Primality Testing and Attacks on RSA

Murat Kantarcioglu

Based on Prof. Ninghui Li's Slides

Review of RSA

Public key: (e, n)

Secret key: d

where n=pq and $ed \equiv 1 \pmod{\Phi(n)}$

Encrypting M: $M^e \bmod n$

Decrypting C: $C^d \bmod n$

2

Lecture Outline

- Number of prime numbers
- Cyclic groups
- Quadratic residues
- Primality testing
- Factorization
- Attacks on RSA

3

Number of Prime Numbers

Theorem

The number of prime numbers is infinite.

Proof: For the sake of contradiction, assume that the number of prime numbers is finite. Let p_1, p_2, … p_k be all primes. Let $n = p_1 p_2 \dots p_k+1$, then n must be composite.

Then there exists a prime p s.t. $p \mid n$ (fundamental theorem of arithmetic), and p cannot be any of the p_1, p_2, … p_k. (Why?)

Therefore, p_1, … p_k were not all the prime numbers.

4

Distribution of Prime Numbers

Theorem (Gaps between primes)

For every positive integer n, there are n or more consecutive composite numbers.

Proof Idea:

The consective numbers

$(n+1)! + 2, (n+1)! + 3, \ldots., (n+1)! + n+1$

are composite.

(Why?)

5

Distribution of Prime Numbers

Definition

Given real number x, let $\pi(x)$ be the number of prime numbers $\leq x$.

Theorem (prime numbers theorem)

$$\lim_{x \to \infty} \frac{\pi(x)}{x / \ln x} = 1$$

For a very large number x, the number of prime numbers smaller than x is close to x/ln x.

6

Generating large prime numbers

- Randomly generate a large odd number and then test whether it is prime.
- How many random integers need to be tested before finding a prime?
  - the number of prime numbers ≤ p is about p/ ln p
  - roughly every ln p integers has a prime
    - for a 512 bit p, ln p = 355. on average, need to test about 177=355/2 odd numbers
- Need to solve the Primality testing problem
  - the decision problem to decide whether a number is a prime

7

{Complexity}

- **Complexity theory**: mathematical discipline that classifies problems based on the difficulty to solve them.
- **P-class** (polynomial-time): number of steps needed to solve a problem is bounded by some power of the problem's size.
- **NP-class** (nondeterministic polynomial-time): it permits a nondeterministic solution and the number of steps to verify the solution is bounded by some power of the problem's size.

8

Testing for Primality

Theorem

Composite numbers have a divisor less than equal to their square root.

Proof idea:

n composite, so $n = ab$, $0 < a \le b < n$, then $a \le \mathrm{sqrt}(n)$, otherwise we obtain $ab > n$ (contradiction).

Algorithm 1

```
for (i=2, i < sqrt(n) + 1); i++) {
    If i a divisor of n {
        n is composite
    }
}
n is prime
```

Running time is O(sqrt(n)), which is exponential in the size of the binary representation of n

9

More Efficient Algorithms for Primality Testing

- Primality testing is easier than prime factorization, and is in P-class.

How can we tell if a number is prime or not without factoring the number?

- The most efficient algorithms are randomized.
  - Solovay-Strassen
  - Rabin-Miler

10

Groups

- A group donated by (G,*) is a set of non-empty elements with binary operation *
- Closure: $a*b \in G$ for all $a,b \in G$
- Associativity: $(a*b)*c=a*(b*c)$ for all $a,b,c \in G$
- Identity Element: There exists unique e s.t. $e*a=a*e=a$ for all $a \in G$
- Inverse: Every element $a \in G$ has an inverse b s.t. $a*b=b*a=e$
- Commutativity: $a*b=b*a$ for all $a,b \in G$

More Number Theory First

- Definition: Given a group $(G,\bullet)$,
  - the order of G is $|G|$
  - the order of an element a in G is the smallest positive integer such that $a^m=1$
  - $\{a,a^2,\ldots,a^m\}$ is a subgroup of G
    - (why?)
- Definition: a group $(G,\bullet)$ is a cyclic group if there exists $g \in G$ such that $G=\{g, g\bullet g, g^3, \ldots, g^{|G|}\}$
  - g is known as a generator
  - the order of g is $|G|$
    - (why?)

