M s.t. $\left[\prod_{i=2}^{k} p_i^{e}\right] \cdot M \;\%\; p_1 = 1$

$\therefore x_1 = \left[\prod_{i=2}^{k} p_i^{e}\right]\left[\left(\prod_{i=2}^{k} p_i^{e}\right)^{-1} \bmod p_1\right]$

$\Rightarrow x_1 = \left\{\left[\prod_{i=2}^{k} p_i^{e}\right]\left[\left(\prod_{i=2}^{k} p_i^{e}\right)^{-1} \bmod p_1\right]\right\} \bmod \prod_{i=1}^{k} p_i^{e}$

→ If $x_1 \equiv 0 \pmod{p_1}$
$x_2 = 1 \pmod{p_2}$ $\therefore x_2 = \left\{\left[\frac{\prod_{i=1}^{k} p_i^{e}}{p_2}\right]\left[(\;)^{-1} \bmod p_2\right]\right\} \bmod \prod_{i=1}^{k} p_i^{e}$
$x_3 = 0 \pmod{p_3}$
⋮
$x_k = 0 \pmod{p_k}$

Similarly, we'll find $x_i \Rightarrow i^{th}$ one → 1
rest all → 0

$\therefore x \equiv a_1 \pmod{p_1} \equiv a_1 (1 \cdot \bmod p_1)$
$x \equiv a_2 \pmod{p_2} \equiv a_2 (1 \cdot \bmod p_2)$
⋮
$x \equiv a_k \pmod{p_k} \equiv a_k (1 \cdot \bmod p_k)$

$$x = \sum_{i=1}^{k} a_i \left[\left(\frac{\prod_{j=1}^{k} p_j^{e}}{p_i^{e}}\right)\left[\left(\frac{\prod_{j=1}^{k} p_j^{e}}{p_i^{e}}\right)^{-1} \bmod p_i^{e}\right]\right] \bmod \prod_{i=1}^{k} p_i^{e}$$

(*) Common modulus attack (but for only one person)

$\Rightarrow m^e \bmod N$

$ed = 1 \bmod \phi(N)$ $\Rightarrow \gcd(e,d) = 1$
$\Rightarrow \exists\ x, y.$
$ex + dy = 1$

$C_1 = m^{e_1} \bmod N$
$C_2 = m^{e_2} \bmod N$ we don't know d.
$\gcd(e_1, e_2) = 1$
$\exists\ x, y \Rightarrow e_1 x + e_2 y = 1$ → find x, y through extended euclidean algorithm

e_1, e_2 → public info.
but d is not

$C_1^{x} C_2^{y} \pmod N = m^{e_1 x} m^{e_2 y} \pmod N$
$= m^{e_1 x + e_2 y} \pmod N$
$= m$

PKCS #1.5 (Practical use of RSA)

→ Padded RSA :
$C = (r \,||\, m)^e \bmod N$
↳ random noise.

(Hardwore predicate for RSA → LSB) → ∴ r is prepended.
(" " " DLP → MSB) → ∴ r should be appended

r : min 64 bits of 8 bytes
↳ doesn't contain all 0 byte

$C = \{0000\,0000 \,||\, 0000\,0010 \,||\, r \,||\, 0000\,0000 \,||\, m\}^e \bmod N$
fixed

Must be used internally

∴ Now probabilistic encryption
∴ hope met → CPA secure
(yet to be proven
but definitely not CCA secure.

⇒ RSA OAEP
(Optimal asymmetric encryption padding)
Helps to find a way to pad to make it probabilistic → Try to make it CPA secure
Public key

★ Random Oracle model :

21st Feb

Overall review :

→ Kerckhoffs's principle.
↳ is mandatory; not just desirable to be followed.
→ Makes things 'impossible' and common sensical sol^ns aren't considered.

→ Designing secure systems is hard.
- Make and break of clerical ciphers.
- What does it (even) end?

→ Shannon's theory.
- defining perfect secrecy.
- Vernam cipher is perfect. (one-time pad)
- limitations of perfect secrecy.
$|K| \geq |M|$

→ 2 relaxations to Shannon.
- PPTM adversaries
- Negligible errors.

→ Bom are necessary & (almost) sufficient
→ With the assumption that one way func^s exist:
• sufficient for minicrypt
• trapdoor one-way func^ suffices for Cryptomania.

→ Minicrypt includes :
- PRGs
- Block ciphers.
- Stream ciphers
- CPA secure encryption.
- PRF
- MACs

- CCA secure encryption.
- Hashing (collision resistant)
- Bit commitment.
- Digital signatures. → OWF sufficient but not necessary for its existence
OWFs are often used to construct them.

