28.02.2020

NO CLASS

03.03.2020

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. $ed \equiv 1 \bmod [(p-1)(q-1)]$

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

$C = m^e \bmod N$

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

$m = c^d \bmod N$

Efficient algos for both exist

doubling technique

Now for correctness, by Euler's theorem

euler's theorem

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

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

Euler's totient

euler's totient

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

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

euler's totient of N=pq

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

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

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

TODO COMPLETE

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

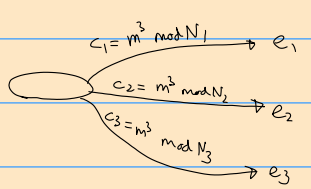
$c = m^3 \bmod N$

$\Rightarrow c = m^3$

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

⊙ Same e with multiple people

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



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 $x \in [0, N_1 N_2 N_3 - 1]$ s.t.

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 $C = [\quad]^e \bmod N$

$\log N$ bit no. / $(\log N)_8$ byte no. of "K"-byte no.

Constructing the number:

- first byte : fixed (01000000, 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 00000000

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 build easy algo

⇒ getting to hard-core predicate (RSA ka LSB) difficulty.

h(RSA) is LSB

~~ this is modern internet.

Q: Can RSA be made provably secure?

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

Kannan Trivia

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

RSA-based Digital Signatures (will show deterministic → attacks → randomized)

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 public of A to verify signature done by A.

RSA – signature says,

sign (d, m) : $m^d \bmod N = \sigma$ ← signature

verify $(m, e, \sigma) =$ yes if $\sigma^e \bmod N = m$

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 $\langle \sigma^e \bmod N, \sigma \rangle \longrightarrow$

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

s.t. $M = m_1 m_2$ } Done

New Paradigm: → instead of signing directly

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"

06.03.2020

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 →

Problem: prove you know a secret without revealing it.

summary →

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 proof exists.

Useless: Exponential time to construct

Proof too long to comprehend.

Aside: in complexity, compression problem? PSPACE ≠ P?

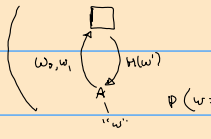
Again,

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(lie) is low

We will get back to this, hopefully.

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

Proof: randomized blind testing



- any incorrect prediction is proof of Lie
- with a few million games, the P(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 can be gamified such
- ZKP exists iff OWF exists
- ZKP → Digisigs
- Digisigs 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 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 b and 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 — this is something complex NP 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?

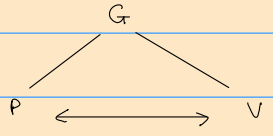
∃ exp algos

((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-colourable)
" do not want to reveal solution
" " " prove to world you know solution)



3-colouring.

ZKP has to satisfy:

- Soundness condition
- Completeness condition

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

ZKP Algo:

Prover does:

- Prover knows 3 colours R G B

→ Prover permutes the three colours

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

→ 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

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

Verifying the proof

Completness: 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 ⇒ rejection with Pr: $\frac{1}{|E|}$ reject.

$\left(1 - \frac{1}{|E|}\right)^k$ ← 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

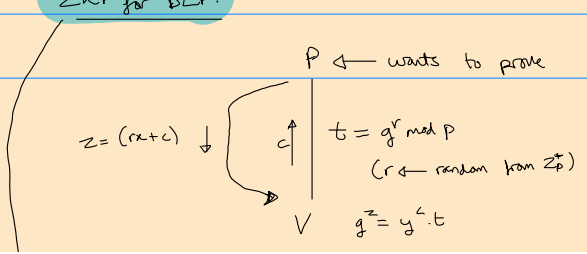
Now, we want to think of Dig. Sign based on DLP.

Considering interactive version

$y = g^x \bmod p$

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

ZKP for DLP:



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 $g^z = y^c . t$

Completeness: $g^{cx+r} = (g^x)^c . g^r = y^c . t$

Soundness: if P does not know x, pick r', r' s.t $P_r(cx'+r' = cx+r) = \frac{1}{p}$

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:

??? ASHLA THEOREM

: if hash functions are random outputs, then oracle.?.?.

