Public Key Encryption

The essence of this procedure is that as far as we currently know, it is difficult to factor a number that is the product of two primes each having many, say 100, digits.

Some Introductory Number Theory

I assume you know what a prime number is. Euclid’s *Elements* contains the first proof that there are infinitely many prime numbers. Although it is completely elementary, it is not obvious. The proof shows that if you know the first $n$ primes, $2 = p_1 < p_2 < \cdots < p_n$, then it concludes there is a larger prime. Note: it doesn’t exhibit a larger prime but just shows that a larger prime exists. Here is the beautiful reasoning. Let

$$N = p_1 p_2 \cdots p_n + 1.$$ 

Either $N$ is prime or it isn’t. If it is prime, then we are done. If it isn’t, then it is divisible by a prime. However, it is clearly not divisible by any of $p_1$, $p_2$, \ldots, $p_n$ since upon division, they all give a remainder of 1. Thus it is divisible by some prime larger than $p_n$.

**Notation:** We write $a \equiv b \pmod{n}$ to mean that the integers $a$ and $b$ have the same remainder when divided by $n$. This is equivalent to saying that $a - b$ is divisible by $n$. Here are some immediate consequences. Obviously the only possible remainders after dividing by $n$ are 0, 1, 2, \ldots, $n - 1$.

If $a \equiv b \pmod{n}$ and $c \equiv d \pmod{n}$ then $a + c \equiv b + d \pmod{n}$.

If $a \equiv b \pmod{n}$ and then $ac \equiv bc \pmod{n}$ for any integer $c$.

A natural question is, if $ab \equiv 0 \pmod{n}$, does it follow that either $a \equiv 0 \pmod{n}$ or $b \equiv 0 \pmod{n}$ (or both)? This is false, as illustrated by the simple counterexample $2 \cdot 3 \equiv 0 \pmod{6}$, although neither 2 nor 3 are divisible by 6. However, if $n$ is a prime number, this is true.

**Theorem** If $p$ is a prime and $ab \equiv 0 \pmod{p}$, then either $a \equiv 0 \pmod{n}$ or $b \equiv 0 \pmod{n}$ (or both).

One reasonable approach to proving this is to use the fact that every integer $n$ can be factored into a product of primes, as $52 = 2^2 \cdot 13$, and this factoring is unique except for possibly reordering the way this product is presented, as $52 = 13 \cdot 2^2$. However, the customary proof of this factorization into a product of primes uses this theorem so the reasoning would be circular. We’ll simply accept the result.

**Corollary** If $b$ and $n$ have no common factors and $ab \equiv 0 \pmod{n}$, then $a$ is divisible by $n$, that is, $a \equiv 0 \pmod{n}$. 


Fermat’s Little Theorem and Euler’s Generalization

**Fermat:** If $p$ is a prime and the integer $a$ that is not a multiple of $p$, then $a^{p-1} \equiv 1 \pmod{p}$. An immediate consequence is $a^p \equiv a \pmod{p}$ for any $a$.

**Proof:** Using the previous theorem we first assert that the integers $a$, $2a$, $3a$, $\ldots$, $(p-1)a$ are all distinct mod $p$. To see this, assume that $ka \equiv \ell a \pmod{p}$ for some integers $k \geq \ell$. This means that $(k - \ell)a$ is a multiple of $p$. But $a$ is not divisible by $p$. Thus $k - \ell$ must be divisible by $p$. Since $1 \leq \ell < k < p - 1$, this is impossible. Since $a$, $2a$, $3a$, $\ldots$, $(p-1)a$ are all distinct mod $p$, then mod $p$ they must just be $1, 2, \ldots, p - 1$, possibly in some other order, so

$$(a)(2a)(3a) \cdots (p-1)a \equiv (1)(2)\cdots(p-1) \pmod{p},$$

that is

$$[a^{p-1} - 1](1)(2) \cdots (p - 1) \equiv 0 \pmod{p}. \quad (1)$$

Since $(1)(2)\cdots(p - 1)$ is not divisible by $p$, then $a^{p-1} \equiv 1 \pmod{p}$, as we wished to prove.

One can use this for the interesting (and useful to cryptography) application to show that certain numbers $n$ are not prime without factoring them. For instance, one can show that $n = 1763$ is not a prime. If it were a prime, then by Fermat with $a = 2$, $2^{1762} \equiv 1 \pmod{1763}$. But by a direct computation $2^{1762} \equiv 742 \pmod{1763}$. This crude test is fairly efficient even for candidates $n$ having several hundred digits.

