RSA and Primes

CS255
Chris Pollett
Apr. 19, 2006.

Outline

- Modular Exponentiation
- The RSA Public-key Cryptosystem

Powers of an Element

- Two useful theorems which are corollaries of earlier results:

Theorem. For any integer n > 1,

$a^{\phi(n)} \equiv 1 \pmod{n}$ for all a in $\mathbf{Z}_n^*$.

Theorem. If p is primes, then

$a^{p-1} \equiv 1 \pmod{p}$ for all a in $\mathbf{Z}_n^*$.

- The next theorem tells us the values of n for which $\mathbf{Z}_n^*$ is cyclic.

Theorem (#). The values of n> 1 for which $\mathbf{Z}_n^*$ is cyclic (that is, generated by one element) are 2, 4, p^e, and $2p^e$, for all primes $p > 2$ and all positive integers e.

More Powers of an Element

- g is a **primitive root** or **generator** of $\mathbf{Z}_n^*$ if $\langle g \rangle = \mathbf{Z}_n^*$.
- If g is a primitive root then the equation $g^x \equiv a \bmod n$ has a solution called the **discrete logarithm** or **index** of a mod n, which we write as $\mathrm{ind}_{n,\, g}(a)$.
- The next theorem concerns the discrete logarithm problem which is connected to factoring which is the basis of RSA.

Theorem (##). If g is a primitive root of $\mathbf{Z}_n^*$, then the equation $g^x \equiv g^y \pmod{n}$ holds if and only if the equation $x \equiv y \pmod{\phi(n)}$ holds.

Proof. Suppose $x \equiv y \pmod{\phi(n)}$ holds. Then $x = y + k\phi(n)$ for some k. So

$$g^x \equiv g^{y+k\phi(n)} \equiv g^y g^{k\phi(n)} \equiv g^y 1^k \equiv g^y \pmod{n}$$

Conversely, suppose $g^x \equiv g^y \pmod{n}$ holds. Since g is a generator, $|\langle g \rangle| = \phi(n)$. So we know g is periodic with period $\phi(n)$. Therefore, if $g^x \equiv g^y \pmod{n}$ we must have $x \equiv y \pmod{\phi(n)}$.

Square Roots

Theorem. If p is an odd prime, and $e \geq 1$, then the equation
$x^2 \equiv 1 \pmod{p^e}$
has only two solutions, $x = 1$ and $x = -1$.

Proof. Let $n = p^e$. Theorem (#) implies $\mathbf{Z}_n^*$ has a generator g. So the above equation can be rewritten as $(g^{ind(x)})^2 \equiv g^{ind(1)} \pmod{n}$. Note ind(1) = 0, so Theorem (##) implies this is equation is equivalent to $2 \cdot \text{ind}(x) \equiv 0 \pmod{\phi(n)}$, a modular linear equation we can solve. We know $\phi(n) = p^e(1 - 1/p) = (p-1)p^{e-1}$. If d is gcd(2, $\phi(n)$), then d=2 (as if p is odd divides p-1) and $d \mid 0$, we know this equation has 2 solutions, which we can compute using our algorithm or by inspection as 1 and -1.

- A number x is a **nontrivial square root of 1, modulo n**, if it is a square root but not equivalent to ±1 mod n. For example 6 mod 35.

Corollary. If there exists a nontrivial square root of 1, modulo n, then n is composite.

Modular Exponentiation

- We next give an algorithm based on repeated squaring to compute a^b mod n where a and b are nonnegative integers and n>0.
- We assume the number are written in binary and we use a subscript to denote the ith bit of a number. For example, b_i for the ith bit of b.

Modular-Exponentiation(a, b, n)

1. $d = 1$
2. for $i = k$ downto 0
3. $\quad d = (d \cdot d)$ mod n
4. $\quad$ if $b_i = 1$ then $\{d = (d \cdot a)$ mod $n\}$
5. return d

Public Key Cryptosystems

- We now apply what we've learned to **public key cryptography**.
- In public key cryptography, we have two participants Alice and Bob (i.e., A and B) who want to exchange messages securely.
- Each has a **public key** P_A, P_B which they let everyone know.
- They also each have a **private key** S_A, S_B which only they know.
- Each of these keys is a permutation in some space of strings and the public keys are inverses of the private keys. That is, $M = P_A(S_A(M)) = S_A(P_A(M))$. Here M is the message.
- If Alice want to send Bob a message M. She computes some hash function of M, h(M) and signs this with her private key to make $S_A(h(M))$. She concatenates this to M to make $<M, S_A(h(M))>$. Then she sends $P_B(<M, S_A(h(M))>)$ to Bob.
- To decode, Bob applies his private key to get $S_B(P_B(<M, S_A(h(M))>)) = <M, S_A(h(M))>$.
- To check this is from Alice, he applies her public key to the end $P_A(S_A(h(M))) = h(M)$ then he computes the hash of the message received and verifies it equal h(M).