If assume hash functions give random outputs,

P — r @ random, $H(g, p, y) = c$ (public info)

$z = (H(g,p,y).x + r)$ ↓ ; $t = g^r \bmod p$ ↓

V — Accepts if $g^z = y^{H(g,p,y)} . t$

← 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(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 $S \in F$ (for eg. Z_p, prime p)

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

$Q(0) = r_0 = S$

$r_{j>0} \leftarrow$ F rand

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

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

$S_i = Q(\alpha_i)$

General Access Structure

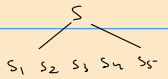
$\mathbb{A} \subseteq 2^{[1,n]}$

- All subsets $B \in \mathbb{A}$ can access S
- Others should have no info about S

$B \in A$ if $|B| \geq t-1$ else $B \notin A$

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

eg: $n=5$



When 1,2 can reconstruct secrets, every superset can automatically access it (monotone supersonic property)

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

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

So the $\mathbb{A} = \{ \{1,2\}$ (4 points), $\{3,4,5,6\}$ (4 points), $\{1,3,4\}$ (4 points) $\cdots \}$.

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

$\mathbb{A}$ → access structure

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

$\mathbb{A}_f = \{ B \mid f(B) = 1 \}$

0110 ..... 1 ⟷ n-bit ↕ subsets

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

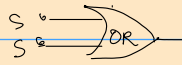
For any access structure, there will be a corresponding boolean functions. Boolean functions are not monotone. They will be monotone if they can be implemented using only AND and OR gates (monotone function theory).

Conversion From AND, OR circuits for A to secret sharing schemes for A

$A = \{\{1,2\}\}$

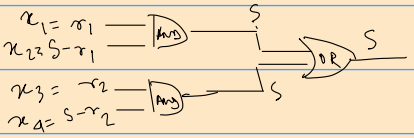
$\bar{A} = \{\phi, \{1\}, \{2\}\}$

Give r to one and r − s to other



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

$A = \{\phi\}$

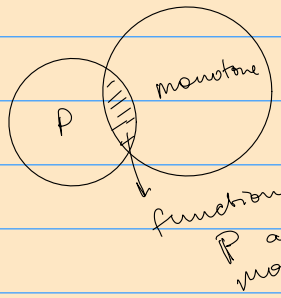


Given $A \subseteq 2^{[1..n]}$

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.



mP → poly sized AND-OR circuit

P → poly sized AND-OR-NOT circuit.

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

$\neq$ 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 $P[D(\text{shares } B) = s] \leq \text{negl}(n)$.

$|S| = \log |F|$ bits

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

This can be broken

Secret S, $\text{Enc}_K(S) = C$ → Rabin's Information Dispersal Algorithm or Reed-Solomon code $\geq t+1$

K small → Shamir secret sharing

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

Complexity: $n|K| + \frac{n}{t+1} \cdot \log |F|$

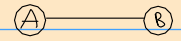
OPEN: Smallest computational secret sharing? $A \in P \cap$ Monotone but $A \notin mP$? NO known poly-sized soln EXISTS!

Oblivious Transfer (OT) and Secure Two-Party Computation

(General)

13.3.2020

OT (specific Problem)



index i
Output $= b_i$
Shouldn't know $b_{j \neq i}$

Array $B = [b_1, b_2, \ldots b_n]$
Shouldn't know i

(How to query DB without DB revealing anything other than query without knowing query)

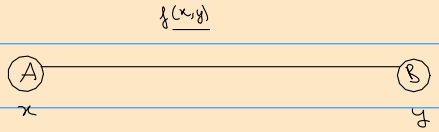
Kinds of (unclear objectives)

impossible . to prove
but if PKC exists, provable

logically impossible (intuitively)
(can formalise)

Showing that solutions exist if PKCs exist (trapdoor OWFs exist)

Secure General Two-Party Computation (General version)



goal: Compute some $f(x,y)$, but
A does not want to reveal x
B does not want to reveal y