→ DDH and RSA assumption. & is general trapdoor 1-way func^s.
→ Cryptomania includes : PKC & SKE
(and signatures)

future in this course → OT
↳ SMPC

Dining Philosophers problem

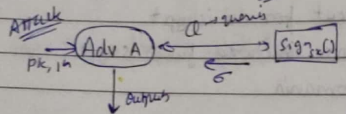
★ Digital Signatures

Gen (1^n) : $\langle pk, sk \rangle$

Sign : $Sign_{sk}(m) = \sigma$

$Vrfy_{pk}(m, \sigma)$: T/F

→ asymmetry of keys



$m \notin Q$
s.t. $Vrfy_{pk}(m, \sigma) = T$

pk → public key
sk → secret key.

$\forall$ PPTM A
$P[Vrfy_{pk}(m, \sigma) = T] \leq negl(n)$

unforgeable signature
against adaptively chosen attacks.

★ Textbook RSA signatures :

Public key : $\langle N, e \rangle$
Secret key : d
$Sign_d(m) = m^d \bmod N = \sigma$

σ could have generated only been generated by person using secret key ③

PAGE NO.
DATE

$Vrfy_{pk}(m, \sigma)$: $\sigma^e \bmod N \stackrel{?}{=} m$

Signature should not be dependent on message
↳ despite being a func" of msg → should not be dependent on msg.

Attacks:

① No message attack
→ without asking oracle, produce a forgery.
→ Choose random σ.
Compute $m = \sigma^e \bmod N$
$\langle m, \sigma \rangle$ pair
This will pass the verify test ($\sigma^e \bmod N \stackrel{?}{=} m$)
(but msg will be gibberish)

②

N, e → public, i.p $m^d \bmod N$

$\langle m, \sigma \rangle$ $\xrightarrow{Q_1}$ $\langle m_1, \sigma_1 \rangle$ → using oracle.

$m = \sigma^e \bmod N$

⇒ Set $m_2 = m/m_1 \xrightarrow{Q_2} \langle m_2, \sigma_2 \rangle$ d
↳ $m_2 \bmod N$

$\sigma = \sigma_1 \sigma_2 \bmod N$
$= (m_1 m_2)^d \bmod N$
$= m^d \bmod N$.

∴ σ is indeed a signature of m

∴ $\langle m, \sigma \rangle$ → passes the verification test & msg will no longer be gibberish (with just 2 queries to oracle).

∴

PAGE NO.
DATE

★ Hash-and-Sign Paradigm : (Performance of algo & security → both improve)

RSA signatures :

$Sign_{sk}(m) = (H(m))^d \bmod N = \sigma$

$Vrfy_{pk}(m, \sigma) = \sigma^e \bmod N \stackrel{?}{=} H(m)$

1st attack fails : ∵ Hash → OWF, we get $H(m)$ but not m.

2nd attack also fails : $H(m_1)H(m_2) \neq H(m_1, m_2)$

If $H \to TRF \to$ secure

~~If $H \to$ truly random function $\Rightarrow$ SI~~

⊙ Random Oracle model : (better than nothing)
(weaker)

better in terms of speed as well ∵ $|H(m)| < |m|$
⇒ lesser time to compute signature of $H(m)$ than m

→ Decentralization → Resource bound.
→ Quantum Mechanics → Physical impossibility
→ Noisy channel → Pragmatic ..
→ Asynchrony → Philosophical impossibilities
→ ~~Relativity~~ / Phy → Logical ..

PAGE NO.
DATE

24th Feb

★ Major sources of Impossibilities
(for "interference" with securing impossibility)

① ⇒ Resource Boundedness :

* TIME
  - PPTM, negl(n) error
  - DLP, DDH, (CDH, DH) → computational
  - RSA, I.F. (ASSO ONR)
  - Other former ones
  - (Post Quantum Crypto) - SVP (Shortest vector in a Lattice)
  - Decoding Random Linear ____,

* SPACE
  - Space bounded adversaries / cryptography.
* ENERGY
* COST
* RANDOMNESS

} Not popular yet.

② ⇒ Physical limits :

* QUANTUM UNCERTAINTY
  - No cloning
  - Indistinguishability
  - Heisenberg's Principle
  - etc
* RELATIVISTIC LIMITS :
  - Speed (limit) of Causality.
  - Space / Time dilations, etc

③ ⇒ Logical limits
- COMPUTATION
  - Undecidability / Incompleteness
  - Storage/Retrieval limits
  - Instruction density limits, etc.
- PRECISION limits
  - Accuracy
  - Robustness / Volatility

④ ⇒ Pragmatic limits
- CHANNEL NOISE
  - Founding crypto on noise
- TIMING ISSUES
  - (A) synchrony
- SYSTEM CRITICAL SECTION
  - Race conditions
  - Deadlocks, etc.
- FAULTS and FAILURES (Unpredictability)

⑤ ⇒ Philosophical questions
- TRUST
  - Game theoretic perspective. (like in Blockchain)
- EMERGENT PROPERTIES
  - Intelligence?
  (mainly) decentralization
  - Is security ...?
- ETHICS

- Background Processing
- PROOF SYSTEMS
- REPRE... THEORY, etc