**CS70: New Discussion Format**

Small group:
- Three modes of working.
  - (A) Individual working.
  - (B) Pairs working together.
  - (C) Pairs: one works one forces talking.

Supported by course staff and course volunteers.

Why? It works better for learning.

Evidence:
- (1) Experience. (years and years, faculty agree.)
- (2) Literature.

Students hate it.

Students happy (in the moment): negatively correlated to learning.

Do you remember the first lecture?

**Simple Chinese Remainder Theorem.**

My love is won. Zero and One. Nothing and nothing done.

Find \( x = a \pmod{m} \) and \( x = b \pmod{n} \) where \( \gcd(m,n)=1 \).

**CRT Thm:** There is a unique solution \( x \pmod{mn} \).

**Proof (uniqueness):**

If not, two solutions, \( x \) and \( y \):

\[
\begin{align*}
(x - y) &= 0 \pmod{m} \quad \text{(since } \gcd(m,n) = 1) \\
(x - y) &= 0 \pmod{n}
\end{align*}
\]

\[\implies (x - y) \text{ is multiple of } m \text{ and } n\]

\[\implies mn | (x - y)\]

Thus, only one solution modulo \( mn \).

**Simple Chinese Remainder Theorem.**

**Do you remember the first lecture?**

Veritassium on Khan

![Graph showing recall rates](image)

**CS70: Lecture 9. Outline.**

1. Public Key Cryptography
   - 2. RSA system
     - 2.1 Efficiency: Repeated Squaring.
     - 2.2 Correctness: Fermat's Theorem.
     - 2.3 Construction.
   - 3. Warnings.

**Isomorphisms.**

**Bijection:**

\( f(x) = ax \pmod{m} \) if \( \gcd(a,m) = 1 \).

**Simplified Chinese Remainder Theorem:**

If \( \gcd(n,m) = 1 \), there is unique \( x \pmod{mn} \) where

\[
\begin{align*}
x &= a \pmod{m} \\
x &= b \pmod{n}
\end{align*}
\]

Bijection between \((a \pmod{n}), b \pmod{m}) and \( x \pmod{mn} \).

Consider \( m = 5, n = 9 \), then if \((a,b) = (3,7)\) then \( x = 43 \pmod{45} \).

Consider \((a',b') = (2,4)\), then \( x = 22 \pmod{45} \).

Now consider: \((a,b) + (a',b') = (0,2)\).

What is \( x \) where \( x = 0 \pmod{5} \) and \( x = 2 \pmod{9} \)?

Try \( 43 + 22 = 65 \equiv 20 \pmod{45} \).

Is it \( 0 \pmod{5} \)? Yes! Is it \( 2 \pmod{9} \)? Yes!

Isomorphism:

the actions under \( \pmod{5}, \pmod{9} \) correspond to actions in \( \pmod{45} \).
Poll

\[
\begin{align*}
  x &= 5 \mod 7 \text{ and } x = 5 \mod 6 \\
  y &= 4 \mod 7 \text{ and } y = 3 \mod 6 \\
\end{align*}
\]

\textbf{What's true?}

(A) \( x + y = 2 \mod 7 \)  \\
(B) \( x + y = 2 \mod 6 \)  \\
(C) \( xy = 3 \mod 6 \)  \\
(D) \( xy = 6 \mod 7 \)  \\
(E) \( x = 5 \mod 42 \)  \\
(F) \( y = 39 \mod 42 \)

All true.

Poll

\[
\begin{align*}
  x &= 5 \mod 7 \text{ and } x = 5 \mod 6 \\
  y &= 4 \mod 7 \text{ and } y = 3 \mod 6 \\
\end{align*}
\]

\textbf{What's true?}

(A) \( x + y = 2 \mod 7 \)  \\
(B) \( x + y = 2 \mod 6 \)  \\
(C) \( xy = 3 \mod 6 \)  \\
(D) \( xy = 6 \mod 7 \)  \\
(E) \( x = 5 \mod 42 \)  \\
(F) \( y = 39 \mod 42 \)

All true.

Xor

\textbf{Computer Science:}

\[
\begin{align*}
  1 &- \text{ True} \\
  0 &- \text{ False} \\
  1 \lor 1 &= 1 \\
  1 \lor 0 &= 1 \\
  0 \lor 1 &= 1 \\
  0 \lor 0 &= 0 \\
  A \oplus B &- \text{Exclusive or.} \\
  1 \oplus 1 &= 0 \\
  1 \oplus 0 &= 1 \\
  0 \oplus 1 &= 1 \\
  0 \oplus 0 &= 0 \\
\end{align*}
\]

\textbf{Property:} \( A \oplus B \oplus B = A \).

\textbf{By cases:} \( 1 \oplus 1 \oplus 1 = 1 \).

\textbf{Disadvantages:}

\textbf{Shared secret!}

\textbf{Uses up one time pad.} or less and less secure.

Cryptography ...

