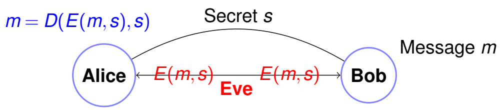
Public-Key Cryptography

1. Cryptography $\Rightarrow$ relation to Bijections
2. Public-Key Cryptography
3. RSA system
   - 3.1 Efficiency: Repeated Squaring.
   - 3.2 Correctness: Fermat's Little Theorem.
   - 3.3 Construction.

Cryptography ...



What is the relation between D and E (for the same secret s)?

Excursion: Bijections.

$f : S \to T$ is **one-to-one mapping**.
one-to-one: $f(x) \neq f(x')$ for $x, x' \in S$ and $x \neq x'$.
$f(\cdot)$ is **onto**
if for every $y \in T$ there is $x \in S$ where $y = f(x)$.

Bijection is one-to-one and onto function.
Two sets have the same size
if and only if there is a bijection between them!

Same size?
$\{red, yellow, blue\}$ and $\{1,2,3\}$?
$f(red) = 1$, $f(yellow) = 2$, $f(blue) = 3$.
$\{red, yellow, blue\}$ and $\{1,2\}$?
$f(red) = 1$, $f(yellow) = 2$, $f(blue) = 2$.
two to one! not one to one.
$\{red, yellow\}$ and $\{1,2,3\}$?
$f(red) = 1$, $f(yellow) = 2$.
Misses 3. not onto.

Modular arithmetic examples.

$f : S \to T$ is **one-to-one mapping**.
one-to-one: $f(x) \neq f(x')$ for $x, x' \in S$ and $x \neq y$.
$f(\cdot)$ is **onto**
if for every $y \in T$ there is $x \in S$ where $y = f(x)$.

Recall: $f(red) = 1$, $f(yellow) = 2$, $f(blue) = 3$
One-to-one if inverse: $g(1) = red$, $g(2) = yellow$, $g(3) = blue$.

Is $f(x) = x+1 \pmod{m}$ one-to-one? $g(x) = x-1 \pmod{m}$.
Onto: range is subset of domain.
Is $f(x) = ax \pmod{m}$ one-to-one?
If $\gcd(a,m) = 1$, $ax \neq ax' \pmod{m}$ for $x \neq x'$.

Injective? Surjective?
We tend to use one-to-one and onto.

Bijection is one-to-one and onto function.
Summary: Two sets have the same size
if and only if there is a bijection between them!

Inverses: continued.

Claim: $a^{-1} \pmod{m}$ exists when $\gcd(a,m) = 1$.

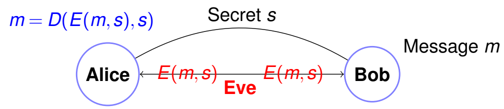
Fact: $ax \neq ay \pmod{m}$ for $x \neq y \in \{0, \ldots m-1\}$

Consider $T = \{0a \pmod{m}, 1a \pmod{m}, \ldots, \ldots (m-1)a \pmod{m}\}$
Consider $S = \{0, 1, \ldots, \ldots (m-1)\}$
$S = T$. Why?
$T \subseteq S$ since $ax \pmod{m} \in \{0, \ldots, m-1\}$
One-to-one mapping from S to T!
$\implies |T| \geq |S|$
Same set.

Why does a have inverse? T is S and therefore contains 1!

What does this mean? There is an x where $ax = 1$.
There is an inverse of a!

Back to Cryptography ...



What is the relation between D and E (for the same secret s)?
D is the inverse function of E!
Example:
One-time Pad: secret s is string of length $|m|$.
$E(m,s)$ – bitwise $m \oplus s$.
$D(x,s)$ – bitwise $x \oplus s$.
Works because $m \oplus s \oplus s = m$!
...and totally secure!
...given $E(m,s)$ any message m is equally likely.

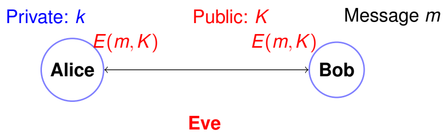
Disadvantages:

Shared secret!

Uses up one time pad..or less and less secure.

Public key cryptography.

$m = D(E(m,K),k)$



Everyone knows key K!
Bob (and Eve and me and you and you ...) can encode.
Only Alice knows the secret key k for public key K.
(Only?) Alice can decode with k.