12

Z_p* is a Cyclic Group

- Fact: Given a prime p, Z_p* is a cyclic group.
  - we won't prove it here.
- There exists $g \in Z_p^*$ s.t. $\{g^j \mid 1 \le j \le p-1\} = Z_p^*$
  - g is a generator of Z_p*,
  - g is also known as the primitive element modulo p
  - what is the order of g
- For example, 2 is a generator for Z_{11}*
  - $\{2^j \mid 1 \le j \le p-1\} = \{2,4,8,5,10,9,7,3,6,1\}$
  - what is the order of $4=2^2$? what is the order of $8=2^3$?
- Let g be a generator of Z_p*, and let $a=g^j$
  - the order of a is (p-1)/gcd(p-1,j)
  - what are the primitive elements in Z_{11}*?

13

Testing Primitive Elements Modulo p

- The number of primitive elements modulo p is $\phi(p-1)$.

Theorem: Let p be a prime, $a \in Z_p^*$ is a primitive element modulo p iff. $a^{(p-1)/q} \neq 1 \pmod p$ for all primes q such that q|(p-1).

Proof. The only if direction is straightforward.

For the if direction. If a is not primitive, it has order d<(p-1). Then d is a divisor of (p-1). Let q be a prime factor of (p-1)/d, i.e., (p-1)/d=cq. Then (p-1)/q=cd. Then $a^{(p-1)/q} = 1 \pmod p$.

14

Quadratic Residues Modulo A Prime

Definition

- a is a quadratic residue modulo p if $\exists\, b \in Z_p^*$ such that $b^2 \equiv a \bmod p$,
- otherwise when $a \neq 0$, a is a quadratic nonresidue
- Q_p is the set of all quadratic residues
- $\overline{Q}_p$ is the set of all quadratic nonresidues
- If p is prime there are (p-1)/2 quadratic residues in Z_p^*, $|Q_p| = (p-1)/2$
  - let g be generator of Z_p^*, then $a=g^j$ is a quadratic residue iff. j is even.

15

How Many Square Roots Does an Element in Q_p has

- A element a in Q_p has exactly two square roots
  - a has at least two square roots
    - if $b^2 \equiv a \bmod p$, then $(p-b)^2 \equiv a \bmod p$
  - a has at most two square roots in Z_p^*
    - if $b^2 \equiv a \bmod p$ and $c^2 \equiv a \bmod p$, then $b^2 - c^2 \equiv 0 \bmod p$
    - then p | (b+c)(b-c), either b=c, or b+c=p

16

Legendre Symbol

- Let p be an odd prime and a an integer. The Legendre symbol is defined

$$\left(\frac{a}{p}\right) = \begin{cases} 0, \text{ if } p \mid a \\ 1, \text{ if } a \in Q_p \\ -1, \text{ if } a \in \overline{Q}_p \end{cases}$$

17

Euler's Criterion

Theorem: If $a^{(p-1)/2} \equiv 1$ mod p, then a is a quadratic residue (if $\equiv$ -1 then a is a quadratic nonresidue)

I.e., the Legendre symbol $\left(\frac{a}{p}\right) = a^{(p-1)/2}$ mod p

Proof. If $a = y^2$, then $a^{(p-1)/2} = y^{(p-1)} = 1$ (mod p)

If $a^{(p-1)/2}=1$, let $a = g^j$, where g is a generator of the group Z_p^*. Then $g^{j(p-1)/2} = 1$ (mod p). Since g is a generator, (p-1) | j (p-1)/2, thus j must be even. Therefore, $a=g^j$ is QR.

