NO CLASS

RSA

- "Textbook" RSA

- Some Attacks

- PKCs v1.5

- RSA signatures

Outline for today's class

and how the internet standards deal with it.

Textbook RSA

Gen: Choose 2 large distinct primes p, q and $N = pq$

choose (e, d) s.t. $ed \equiv 1 \mod [(p-1)(q-1)]$

Public Key: $\langle N, e \rangle$

Private Key: $\langle d, p, q \rangle$

Given we know Primality testing (Miller Rabin, etc), [Algorithms course]
(Kannan, et al)
(Monsoon 2018)

so we do not prove that this is efficiently computable.

Enc: $m \in \{0, 1, 2, \dots, N-1\}$

$C = m^e \mod N$

Dec: $c \in \{0, 1, 2, \dots N-1\}$

$m = c^d \mod N$

Efficient algos for both exist

doubling technique

Now for correctness, by Euler's theorem

$$a^{\phi(N)} \equiv 1 \mod N$$

Defn: $\phi(N)$ is the no. of nos. $< N$ that are co-prime to N

Euler's totient

Now, $(m^e)^d \mod N = m^{ed} \mod N$

Again, and now, $m^{\phi(N)} \mod N = 1$

and $\phi(pq) = (p-1)(q-1)$

and we know $ed \equiv 1 \mod N$

or $N \mid k(ed) + 1$

Textbook RSA is not CPA-secure

Now going back to Textbook RSA,

- it is deterministic

- so, is ciphertext-only secure, not CPA secure

> but in public key systems, notion of adversary does not have oracle access is meaningless. As it is public
>
> ⇓
>
> so, CPA is easily mountable
> ⇒ minimum security required for PKC is CPA-secure
>
> ⇒ there can be **no PKC system** that is **deterministic**, as
>   - deterministic ⇒ no CPA security
>   - PKC needs a min of CPA security.

Looking at other attacks

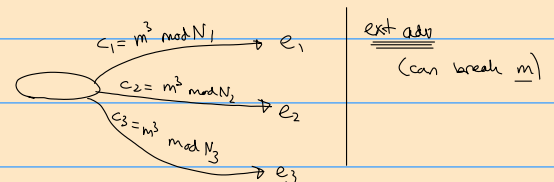⊙ **If $e=3$ (or small) and m is also small**

eg: $m < \sqrt[3]{N}$

$$c = m^3 \bmod N$$

$$\Rightarrow c = m^3$$

or $m = \sqrt[3]{c}$

⊙ **Same e with multiple people**

eg: $e=3$, N_1, N_2, N_3

$c_1 = m^3 \bmod N_1 \to e_1$
$c_2 = m^3 \bmod N_2 \to e_2$
$c_3 = m^3 \bmod N_3 \to e_3$

| ext adv (can break m)

Consider: $x = c_1 \bmod N_1$
$x = c_2 \bmod N_2$
$x = c_3 \bmod N_3$

CRT says ∃ unique x such that all 3 hold.

Unique $x \in [0, N_1 N_2 N_3 - 1]$ st
Chinese Remainder Theorem

Looking at the suggestions for modified RSA

"recommended by current Internet" : **PKCS v. 1-5**

((∃ no proof that this rec is secure))

((but ∃ other algos (nonstandard) that are provably secure))

Gen: • Same as regular RSA

• there are some restrictions on prime selection, but we can just say "any large primes should work"

we modify this to make it secure.
↓

Enc: Ciphertext $C = \left[\right]^e \bmod N$

$\log N$ bit no. / $(\log N)_8$ byte no. $\ddot{a}$ "K"-byte no.

Constructing the number

- **first byte** : fixed (0100 0000, he says from memory)
- **second "** : " all 0's
- then, at least **8 more bytes** are filled with random (not truly random — none can be all 0s)
- then, another set of 0000 0000
- then, message m (at most K-11 bytes)

for decryption:

- ignore first byte
- " second "
- " byte after byte after byte until we hit another all-0 byte.
- then, print the rest as message.

General Idea of Working

somehow, they decided

— first 16 MSBs are very weak ???

