Great Theoretical Ideas in CS

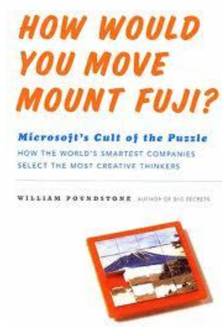
V. Adamchik CS 15-251

Lecture 24 Carnegie Mellon University

Cryptography and RSA

Upcoming Interview?

- How the World's Smartest Company Selects the Most Creative Thinkers



Outline

Groups
Generators
Euler's theorem
Fermat's little theorem
Diffie-Hellman Key Exchange
RSA algorithm

$Z_n = \{0, 1, 2, \ldots, n-1\}$

Define $+_n$:
$a +_n b = a + b \pmod{n}$

Define $-_n$:
$a -_n b = a +_n (-b)$

$(-b)$ is an additive inverse
$b +_n (-b) = 0$

Special element 0 is called an identity element

Group (1854, Cayley)

A group G is a pair $(S,\blacklozenge)$, where S is a set and $\blacklozenge$ is a binary operation $S \times S \to S$ such that:

1. (Closure) For all a and b $\in S$, $a\blacklozenge b \in S$

2. $\blacklozenge$ is associative, $(a\blacklozenge b)\blacklozenge c= a\blacklozenge(b\blacklozenge c)$

3. (Identity) There exists an element $e \in S$ s.t.

   $e \blacklozenge a = a \blacklozenge e = a$, for all $a \in S$

4. (Inverses) For every $a \in S$ there is $b \in S$ s.t.

   $a \blacklozenge b = b \blacklozenge a = e$

Commutative or "Abelian" Groups

If $G = (S,\blacklozenge)$ and $\blacklozenge$ is commutative, then

G is called a commutative group

remember, "commutative" means

$a \blacklozenge b = b \blacklozenge a$ for all a, b in S

Some examples…

$(Z_n, +_n)$ is a group

$(Z, +)$ is a group

$(N, +)$ is not a group

$(Z_n, *_n)$ is not a group

$Z^*_n = \{x \in Z_n \mid GCD(x,n) = 1\}$

$(Z^*_n, *_n)$ is a group

Some properties of groups…

Identity Is Unique

Theorem: A group has exactly one identity element

Proof:

Suppose $e \neq f$ are both identities of $G=(S,\blacklozenge)$

Then $f = e \blacklozenge f = e$. Contradiction!

We will always denote an identity by e.

Inverses Are Unique

Theorem: Every element in a group has a unique inverse

Proof:

Suppose $b \neq c$ are both inverses of a.

Then $b = b \blacklozenge e = b \blacklozenge (a \blacklozenge c) = (b \blacklozenge a) \blacklozenge c = c$

Contradiction!

Order of a group

A group $G=(S,\blacklozenge)$ is finite if S is a finite set

Define $|G| = |S|$ to be the order of the group (i.e. the number of elements in the group)

What is the group with the least number of elements?

$G = (\{e\},\blacklozenge)$ where $e \blacklozenge e = e$

$Z_n = (\{1\}, +_n)$

Z_n can be generated by a single element

Generators

An element $g \in S$ is called a generator of $G=(S,\blacklozenge)$ if the set $\{g\}$ generates G

A set $T \subseteq S$ is said to generate the group $G = (S,\blacklozenge)$ if every element of S can be expressed as a finite combination of elements in T under $\blacklozenge$.

A group G is cyclic if it is generated by a single element. A cyclic group can have more than one generator.

$(Z_n, +_n)$ is cyclic
$(Z, +)$ is cyclic

More generators for $(Z_n,+_n)$

Consider $(Z_4,+_4)$

2+0=2; 2+2=0; 2+2+2=2; 2+2+2+2=0

3+0=3; 3+3=2; 3+3+3=1; 3+3+3+3=0

3 is a generator, but 2 is not.

Claim: Any $a \in Z_n$ s.t. $GCD(a,n)=1$ generates $(Z_n,+)$

Def: The order of an element g is the least k s.t. $g^k = e$

$Z^*_n = \{x \in Z_n \mid GCD(x,n) = 1\}$

$g \in (Z^*_n, *_n)$ is a generator if the powers of g hit every element of Z_n

This will mean that Z_p^* has an alternative representation as the powers of g:
$\{g, g^2, g^3, \ldots, g^{p-1}\}$.