\textbf{Bob Alice Eve}

\textbf{Secret} \( s \)

\textbf{Message} \( m \)

\( m = D(E(m,K),s) \)

Public key cryptography.

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

\textbf{Private:} \( k \)

\textbf{Public:} \( K \)

\textbf{Message} \( m \)

\textbf{Alice}

\( E(m,K) \)

\textbf{Bob}

\( E(m,K) \)

\textbf{Eve}

\textbf{What is a piece of RSA?}

Bob has a key \((N,e,d)\). Alice is good, Eve is evil.

(A) Eve knows \( e \) and \( N \).

(B) Alice knows \( e \) and \( N \).

(C) \( ed \equiv 1 \pmod{N-1} \).

(D) Bob forgets \( p \) and \( q \) but can still decode?

(E) Bob knows \( d \).

(F) \( ed \equiv 1 \pmod{(p-1)(q-1)} \) if \( N = pq \).

\textbf{(A), (B), (D), (E), (F)}
Repeated Squaring: $x^y$

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

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

Example: $43 = 101011$ in binary,

$x_{\text{as 32-bit}} = 1 \times 2^5 + 1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 = 11011$.

Modular Exponentiation: $x^y \mod N$. All $n$-bit numbers. Repeated Squaring:

- $O(n^2)$ multiplications.
- $O(n^2)$ time per multiplication.
- $O(n^2)$ time.

Conclusion: $x^y \mod N$ takes $O(n^2)$ time.

Recursive.

$x^y$.  

xseven, $x = 2k$, $x^y = x^{2k} = (x^2)^k$.

power $(x,y) = power (x^2, y/2)$.

xisodd, $x = 2k+1$, $x^y = x^{2k} = (x^2)^{k-1}$.

power $(x,y) = x \cdot power (x^2, y/2)$.

Base case: $x^0 = 1$.

RSA is pretty fast.

Modular Exponentiation: $x^y \mod N$. All $n$-bit numbers.

$O(n^2)$ time.

Remember RSA encoding/decoding!

$E(m: (N,e)) = m^e \mod N$.

$D(m: (N,d)) = m^d \mod N$.

For 512 bits, a few hundred million operations.

Easy, peasy.

Iterative Extended GCD.

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

$N = 77$.

$(p-1)(q-1) = 60$

Choose $e = 7$, since $\text{gcd}(7,60) = 1$.

$\text{egcd}(7,60)$.

7(0) + 60(1) = 60

7(1) + 60(0) = 7

7(-8) + 60(1) = 4

7(9) + 60(-1) = 3

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

Confirm: −119 + 120 = 1

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

Encryption/Decryption Techniques.

Public Key: (77, 7)

Message Choices: {0, ..., 76}.

Message: 2!

$E(2) = 2^7 = 128 = 51 \mod 77$

$D(51) = 51^{43} \mod 77$

uh oh!

Obvious way: 43 multiplications. Ouch.

In general, $O(N)$ or $O(2^n)$ multiplications!

Repeated squaring.

$51^{43} = 51^{64 \cdot 8 + 2 + 1} = 51^{53} \cdot 51^{2} \cdot 51^{1} \mod 77$.

Need to compute $51^{53} \cdot 51^{2} \cdot 51^{1}$.?

$51^{1} = 51 \mod 77$

$51^{2} = (51) \cdot (51) = 2601 = 60 \mod 77$

$51^{3} = (51)^{2} \cdot (51) = 60 + 60 - 3600 = 58 \mod 77$

$51^{4} = (51)^{3} \cdot (51) = 58 - 58 = 3364 = 53 \mod 77$

5 more multiplications.

$51^{53} = 51^{16} \cdot 51^{16} \cdot 51^{16} \cdot 51^{3} \cdot 51 = 37 \cdot 37 = 1369 = 60 \mod 77$

Decoding got the message back!

Repeated Squaring took 8 multiplications versus 42.
Always decode correctly?

\[
E(m, (N, e)) = m^e \pmod{N},
D(m, (N, d)) = m^{d} \pmod{N}.
\]
\[
N = pq \text{ and } d = e^{-1} \pmod{(p-1)(q-1)}.
\]
Want: \((m^d)^d = m^{md} = m \pmod{(mod N)}.

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

**Proof:** Consider \(S = \{a \cdot 1 \ldots \cdot a \pmod{p} \} \) modulo \(p\).
Each of \(a, \ldots, a^{p-1}\) has an inverse modulo \(p\), solve to get...
\[a(a^{p-1})^{-1} \equiv 1 \pmod{p}.
\]

Correct decoding...