— at least 64 bits of random r recommended
  ↳ but not fixed. So message start unknown, cannot built easy algo
  ⇒ getting to hard-core predicate (RSA hc LSB) difficulty.

~~ this is modern internet.

$$\boxed{h(RSA) \text{ is } LSB}$$

Q: Can RSA be made provably secure?

A: Yes. RSA-OAEP (Optimal Assymetric Encryption Padding)

RSA based schemes are hard to prove. For provable we use discrete log based schemes, like El Gamal.

RSA-based Digital Signatures (we'll show deterministic → attacks → randomised)

Defining signatures

┌───┐
│ **Req:** • Adv should not be able to produce new m, σ │
│   without sk. │

$A \xrightarrow{\;m\;} B$ • auth data
                              • auth user

so

$A \xrightarrow{\;m, \sigma_A\;} B$ s.t. • σ Alg known to everyone
                                          • ∃ sign, uses **private key of A**
                                          • ∃ Verify, uses **publng of A** to verify signature done by A.

RSA-signature says,

sign (d, m) : $\boxed{m^d \bmod N = \sigma}$ → signature

verify (m, e, σ) = yes if $\boxed{\sigma^e \bmod N = m}$

} sign yourself with sk
  verify by others " pk.

Breaking Vanilla (Textbook) RSA-sign

I. • choosing m, then σ is hard.

|Algorithm|

• But, do reverse:

1. Choose random signature σ

2. Compute $\sigma^e \bmod N$

3. Send $\langle \sigma^e \bmod N, \sigma \rangle$ ⟶

Verify $(\sigma^e \bmod N, \sigma, (N,e))$ = yes.

II. Improved Attack

|Algorithm|

• suppose I have $\langle m_1, \sigma_1 \rangle$ sent by A.

• A also sent $\langle m_2, \sigma_2 \rangle$

• We produce third message that A never sent : $\langle m_1 m_2 \bmod N, \sigma_1 \sigma_2 \bmod N \rangle$

now, CPA from this :

we want to sign M
  • ask for σ for m_1
  • ask for σ for m_2 st $M = m_1 m_2$

} Done

→ instead of signing directly

New Paradigm:

Hash and Sign Paradigm

signature $\sigma = [H(m)]^d \bmod N$ | Yes if $\sigma^e \bmod N = H(m)$

Signing   Verifying

why do old attacks not work? |Collision resistance|

Proof: None for RSA

Rest of Course: a) A few more PKCs

b) Proof of CPA security for El-Gamal PKC

c) El-Gamal is not CCA-secure

d) CCA-secure PKCs

This should be done by mid-march

then sir will "talk about what we actually want to do"

"freedom to work, not freedom from work"

Digital signatures & Zero Knowledge Proofs

Q: How to Authenticate the user?
A: Ask user to prove they know the secret key.

Options: — Reveal the key ⟵————————————————— pretty useless

5 minutes of talking about this →

summary →

┌───┐
│ Problem: prove   you   know   a   secret   without   revealing   it. │
└───┘

Q: is it only for special kinds of knowledge (like sk of PKC) that can

be given ZKP for, or does ZKP exist for all kinds of knowledge.

More elaboration of problem

eg: • Suppose Alien says "I have winning strat. in chess"

• How would Alien prove this?

best: constructing a decision tree for every move ⎫ We do not know if short

Useless: Exponential time to construct ⎬ proof exists.

Proof too long to comprehend. ⎭

┌───┐
│ Aside: in complexity, compression problem?   PSPACE ≠ P ? │
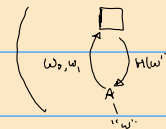└───┘

Again,

how to prove winning strat?
— Math proof impossible to show, if exists
— so, just play the game. Any loss/draw is proof of lie.
— with a few million games, the $P(\text{lie})$ is low

We will get back to this, hopefully.

eg2: Alien needs to prove it can see more colours.