Example, $(Z^*_n, *_n)$

$Z_7^* = \{1,2,3,4,5,6\}$

$2^0 = 1;\ 2^1 = 2;\ 2^2 = 4;\ 2^3 = 1$

$3^0=1;\ 3^1 = 3;\ 3^2 = 2;\ 3^3 = 6;\ 3^4 = 4;\ 3^5 = 5;\ 3^6 = 1$

3 is a generator, but 2 is not.

5 is another generator

Generator Theorem:

If p prime, then $(Z^*_n, *_n)$ has a generator g.

In fact, it has $\phi(p-1)$ generators.

Proof is not given here.

Open Problem (Gauss)

Is there an efficient algorithm, given a prime p, to find a single generator in Z_p^* ?

That is, for every integer $a \in \mathbf{Z}_p^*$, find an integer k such that $g^k \equiv a \pmod{p}$.

Such k is called the discrete logarithm of a to the base g modulo p.

Euler Phi Function $\phi(n)$

$\phi(n)$ = size of Z_n^*

p prime $\Rightarrow \phi(p) = p-1$

p, q distinct primes $\Rightarrow$
$\phi(p\ q) = (p-1)(q-1)$

Fundamental lemma of powers.

If $a \in Z_n^*$ then $a^x \equiv_n a^{x \bmod \phi(n)}$

$5^{121242653} \pmod{11} = 5^{121242653 \pmod{10}} \pmod{11}$
$= 5^3 \pmod{11} = 4$

Note, $a^{\phi(n)-1} \equiv_n a^{-1}$
This can be used to compute a^{-1}.

for $a \in Z_n^*$, $a^x \equiv_n a^{x \bmod \phi(n)}$

Hence, we can compute
$a^m \pmod n$
while performing at most
$2 \lfloor \log_2 \phi(n) \rfloor$ multiplies

where each time we multiply
together numbers
with $\lfloor \log_2 n \rfloor + 1$ bits

Euler's Theorem

For $a \in Z_n^*$, $a^{\phi(n)} \equiv_n 1$

Note, $a^{\phi(n)-1} \equiv_n a^{-1}$
This can be used to compute a^{-1}.

Proof of Euler's Theorem

Define $a Z_n^* = \{a *_n x \mid x \in Z_n^*\}$ for $a \in Z_n^*$

By the permutation property, $Z_n^* = aZ_n^*$

$\prod x \equiv_n \prod ax$ [as x ranges over Z_n^*]

$\prod x \equiv_n \prod x$ ($a^{\text{size of } Zn^*}$) [Commutativity]

$1 =_n a^{\text{size of } Zn^*}$ [Cancellation]

$\boxed{a^{\phi(n)} =_n 1}$

Fermat's Little Theorem

If n is prime, $a \in Z_n^* \Rightarrow a^{n-1} \equiv_n 1$

Note, $a^{n-2} \equiv_n a^{-1}$
This can be used to compute a^{-1}.

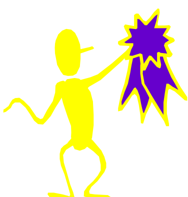
Basic Cryptography

Cryptography

Cryptography is the mathematics of devising secure communication systems

Cryptanalysis is the mathematics of breaking such systems.

RSA Cryptography



Basically unbreakable method for encoding messages

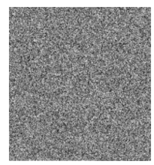
One Time Pads

CIA, KGB, Mossad, Mi6 in "old" times

19456 A
34139 Aardvark
03458 Able
34347 ...

96350 Apple
67295 ...

A code book



A sheet of random numbers

Convert a message according to a code book and then XOR with a sequence of random numbers.

One Time Pads

Gives perfect security!
For random shared key,
leaks no information about message

To be able to read a message encrypted like this,
the receiver has to know
the details of the encryption process.

Agreeing on a secret

One time pads rely on having a shared secret!

Alice and Bob have never talked before
but they want to agree on a secret…

How can they do this?

Diffie-Hellman Key Exchange (1976)

Suppose we have two people wishing to communicate: Alice and Bob.

They do not want Eve (eavesdropper) to know their message.

Alice and Bob agree upon and make public two numbers prime p, and a generator g in Z_p^*

p and g are public!

Diffie-Hellman Key Exchange (1976)

Alice chooses a random $a \in Z_p^*$ and computes $g^a \pmod p$ and sends it to Bob.

Bob chooses a random $b \in Z_p^*$ and computes $g^b \pmod p$ and sends it to Alice.

Punchline: Now both Alice & Bob can compute the "shared secret" $m = g^{ab} \pmod p$

What about Eve?