18

Jacobi Symbol

- let n ≥ 3 be odd with prime factorization

$$n = p_1^{e_1} p_2^{e_2} \dots p_k^{e_k}$$

- the Jacobi symbol is defined to be

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p_1}\right)^{e_1} \left(\frac{a}{p_2}\right)^{e_2} \dots \left(\frac{a}{p_k}\right)^{e_k}$$

- the Jacobi symbol can be computed without factoring n (see the textbook for details)

19

Euler Pseudo-prime

- For any prime p, the Legendre symbol $\left(\frac{a}{p}\right)$ = $a^{(p-1)/2}$ mod p
- For a composite n, if the Jacobi symbol $\left(\frac{a}{n}\right)$ = $a^{(n-1)/2}$ mod n then n is called an Euler pseudo-prime to the base a,
  - i.e., a is a "pseudo" evidence that n is prime
- For any composite n, the number of "pseudo" evidences that n is prime for at most half of the integers in Z_n^*

20

Randomized Algorithms

- A yes-biased Monte Carlo algorithm is a randomized algorithm for a decision problem in which a “yes” answer is (always correct), but a “no” answer may be incorrect
  - error probability for an instance is the probability that instance is answered incorrectly
  - error probability for the algorithm is the max among all instance error probabilities
- A no-biased Monte Carlo algorithm is defined similarly
- A Las Vegas algorithm may not give an answer, but any answer it gives is correct

The Solovay-Strassen Algorithm

Solovay-Strassen(n)

choose a random integer a s.t. $1 \le a \le n-1$

$x \leftarrow \left(\frac{a}{n}\right)$

if x=0 then return (“n is composite”) // gcd(x,n)≠1

$y \leftarrow a^{(n-1)/2} \bmod n$

if (x=y) then return (“n is prime”)

// either n is a prime, or a pseudo-prime

else return (“n is composite”)

// violates Euler’s criterion

If n is composite, it passes the test with at most ½ prob. Use multiple tests before accepting n as prime.

Rabin-Miller Test

- Another efficient probabilistic algorithm for determining if a given number n is prime.
  - Write n-1 as $2^k m$, with m odd.
  - Choose a random integer a, $1 \le a \le n-1$.
  - $b \leftarrow a^m \bmod n$
  - if b=1 then return "n is prime"
  - compute $b, b^2, b^4, \ldots, b^{2^{(k-1)}}$, if we find -1, return "n is prime"
  - return "n is composite"
- A composite number pass the test with ¼ prob.
- When t tests are used with independent a, a composite passes with $(1/4)^t$ prob.
- The test is fast, used very often in practice.

23

Why Rabin-Miller Test Work

Claim: If the algorithm returns "n is composite", then n is not a prime.

Proof: if we choose a and returns composite on n, then

- $a^m \neq 1$, $a^m \neq -1$, $a^{2m} \neq -1$, $a^{4m} \neq -1$, …, $a^{2^{k-1}m} \neq -1$ (mod n)
- suppose, for the sake of contradiction, that n is prime,
- then $a^{n-1} = a^{2^k m} = 1$ (mod n)
- then there are two square roots modulo n, 1 and -1
- then $a^{2^{k-1}m} = a^{2^{k-2}m} = a^{2m} = a^m = 1$ (contradiction!)
- so if n is prime, the algorithm will not return "composite"

24

Quadratic Residues Modulo a Composite

Definition: a is a quadratic residue modulo n ($a \in Q_n$) if $\exists$ b $\in Z_n^*$ such that $b^2 \equiv a \bmod n$, otherwise when $a \neq 0$, a is a quadratic nonresidue

Fact: $a \in Q_n^*$, where n=pq, iff. $a \in Q_p$ and $a \in Q_q$

- If $b^2 \equiv a \bmod n$, then $b^2 \equiv a \bmod p$ and $b^2 \equiv a \bmod q$
- If $b^2 \equiv a \bmod p$ and $c^2 \equiv a \bmod q$, then the solutions to