Proof: randomized blind testing ⎛ w_0, w_1 ⟳ $H(w')$
                                    "w" $P(w = w') \leq \frac{1}{2} \pm \text{negl}$ ⎞
— any incorrect prediction is proof of Lie
—— with a few million games, the $P(\text{lie})$ is low

Q. Where is crypto in this?

A. Instead of using random assumption (like alien cannot see through body) as a "shield"

use cryptographic " (like OWF exist) as shield.

this class:
— Showing all knowledge canbe gamified such
— ZKP exists iff OWF exists
— ZKP ➙ DSigns
— DSigns do not need trapdoor OWF, as it is not PKC dependent.
— OWF ⟺ bit commitment

Concept of: **Bit Commitment**   ← same as blockchain

Binding ——→ ☐ b

Blinding ——▷ b is secret

eg: consider Use Case of Digital Auction.

— make a bid
    — nobody else can see it
    — you cannot modify it
    — others can verify it when you reveal it after bidding

Assume ∃ One-way permutation $f()$, building Bit-Commitment scheme as:

Commit bit (b):

Pick a random $\underline{s}$

Publish $\langle f(s), h(s) \oplus b \rangle$ eg: $\langle g^s \bmod p, MSB(s) \oplus b \rangle$

now, nobody knows anything about 's'

Reveal phase :

Give $\underline{b}$ and $\underline{s}$

Now, everybody can check : if $f(s)$, is unchanged

and if so, $h(s) \oplus (h(s) \oplus b) = b$?

∴ if DLP is a OWF, Bit- Commitment scheme exists

Something something NP complete Problem

Opaque ↑ locked box ☐

Assume such boxes exist. (replace with bit- commitment)

Show it is in language, NP?

Looking at problem: Graph 3 - Colouring ((is NP-complete, not proving))

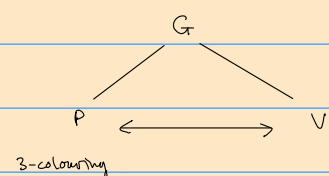given a graph $G = (V, E)$

is it 3- Colourable or not?   ∃ exp algos

((eg. question: is G tri-partite))   pol algos! NP-hard, so can't say (unlikely)

ZKP for graph 3-colouring:

a.k.a
Given a graph
    you know the answer (it is 3- colorable)
    ∴ do not want to reveal solution
    " " " prove to world you know solution

G

$P \longleftrightarrow V$

3-colouring

ZKP has to satisfy :   o Soundness condition
                            o Completeness condition

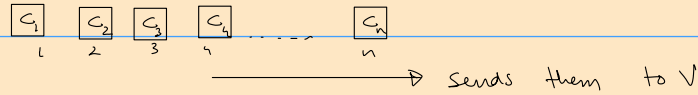((all "locked boxes" = bit commitment scheme))

ZKP Algo:

Prover does:

- Prover knows 3 colours RGB

→ Prover permutes the three colours

→ " creates n locked boxes, each with the colour of the corresponding vector

$\boxed{c_1}$ $\boxed{c_2}$ $\boxed{c_3}$ $\boxed{c_4}$ ...... $\boxed{c_n}$
  1   2   3   4   n

——————————————→ sends them to V

Verifier (V) does:

→ V picks one edge (i,j) at random and asks to reveal c_i and c_j

→ If $c_i = c_j$ Reject

                              (convinced)
If $c_i = c_j$ Accept if soundness $\wedge$ else repeat.

Verifying the proof

Completeness:
if G is 3-colourable,

V always accepts

Soundness:
if G is not 3-colourable,

V rejects with high P.

seeing how high,

$|E|$ edges $\Rightarrow$ rejection with Pr: $\dfrac{1}{|E|}$ reject.

$\left(1 - \dfrac{1}{|E|}\right)^k$ ← pr of acceptance with k trials.

At the end of this

V: - does not know any more about the solution

- is convinced P knows the solution

Now, we want to think of Dig. Sign based on DLP.
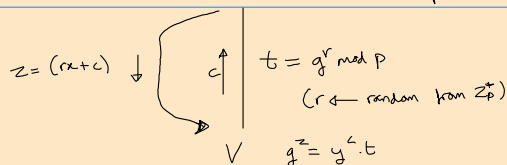
Considering interactive version

$$y = g^n \bmod p$$

we will use ZKP for DL to get Dig. Sign for DL.