Is public key crypto unbreakable?

We don't really know....but we do it every day!!!

RSA (Rivest, Shamir, and Adleman)

Pick two large primes p and q. Let $N = pq$.
Choose e relatively prime to $(p-1)(q-1)$.[1]
Compute $d = e^{-1} \mod (p-1)(q-1)$. d is the private key!
Announce $N (= p \cdot q)$ and e: $K = (N,e)$ is my public key!

Encoding: $E(x)$ is $\mod (x^e, N)$.

Decoding: $D(y)$ is $\mod (y^d, N)$.

Does $D(E(m)) = m^{ed} = m \mod N$?

Yes!

[1] Typically small, say $e = 3$.

Example: $p = 7$, $q = 11$.

$N = 77$.
$(p-1)(q-1) = 60$

Choose $e = 7$, since $\gcd(7,60) = 1$.
How to compute d? egcd(7,60).

$7(-17) + 60(2) = 1$

Confirm: $-119 + 120 = 1$

$d = e^{-1} = -17 = 43 = \pmod{60}$

Important Considerations

Q1: Why does RSA work correctly? Fermat's Little Theorem!

Q2: Can RSA be implemented efficiently? Yes, repeated squaring!

RSA on an Example.

Public Key: $(77,7)$
Message Choices: $\{0,\ldots,76\}$.

Message: 2

$E(2) = 2^e = 2^7 \equiv 128 \pmod{77} = 51 \pmod{77}$
$D(51) = 51^{43} \pmod{77}$
uh oh!

Obvious way: 43 multiplications! Expensive!

In general, $O(N)$ multiplications in the *value* of the exponent N!
That's not great.

Repeated Squaring to the rescue.

$51^{43} = 51^{32+8+2+1} = 51^{32} \cdot 51^8 \cdot 51^2 \cdot 51^1 \pmod{77}$.
Note: No 51^4, 51^{16}, ... 0s vs 1s in the binary representation of 43.
How many multiplications do we have?
Need to compute $51^{32} \ldots 51^1$.?
$51^1 \equiv 51 \pmod{77}$
$51^2 = (51)*(51) = 2601 \equiv 60 \pmod{77}$
$51^4 = (51^2)*(51^2) = 60*60 = 3600 \equiv 58 \pmod{77}$
$51^8 = (51^4)*(51^4) = 58*58 = 3364 \equiv 53 \pmod{77}$
$51^{16} = (51^8)*(51^8) = 53*53 = 2809 \equiv 37 \pmod{77}$
$51^{32} = (51^{16})*(51^{16}) = 37*37 = 1369 \equiv 60 \pmod{77}$

5 more multiplications.

$51^{32} \cdot 51^8 \cdot 51^2 \cdot 51^1 = (60)*(53)*(60)*(51) \equiv 2 \pmod{77}$.

Decoding got the message back!

Repeated Squaring took 9 multiplications versus 43.

Repeated Squaring: x^y

Repeated squaring $O(\log y)$ multiplications versus y!!!

1. x^y: Compute $x^1, x^2, x^4, \ldots, x^{2^{\lfloor \log y \rfloor}}$.
2. Multiply together x^i where the $(\log(i))$th bit of y is 1.

Always decode correctly?

Fermat's Little Theorem: For prime p, and $a \not\equiv 0 \pmod{p}$,
$a^{p-1} \equiv 1 \pmod{p}$.

Proof: Consider $S = \{a \cdot 1, \ldots, a \cdot (p-1)\}$.

All different modulo p since a has an inverse modulo p. That is: S contains representative of each of $1, \ldots, p-1$ modulo p.

$$(a \cdot 1) \cdot (a \cdot 2) \cdots (a \cdot (p-1)) \equiv 1 \cdot 2 \cdots (p-1) \mod p,$$

Since multiplication is commutative.

$$a^{(p-1)}(1 \cdots (p-1)) \equiv (1 \cdots (p-1)) \mod p.$$

Each of $2, \ldots (p-1)$ has an inverse modulo p, solve to get...

$$a^{(p-1)} \equiv 1 \mod p.$$

$\square$

RSA and Fermat: mathematical connection

Thm: $m^{ed} = m \pmod{pq}$ if $ed = 1 \pmod{(p-1)(q-1)}$
Seems like magic!