  $x \equiv b \bmod p$ and $x \equiv c \bmod q$

  $x \equiv b \bmod p$ and $x \equiv -c \bmod q$

  $x \equiv -b \bmod p$ and $x \equiv c \bmod q$

  $x \equiv -b \bmod p$ and $x \equiv -c \bmod q$

  satisfies $x^2 \equiv a \bmod n$

25

Quadratic Residues Modulo a Composite

- $|Q_n| = |Q_p| \bullet |Q_q| = (p-1)(q-1)/4$
- $\overline{Q}_n = 3(p-1)(q-1)/4$
- Jacobi symbol does not tell whether a number a is a QR

$$\left(\frac{a}{n}\right) = \left(\frac{a}{p}\right)\left(\frac{a}{q}\right)$$

- when it is -1, then either $a \in Q_p \wedge a \notin Q_q$ or $a \notin Q_p \wedge a \in Q_q$
- when it is 1, then either $a \in Q_p \wedge a \in Q_q$ or $a \notin Q_p \wedge a \notin Q_q$
- it is widely believed that determining QR modulo n given
- that $\left(\frac{a}{n}\right) = 1$ is equivalent to factoring n, no proof is known
  - without factoring, one can guess correctly with prob. ½

26

Summary of Number Theory Results Covered

- Z_p^* is a cyclic group
  - has generators
- QR and QNR in Z_p^* can be easily determined by computing the Legendre symbol
- Jacobi symbol (generalizes Legendre symbol to composites)
  - can be computed without factoring n
  - Jacobi symbol does not determine QR in Z_n^*
  - QR in Z_n^* is hard
- Primality Testing
  - Solovay-Strassen
  - Rabin-Miller

27

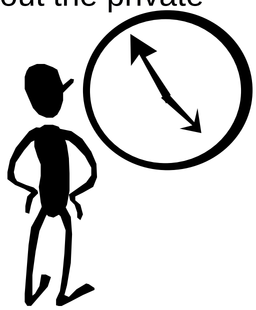
Brief Overview of Attacks on RSA

- Goals:
  - recover secret key d
    - Brute force key search
      - infeasible
    - Timing attacks
    - Mathematical attacks
  - decrypt one message
  - learn information from the cipher texts

28

Timing Attacks

- *Timing Attacks on Implementations of Diffie-Hellman, RSA, DSS, and Other Systems (1996), Paul C. Kocher*
- By measuring the time required to perform decryption (exponentiation with the private key as exponent), an attacker can figure out the private key
- Possible countermeasures:
  - use constant exponentiation time
  - add random delays
  - blind values used in calculations



29

Timing Attacks (cont.)

- Is it possible in practice? YES.

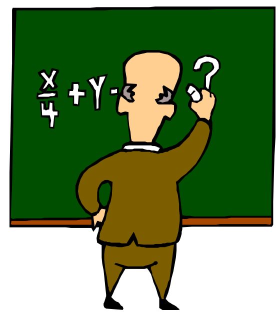
OpenSSL Security Advisory [17 March 2003]
Timing-based attacks on RSA keys
================================
OpenSSL v0.9.7a and 0.9.6i vulnerability

Researchers have discovered a timing attack on RSA keys, to which OpenSSL is generally vulnerable, unless RSA blinding has been turned on.

30

Math-Based Key Recovery Attacks

- Three possible approaches:
  1. Factor n = pq
  2. Determine $\Phi(n)$
  3. Find the private key d directly



- All the above are equivalent to factoring n
  - 1 implies 2
  - 2 implies 3
  - needs to show that 3 implies 1

31

$\Phi(n)$ implies factorization

- Knowing both n and $\Phi(n)$, one knows

$$n = pq$$
$$\Phi(n) = (p-1)(q-1) = pq - p - q + 1$$
$$= n - p - n/p + 1$$
$$p\Phi(n) = np - p^2 - n + p$$
$$p^2 - np + \Phi(n)p - p + n = 0$$
$$p^2 - (n - \Phi(n) + 1)\, p + n = 0$$

- There are two solutions of p in the above equation.
- Both p and q are solutions.