ZKP for DLP:

P ← wants to prove given $(y, g, p) \Rightarrow x$

$z = (rx + c) \downarrow$ $c\uparrow$ $t = g^r \bmod p$
                                            ($r \leftarrow$ random from Z_p^*)

V $g^z = y^c \cdot t$

1. P chooses r, sends $t = g^r \bmod p$

2. V sends c (random from Z_p^*)

3. P sends $z = (cx+r)$

4. V checks if $g^z = y^c \cdot t$

Completeness: $g^{cx+r} = \left(g^x\right)^c \cdot g^r$

$= y^c \cdot t$

Soundness:

if P does not know x, pick x', r' s.t.

$P_r\left(cx'+r' = cx+r\right) = \frac{1}{p}$

↳ follows from,

DLP is one-way

This can act as signature scheme if **interactive signatures allowed**.

to get dig-sign (non interactive), we use **Random Oracle Model**

??? ASLA THREYA

: if hash functions are random output, then oracle ???

If **assume hash functions give random outputs,**

P r @ random, $\underbrace{H(g,p,y) = c}$

$z = \left(H(g,p,y) \cdot x + r\right)$ ↓   ↓ $t = g^r \bmod p$ public info

↓

V   Accepts if $\boxed{g^z = y^{H(g,p,y)} \cdot t}$
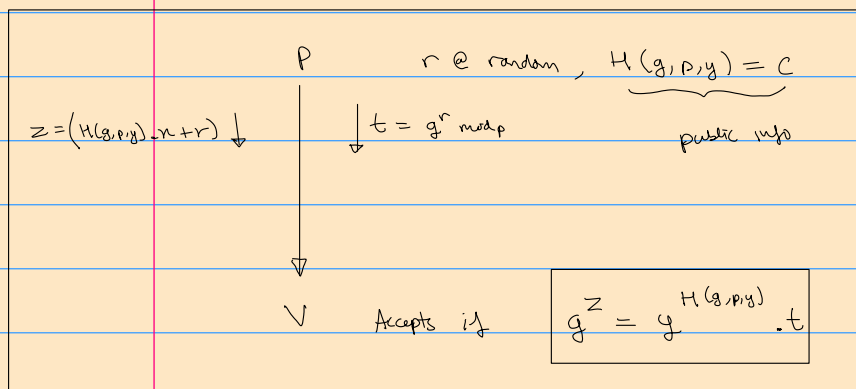
← this is a signature scheme under this assumption
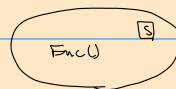
(Schloss? signature)

10.3.20

Secret Sharing

→ the problem

→ Shamir's solution

→ A Generalisation

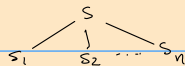→ Beautiful Relation to Computing

(SOME OPEN PROBLEMS)

PROBLEM:

Secret key S

an adversary need not control system, just (access to) key S.

Enc() $\boxed{S}$

• if **key is in a single location, there is no fault-tolerance**

• to leverage **adv. of dist. sys,** fault-tolerance: **key needs**

to be **shared**

— but then **key is no longer secret**

Need secret sharing:

S

s_1 s_2 ... s_n

$0 < t < n$

• Any **$> (t-1)$** s_i's can reconstruct S

• Any **$< t$** s_i's have **no information about S**

Shamir's Secret Sharing:

Define $s \in \mathbb{F}$ (for eg. Z_p, prime p)

$\boxed{Q(x) = \sum_{j=0}^{t} r_j x^j}$ $Q(0) = r_0 = S$

$r_{j>0} \xleftarrow{\text{rand}} \mathbb{F}$

$\alpha_1, \alpha_2, \ldots \alpha_n$

we distinct public elements of $\mathbb{F}$

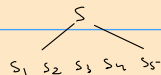$$S_i = Q(\alpha_i)$$

General Access Structure

$$\mathbb{A} \subseteq 2^{[1,n]}$$

- All subsets $B \in \mathbb{A}$ can access S

- Others should have no info about S

$$B \in A \quad \text{if } |B| \geq t-1 \quad \text{else } B \notin A$$

$$A = \{ B \mid |B| \geq t-1 \}$$

eg. $n = 5$

```
        S
      / |  \
   S₁ S₂ S₃ S₄ S₅
```

$S_1 \, S_2 \, S_3 \, S_4 \, S_5$

when $1,2$ can reconstruct secret, every superset can automatically access it to

$$A = \{ \{1,2\}, \{3,4,5,6\} \}.$$

(monotone supersonic property)

$\{1,2\}, \{3,4,5,6\}$ can access but not $\{2,3\}$.