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

**Proof:** Consider \(S = \{a \cdot 1 \ldots \cdot a \pmod{p} \} \) modulo \(p\).
All different modulo \(p\) since \(a\) has an inverse modulo \(p\).
\(S\) contains representative of \(\{1, \ldots, p-1\} \pmod{p}\).

\[(a \cdot 1 \cdot \ldots \cdot a \pmod{p}) \equiv 1 \cdot 2 \cdot \ldots \cdot (p-1) \pmod{p}.
\]
Since multiplication is commutative...
\[a^{(p-1)}(1 \cdot \ldots \cdot (p-1)) \equiv (1 \cdot \ldots \cdot (p-1)) \pmod{p}.
\]
Each of \(2 \ldots (p-1)\) has an inverse modulo \(p\), solve to get...
\[a^{(p-1)} \equiv 1 \pmod{p}.
\]

...Decoding correctness...

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

**Proof:**\(a \equiv 0 \pmod{p}\), of course.
Otherwise
\[a^{1+k(p-1)} = a^1 \equiv (a^{p-1})^k = a \pmod{p}.
\]

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

**Proof:**\(a \equiv 0 \pmod{p}\), of course.
\[a^{1+k(p-1)} = x \pmod{q}\]
\[x^{1+k(q-1)} \equiv x \pmod{q} \iff x^{1+k(q-1)}(p-1) \equiv x \pmod{pq}\]
From CRT: \(y = x \pmod{p}\) and \(y = x \pmod{q} \iff y = x.

---

### Poll

Mark what is true.

- (A) \(2^7 \equiv 1 \pmod{7}\)
- (B) \(2^8 \equiv 1 \pmod{7}\)
- (C) \(2^1, 2^2, 2^3, 2^5, 2^6, 2^7\) are distinct \(\pmod{7}\).
- (D) \(2^1, 2^2, 2^3, 2^5, 2^6\) are distinct \(\pmod{7}\)
- (E) \(2^8 \equiv 2 \pmod{7}\)
- (F) \(2^9 \equiv 1 \pmod{7}\)
- (B), (F)
RSA decodes correctly.

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

Theorem: RSA correctly decodes!
Recall
\[ D(E(x)) = (x^e)^d = x^{ed} = x \pmod{pq}, \]
where $ed ≡ 1 \pmod{(p−1)(q−1)} \implies ed = 1 + k(p−1)(q−1)$
\[ x^{ed} = x^{k(p−1)(q−1)+1} = x \pmod{pq}. \]

Security of RSA.

Security?
1. Alice knows $p$ and $q$.
2. Bob only knows, $N(= pq)$, and $e$.
   Does not know, for example, $d$ or factorization of $N$.
3. I don’t know how to break this scheme without factoring $N$.
No one I know or have heard of admits to knowing how to factor $N$.
Breaking in general sense $\implies$ factoring algorithm.

Construction of keys... ..

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

Prime Number Theorem: $\pi(N)$ number of primes less than $N$
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 $P$).
For $1024$ bit number, $1$ in $710$ is prime.

Security: Eve can't forge unless she “breaks” RSA scheme.

Much more to it.....

If Bobs sends a message (Credit Card Number) to Alice,
Eve sees it.
Eve can send credit card again!!
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 $r$.
Never sends just $c$.
Again, more work to do to get entire system.
CS161...

Signatures using RSA.

Verisign: $k_v, K_v$

Certificate Authority: Verisign, GoDaddy, DigiNotar,...
Verisign's key: $K_v = (N, e)$ and $k_v = d \equiv N = pq$.
Browser "knows" Verisign's public key: $K_v$.
Amazon Certificate: $C = \text{"I am Amazon. My public Key is } K_A,$
Verisign signature of $C$: $S_v(C)$:
$D(C, k_v) = C^d \pmod{N}$
Browser receives: $[C, S_v(C)]$
Checks $E(y, K_y) = C$?
$E(S_v(C), K_y) = (S_v(C))^y = (C^d)^y = C^{de} = C \pmod{N}$
Valid signature of Amazon certificate $C$!
Security: Eve can't forge unless she "breaks" RSA scheme.

Public Key Cryptography:
$D(E(m, K), k) = (m^e)^d \equiv m \pmod{N}$
Signature scheme:
$E(D(C, k), K) = (C^d)^e \equiv C \pmod{N}$

RSA
Poll

Signature authority has public key \((N,e)\).

(A) Given message/signature \((x,y)\) : check \(yd = x \pmod{N}\)
(B) Given message/signature \((x,y)\) : check \(ye = x \pmod{N}\)
(C) Signature of message \(x\) is \(xe \pmod{N}\)
(D) Signature of message \(x\) is \(xd \pmod{N}\)

Other Eve.

Get CA to certify fake certificates: Microsoft Corporation.


... and August 28, 2011 announcement.

DigiNotar Certificate issued for Microsoft!!!

How does Microsoft get a CA to issue certificate to them ... and only them?

Summary.

Public-Key Encryption.

RSA Scheme:
\(N = pq\) and \(d = e^{-1} \pmod{(p-1)(q-1)}\).

\(E(x) = x^e \pmod{N}\).

\(D(y) = y^d \pmod{N}\).

Repeated Squaring \(\Rightarrow\) efficiency.

Fermat’s Theorem \(\Rightarrow\) correctness.

Good for Encryption and Signature Schemes.