((famous formulation: _____ millionaire problem))

Showing that OT is a special case of Secure Gen 2-party Comms

i $\qquad$ $B = [\ \ ]$

$f(i, B) = B[i]$

But it is a "special case", like

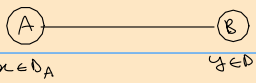
if OT solution is given to you, you can solve any SG2-pc

Showing:

$h(i, B) = B[i]$

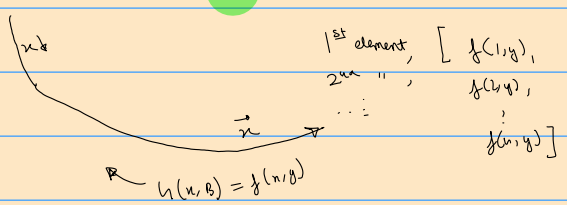
$f(x,y)$ $\qquad$ $f: D_A \times D_B \to D_C$

Suppose



Solving as

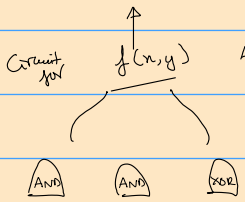
1. WLOG, single bit answer ASSUMPTION :
2. WLOG, $D_A \to [1, \ldots n]$
3. Now, B builds array as follows:
4. now, thesis OT :

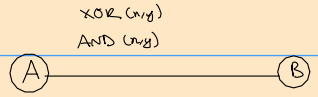


10 minute aside on why arrays are programs.

⇒ if OT can be solved, SG2PC can be solved.

Now, similarly to last class, we are only interested in functions f that are efficiently computable (poly-efficient circuits)



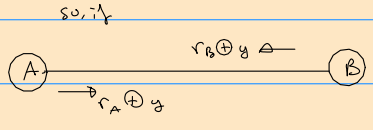


Input: $x \in \{0,1\}$ $y \in \{0,1\}$

Output: $z_A \in \{0,1\}$ $z_B \in \{0,1\}$

$z_A \oplus z_B = f(x,y)$

Start with secret sharing of inp-x and secret sharing of inp-y and ??



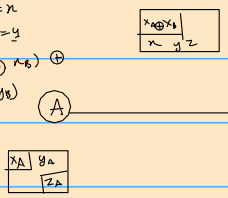
$z \leftarrow x \oplus y$

$x_A \oplus x_B = x$

$y_A \oplus y_B = y$

$z_A \oplus z_B = (x_A \oplus x_B) \oplus (y_A \oplus y_B)$

$(x_A \oplus y_A) \oplus (x_B \oplus y_B)$



if able to do XOR AND then we are done

$z_A \oplus z_B = f(x,y)$

Let us see protocols for XOR and AND

Do it the circuit way.

Can together simulate trusted third Party.

Don't build array

Protocol for XOR

A: $z_A \leftarrow x_A \oplus y_A$

B: $z_B \leftarrow x_B \oplus y_B$

B defines $z_B \xleftarrow{R} \{0,1\}$

B creates an array

x_A, y_A	
0,0	$z_B \oplus (x_B \wedge y_B)$
0,1	$z_B \oplus (x_B \wedge \bar{y}_B)$
1,0	$z_B \oplus (\bar{x}_B \wedge y_B)$
1,1	$z_B \oplus (\bar{x}_B \wedge \bar{y}_B)$

b_{00} b_{01} b_{10} b_{11}

z_A

OT protocol

Step 1: A sends a random array $A = [r_1, r_2, \ldots (x_i) \ldots r_n]$ to B

$enc_B(x_i) = r_i^e \bmod N$ (replaced by encrypted r_i)

Step 2: B decrypts the entire array A $D = [dec_B(r_1), \ldots x_i, \ldots dec_B(r_n)]$

Add $D + B$, $DB = [\forall j \; dec(r_j) \oplus b_j]$

B sends DB to A

Step 3: A obtains $b_i = (DB[i] \oplus r_i)$

F(x) $\phi(x)$: [XOR of random subset of bits of x]