Fermat's Little Theorem: For prime p, and $a \not\equiv 0 \pmod{p}$,

$a^{p-1} \equiv 1 \pmod{p}$.

$3^6 \pmod{7}$? 1.
$3^7 \pmod{7}$? 3.
$3^{19} \pmod{7}$? $3^{3*6+1} \pmod{7}$? $(3^{3*6} * 3) \pmod{7}$? 3.

Corollary: $a^{k(p-1)+1} = a \pmod{p}$

Get a back when exponent is $1 \pmod{p-1}$.
A little like RSA:
$a^{ed} \pmod{(p-1)(q-1)}$ is a
when exponent is $1 \pmod{(p-1)(q-1)}$.

Proof of Corollary. If $a = 0$, $a^{k(p-1)+1} = 0 \pmod{p}$.
Otherwise $a^{1+k(p-1)} \equiv a^1 * (a^{p-1})^k \equiv a * (1)^k \equiv a \pmod{p}$ $\square$

Idea: Fermat removes the $k(p-1)$ from the exponent!

Correctness of RSA...

Lemma 1: For any prime p and any a, b,
$a^{1+b(p-1)} \equiv a \pmod{p}$

Lemma 2: For any two different primes p, q and any x, k,
$x^{1+k(p-1)(q-1)} \equiv x \pmod{pq}$

Let $a = x$, $b = k(p-1)$ and apply Lemma 1 with modulus q.

$x^{1+k(p-1)(q-1)} \equiv x \pmod{q}$

$x^{1+k(q-1)(p-1)} - x \equiv 0 \mod (q) \implies$ multiple of q.

Let $a = x$, $b = k(q-1)$ and apply Lemma 1 with modulus p.

$x^{1+k(p-1)(q-1)} \equiv x \pmod{p}$

$x^{1+k(q-1)(p-1)} - x \equiv 0 \mod (p) \implies$ multiple of p.

$x^{1+k(q-1)(p-1)} - x$ is multiple of p and q.

$x^{1+k(q-1)(p-1)} - x \equiv 0 \mod (pq) \implies x^{1+k(q-1)(p-1)} = x \mod pq$. $\square$

RSA decodes correctly..

Lemma 2: For any two different primes p, q and any x, k,
$x^{1+k(p-1)(q-1)} \equiv x \pmod{pq}$

Theorem: RSA correctly decodes!
Recall

$$D(E(x)) = (x^e)^d = x^{ed} \equiv x \pmod{pq},$$

where $ed \equiv 1 \mod (p-1)(q-1) \implies ed = 1 + k(p-1)(q-1)$

$$x^{ed} \equiv x^{k(p-1)(q-1)+1} \equiv x \pmod{pq}.$$

$\square$

Key Generation...

1. Find large (100 digit) primes p and q?

   Prime Number Theorem: $\pi(N)$ denotes the number of primes less than or equal to N. For all $N \geq 17$

   $$\pi(N) \geq N / \ln N.$$

   Choosing randomly gives approximately $1/(\ln N)$ chance of number being a prime. (How do you tell if it is prime? ... cs170..Miller-Rabin test.. Primes in P).
2. Choose e with $\gcd(e, (p-1)(q-1)) = 1$.
   Use gcd algorithm to test.
3. Find inverse d of e modulo $(p-1)(q-1)$.
   Use extended gcd algorithm.

All steps are polynomial in $O(\log N)$, the number of bits.

Security of RSA.

Security?

1. Alice knows p and q (and d, and other numbers).
2. Bob only knows, $N(= pq)$, and e.
   Does not know, for example, d or factorization of N.
3. Breaking this scheme $\Longrightarrow$ factoring N.
   Don't know how to factor N efficiently on regular computers.

Much more to it in practice!

If Bobs sends a message (Credit Card Number) to Alice,

Eve sees it. (The encrypted CC number.)

Eve can send same credit card number again!!
"Replay attack"

The protocols are built on RSA but more complicated;
For example, several rounds of challenge/response.

One trick:
Bob encodes credit card number, c,
concatenated with random k-bit number r ("nonce").

Never sends just c.

Again, more work to do to get entire system.

Further study: CS161 and CS171.