32

Factoring Large Numbers

- Three most effective algorithms are
  - quadratic sieve
  - elliptic curve factoring algorithm
  - number field sieve
- One idea many factoring algorithms use:
  - Suppose one find $x^2 \equiv y^2 \pmod{n}$ such that $x \neq y \pmod{n}$ and $x \neq -y \pmod{n}$. Then $n \mid (x-y)(x+y)$. Neither $(x-y)$ or $(x+y)$ is divisible by n; thus, gcd(x-y,n) has a non-trivial factor of n

33

Time complexity of factoring

- quadratic sieve:
  - $O(e^{(1+o(1))\mathrm{sqrt}(\ln n \ln \ln n)})$ for n around 2^{1024}, $O(e^{68})$
- elliptic curve factoring algorithm
  - $O(e^{(1+o(1))\mathrm{sqrt}(2 \ln p \ln \ln p)})$, where p is the smallest prime factor
  - for n=pq and p,q around 2^{512}, for n around 2^{1024} $O(e^{65})$
- number field sieve
  - $O(e^{(1.92+o(1)) (\ln n)^{1/3} (\ln \ln n)^{2/3}})$, for n around 2^{1024} $O(e^{60})$
- Multiple 512-bit moduli have been factored
- Extrapolating trends of factoring suggests that
  - 768-bit moduli will be factored by 2010
  - 1024-bit moduli will be factored by 2018

34

Factoring when knowing e and d

- Fact: if n=pq, then $x^2 \equiv 1 \pmod n$ has four solutions that are <n.
  - $x^2 \equiv 1 \pmod n$ if and only if
    both $x^2 \equiv 1 \pmod p$ and $x^2 \equiv 1 \pmod q$
  - Two trivial solutions: 1 and n-1
    - 1 is solution to $x \equiv 1 \pmod p$ and $x \equiv 1 \pmod q$
    - n-1 is solution to $x \equiv -1 \pmod p$ and $x \equiv -1 \pmod q$
  - Two other solutions
    - solution to $x \equiv 1 \pmod p$ and $x \equiv -1 \pmod q$
    - solution to $x \equiv -1 \pmod p$ and $x \equiv 1 \pmod q$
  - E.g., $n=3\times5=15$, then $x^2 \equiv 1 \pmod{15}$ has the following solutions: 1, 4, 11, 14

35

Factoring when knowing e and d

- Knowing a nontrivial solution to $x^2 \equiv 1 \pmod n$
  - compute gcd(x+1,n) and gcd(x-1,n)
- E.g., 4 and 11 are solution to $x^2 \equiv 1 \pmod{15}$
  - gcd(4+1,15) = 5
  - gcd(4-1,15) = 3
  - gcd(11+1,15) = 3
  - gcd(11-1, 15) = 5

36

Factoring when knowing e and d

- Knowing ed such that ed ≡ 1 (mod $\Phi(n)$)

  write $ed - 1 = 2^s r$ (r odd)

  choose w at random such that 1<w<n-1

  if w not relative prime to n then return gcd(w,n)

  (if gcd(w,n)=1, what value is ($w^{2^s r}$ mod n)?)

  compute w^r, w^{2r}, w^{4r}, …, by successive squaring until find $w^{2^t r} \equiv 1 \pmod n$

Fails when $w^r \equiv 1 \pmod n$ or $w^{2^t r} \equiv -1 \pmod n$

Failure probability is less than ½ (Proof is complicated)

37

Summary of Key Recovery Math-based Attacks on RSA

- Three possible approaches:
  1. Factor n = pq
  2. Determine $\Phi(n)$
  3. Find the private key d directly
- All are equivalent
  - finding out d implies factoring n
  - if factoring is hard, so is finding out d
- Should never have different users share one common modulus
  - (why?)

38

Decryption attacks on RSA

- The RSA Problem: Given a positive integer n that is a product of two distinct large primes p and q, a positive integer e such that gcd(e, (p-1)(q-1))=1, and an integer c, find an integer m such that $m^e \equiv c \pmod{n}$
  - widely believed that the RSA problem is computationally equivalent to integer factorization; however, no proof is known

- The security of RSA encryption's scheme depends on the hardness of the RSA problem.

39