If Eve wants to compute g^{ab} she needs either a or b

Otherwise, she needs to compute $g^{ab} \pmod p$ directly.
This is so-called a discrete logarithm problem:
Solve for in x for $y = g^x \pmod p$, given y, g and p,

There is no algorithm to accomplish this in a reasonable amount of time.

Diffie Hellman requires both parties to exchange information to share a secret, so Eve might intercept…

can we get rid of this assumption?

Public Key Cryptography

Goal: Enable Bob to send encrypted message to Alice *without their sharing any secret*

Anyone should be able to send Alice a message in encrypted form.

Only Alice should be able to decrypt.

Anyone can send Alice a message in encrypted form.
Only Alice should be able to decrypt.
HOW ???

Alice holds a special "secret key" or "trapdoor info" that enables her to decrypt

Physical analogy: key to a locked box

Alice holds a "secret key" that enables her to decrypt – aka a key to a locked box

Encryption (Physical analogy):
Place message in a locked box with a "lock" that Alice's key can open.

How to get hold of such lock?

Alice gives it to everyone!!

Alice has a *"public key"* known to everyone which can be used for encryption, and a "private key" for decryption.

Public Key Crypto

Alice generates public (P) and private (S) keys.

Encryption of message m: c = Enc(m, P)

Anyone can encrypt (as P is public)

Decryption of ciphertext c: Dec(c, S)

Alice knows a secret key S, so can decrypt.

Of course, must have Dec(Enc(m,P),S) = m

The RSA Cryptosystem (1977)


Rivest


Shamir


Adleman


Euler


Fermat

Pick secret, random large primes: p,q
Multiply n = p*q
"Publish": n

$\phi(n) = (p-1)*(q-1)$
Pick random $e \in Z^*_{\phi(n)}$
"Publish": e

Compute d = inverse of e in $Z^*_{\phi(n)}$
Hence, $e*d = 1$ [mod $\phi(n)$]
"Private Key": d

p,q random primes
e random $\in Z^*_{\phi(n)}$
n = p*q
$e*d = 1$ [mod $\phi(n)$]

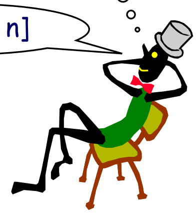
(n,e) is my public key. Use it to send me a message.

p,q prime, e random $\in Z^*_{\phi(n)}$
n = p*q
$e*d = 1$ [mod $\phi(n)$]

n, e

Message m

m^e [mod n]



$(m^e)^d \equiv_n m$

RSA Example

☺

$n = 187 = 11*17$
$e = 7 \in Z^*_{160}$

S M I L E Y
19 13 09 12 05 2

From a codebook

Compute message $m^e (\mathrm{mod}\ n)$
$19^7 = 145 \bmod 187$
$13^7 = 106 \bmod 187$

RSA Example

☺

$n = 187 = 11*17$
$e = 7 \in Z^*_{160}$

S M I L E Y
19 13 09 12 05 25

Compute message $m^e (\mathrm{mod}\ n)$
$19^7 = 145 \bmod 187$
$13^7 = 106 \bmod 187$

RSA Example

☺

$n = 187 = 11*17$
$e = 7$

	S	M	I	L	E	Y
m	19	13	09	12	05	25
m^e mod n	145	106	70	177	146	185

$m^{e*d} \bmod n$, where $d = e^{-1} = 23$

$145^{23} = 19 \bmod 187$

How hard is cracking RSA?

If we can factor products of two large primes, can we crack RSA?

p,q random primes
e random $\in Z^*_{\phi(n)}$
$n = p*q$
$e*d = 1\ [\ \mathrm{mod}\ \phi(n)\ ]$

If we know n and $\phi(n)$, can we crack RSA?

$(m^e)^d \equiv_n m$

Cracking RSA (125-dec.digit)

Team from Bellcore and MIT solved (in 1993-1994) this by using 1600 computers (over the internet) within 8 months.

THE MAGIC WORDS ARE SQUEAMISH OSSIFRAGE

Cracking RSA

The current record:

RSA-768
(232 dec.digits):
Dec., 2009

RSA example

1. $p = 61$, $q = 53$
2. $n = 3233$, $\phi(n) = 60*52 = 3120$
3. $e = 37$ (there are many to choose from)
4. EEA: $d = 253$
   since $1 = (-3)*3120+253*37$

Public key (3233, 37)
Private key 253

Send: $c = m^{37} \bmod 3233$
Read: $m = c^{253} \bmod 3233$

Here's What You Need to Know…

Groups
Generators
Euler's theorem
Fermat's little theorem
Diffie-Hellman Key Exchange
RSA algorithm