‖ I have entirely lost him here ‖

take degree $t = 3$ and 8 points on the polynomial.

Give 2 points to $\underline{1}$, 2 points to 2 and 1 each to rest of $3, 4, 5$ and 6.

So the $A = \{ \underbrace{\{1,2\}}_{\nearrow 4 \text{ points}}, \underbrace{\{3,4,5,6\}}_{\nearrow 4 \text{ points}}, \underbrace{\{1, 3, 4\}}_{\nearrow 4 \text{ points}} \cdots \}.$

> Will weights always work? Answer is NO in general but there are cases where YES but exponential number of weights required.

$\mathbb{A} \rightarrow$ access structure

$$\left(\begin{array}{l} f : \{0,1\}^n \rightarrow \{0,1\} \\ \mathbb{A}_f = \{ B \mid f(B) = 1 \} \end{array} \right.$$

$\underset{\longleftrightarrow}{0110 \cdots 1}$
$n-\text{bit}$

$\downarrow$
subsets

there is a canonical relation between 2 bit strings and subsets.

For any access structure, there will be a corresponding boolean functions. Boolean functions are not monotone. They will be monotone if they can be implemented using only AND and OR gates (monotone function theory).
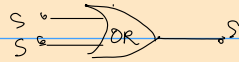
$A = \{\{1,2\}\}$ Give r to one
$\widetilde{A} = \{\phi, \{1\}, \{2\}\}$ and $r-s$ to other

$A = \{\{1\}, \{2\}, \{1,2\}\}$ give s to both.

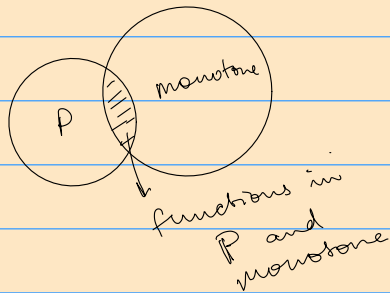$\mathbb{A} = \{\phi\}$

$x_1 = r_1$
$x_2 = s - r_1$
$x_3 = r_2$
$x_4 = s - r_2$
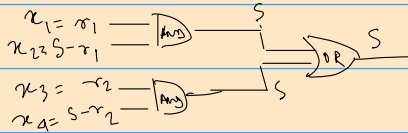
Given $A \subseteq 2^{[1..n]}$

Step 1: Model it as boolean function $f_A : \{0,1\}^n \to \{0,1\}$

Step 2: Design an AND-OR Boolean circuit for f_A

Step 3: Build the secret sharing scheme level-by-level.

functions in
P and
monotone

$mP \Rightarrow$ poly sized AND-OR circuit

$P \Rightarrow$ poly sized AND-OR-NOT circuit

$$mP \overset{?}{=} P \cap \text{Monotone}$$

$\sharp$ Example: Perfect Matching.

what to do if $A \in P \cap$ Monotone but $A \notin mP$?

$\forall B \in A$, poly-time reconstruction algorithm

$\forall B \notin A$, $\forall$ PPTM D $P[D(\text{shares } B) = S] \leq \text{negl}(n)$.

$|S| = \log |A|$ bits
$|S_i| = |S|$ $n \log |F|$ bits.

This can be broken

Secret S, $Enc_K(S) = C \longrightarrow$ Information Dispersal Algorithm   or   Reed-Solomon Code $\geq t+1$

Rabin's

small $\to$ Shamir secret sharing

any $t+1$ should be able to access.

Complexity: $\sqrt{|K|} + \dfrac{n}{t+1} * \log |F|$

OPEN / Smallest computational secret sharing? $A \in P \cap$ Monotone but $A \notin mP$?
NO known poly-sized soln EXISTS!