RSA

- RSA (for the paper by Rivest, Shamir, and Adleman) is a particular public key cryptoscheme.
- It creates public keys and private keys as follows:
  1. Select two large prime numbers p and q such that $p \neq q$. (For instance, the primes might be 512 bits each.)
  2. Compute $n=pq$.
  3. Select a small odd integer e that is relatively prime to $\phi(n) = (p-1)(q-1)$.
  4. Compute the multiplicative inverse d of e mod $\phi(n)$.
  5. Publish the pair $P=(e, n)$ as the **RSA public key**.
  6. Keep secret the pair $S=(d, n)$ as the **RSA secret key**.
- To apply a key to a message $0 \leq M < n$, we compute either $P(M) = M^e \pmod{n}$ or $S(C) = C^d \pmod{n}$. Here C is suppose to mean ciphertext.

Correctness of RSA

Theorem. The RSA function P and S on the last slide define inverse transformations.

Proof. $P(S(M))= S(P(M))= M^{ed}$ (mod n). Since e and d are multiplicative inverses modulo $\phi(n) = (p-1)(q-1)$,

$$ed = 1+k(p-1)(q-1)$$

for some k. If $M \equiv 0$ (mod n), then $M^{ed} \equiv 0$ (mod n) so we are done. If M is not congruent to 0 (mod p), we have

$$\begin{aligned} M^{ed} &\equiv M(M^{p-1})^{k(q-1)} && \pmod{p} \\ &\equiv M(1)^{k(q-1)} && \pmod{p} \\ &\equiv M && \pmod{p} \end{aligned}$$

and a similar result holds mod q. By the chinese remainder theorem, this implies $M^{ed} \equiv M$ (mod n).

Testing for Primes.

- One key component of RSA is to use large primes chosen at random.
- It turns out that primes are not to rare since it is known that $\pi(n)$ = the number of primes less than n grows as $n/\log n$.
- However, we still need a way to check if a odd number is prime.
- One brute force approach is to try to divide each number up to sqrt(n). This is exponential in the number of bits of n.
- Recall if n is prime then $a^{n-1} \equiv 1 \pmod{n}$.
- A number is **pseudo-prime** for a, if it is composite but $a^{n-1} \equiv 1 \pmod{n}$.
- It turns out pseudo-primes are rare, so we could almost check for primality by checking this equation for different values for a.
- Unfortunately, there are even rarer numbers called **Carmichael numbers** which are composite, but such that this equation holds for all a. Rare since can show a Carmichael numbers needs to have at least 3 primes in it.
- For example, 561.

Miller Rabin Primality Testing

- Idea: (1) Try several randomly chosen values for *a*. (2) While computing each modular exponentiation we check, if we ever see a nontrivial square root of 1 mod n. If so, we know for sure the number is composite.
- The Non-Trivial Square root testing is done in the following routine:

Witness(a,n)

1. let $n-1 = 2^t u$, where $t \geq 1$ and u is odd
2. x_0 = Modular-Exponentiation(a,u, n)
3. for i = 1 to t
   a) do $x_i = (x_{i-1})^2 \bmod n$
      I. if $x_i = 1$ and $x_{i-1} \neq 1$ and $x_{i-1} \neq n-1$ then return true
4. if $x_t \neq 1$ then return true
5. return false

Miller Rabin (cont'd)

Miller-Rabin(n,s)

1. for $j = 1$ to s
   a) do a = Random(1, n-1)
      I. if Witness(a, n) then return Composite(a,n)
2. return prime.

Error Rate

- If Miller-Rabin says composite, we know the number is composite. If it says prime, there is some error rate given by the next theorem:

Theorem. If n is composite, the the number of witnesses to compositeness is at least (n-1)/2.

Proof. We show the number of nonwitnesses is at most (n-1)/2. First, any nonwitness must be in $\mathbf{Z}^*_{\mathbf{n}}$ as it must satisfy $a^{n-1} \equiv 1 \pmod{n}$, i.e., $a \cdot a^{n-2} \equiv 1 \pmod{n}$; thus, it has an inverse. So we know gcd(a,n) | 1 and hence gcd(a,n) =1. Next we show that all nonwitnessed are contained in a proper subgroup of $\mathbf{Z}^*_{\mathbf{n}}$. This implies the Theorem. There two cases:

1. There is an x such that $x^{n-1} \neq 1 \pmod{n}$. Then we show all the b such that $b^{n-1} \equiv 1 \pmod{n}$ form a group and we're done.
2. The number n is Carmichael number $x^{n-1} \equiv 1 \pmod{n}$ for all x. We'll describe this case next day.