Euler generalized Fermat’s theorem to $\pmod{n}$ where $n$ is not necessarily a prime. The above proof of Fermat’s Theorem fails since equation (1) becomes

$$[a^{p-1} - 1](1)(2) \cdots (n - 1) \equiv 0 \pmod{n}, \quad (2)$$

which may be trivially true because $(1)(2)\cdots(n - 1)$ may be divisible by $n$, as happens even when $n = 6$. However, Euler observed that the above proof of Fermat’s result still works if in the product $(a)(2a)(3a) \cdots (n - 1)a$ one includes only the factor $ka$ when $k$ and $n$ have no common divisors (other than 1). For any integer let $\phi(n)$ be the number of integers $1, 2, \ldots, n - 1$ that have no common divisors with $n$ (we call this the Euler $\phi$ function).

**Example 1.** If $p$ is a prime, since none of $1, 2, \ldots, p - 1$ have a common divisor with $p$, then $\phi(p) = p - 1$.

**Example 2.** We compute $\phi(10)$. Now $10 = 2 \times 5$ The only integers $1, 2, \ldots, 9$ that have a common factor with 10 are those that are divisible by either 2 or 5. These are the integers 2, 4, 6, 8, and 5. These are $4 + 1 = 5$ integers so

$$\phi(10) = 9 - 5 = 4.$$
Example 3. Say $n = pq$, where $p$ and $q$ are distinct primes. We will compute $\phi(n)$. This is like the previous example.

Which numbers $1, 2, \ldots, pq - 1$ have a common divisor with $pq$? These common divisors can only be multiples of $p$ or $q$, so they are:

$$p, \ 2p, \ 3p, \ldots, (q - 1)p \quad \text{and} \quad q, \ 2q, \ 3q, \ldots, (p - 1)q.$$ 

Thus $(q - 1) + (p - 1)$ integers are not relatively prime to $pq$ so the rest are. The number is $\phi(pq) = (pq - 1) - [(q - 1) + (p - 1)] = pq - p - q + 1$, that is

$$\phi(pq) = (p - 1)(q - 1) = \phi(p)\phi(q).$$

Euler’s Generalization: If $a$ is relatively prime to $n$, then $a^{\phi(n)} \equiv 1 \pmod{n}$. Consequently $a^{\phi(n)+1} \equiv a \pmod{n}$.

Proof This just imitates the above proof of Fermat’s Theorem. In equation (1) only use the factors $k_1a$ where $k_1$ and $n$ have no common divisor (other than 1). Obviously $k_1 = 1$. There are $\phi(n)$ such factors. Then equation (1) is replaced by

$$(a)(k_2a)(k_3a)\cdots(k_{\phi(n)}a) \equiv (1)(k_1)\cdots(k_{\phi(n)}) \pmod{n},$$

that is

$$[a^{\phi(n)} - 1](1)(k_2)\cdots(k_{\phi(n)}) \equiv 0 \pmod{n}.$$ 

Since none of $k_1, k_2, \ldots k_{\phi(n)}$ have any common factors with $n$ (other than 1), we conclude that $a^{\phi(n)} - 1$ must be divisible by $n$, as desired.

Special Case If $a$ is relatively prime to $pq$ for any distinct primes $p$, $q$, then

$$a^{(p-1)(q-1)} \equiv 1 \pmod{pq}.$$ 

The next corollary states that if $n = pq$ we can drop the assumption that $a$ is relatively prime to $pq$.

Corollary Let $n = pq$, where $p$ and $q$ are primes. Then for any integers $a$ and $k$ we have $a^{k\phi(n)+1} \equiv a \pmod{n}$. [If $n = p$ and $k = 1$ this is Fermat’s Theorem].

Exercise: If $n = 10$, verify this with $a = 8$ and $a = 6$.

Proof of the Corollary

Case 1. If $a$ is divisible by both $p$ and $q$, the assertion is obvious.

Case 2. If $a$ is not divisible by either $p$ or $q$, then $a$ is relatively prime to $n = pq$ so this follows from the special case of Euler’s generalization of Fermat’s theorem.

Case 3. If $a$ is divisible by one of $p$ and $q$, say $p$ but not $q$, then clearly $a^{k\phi(n)+1} - a = a[a^{k\phi(n)} - 1]$ is divisible by $p$. 

3
Since \( a \) is not divisible by \( q \), then by Fermat’s theorem \( a^{\phi(q)} = a^{q-1} \equiv 1 \pmod q \) so 
\[
a^{k\phi(n)} = [a^{\phi(q)}]^{k\phi(p)} \equiv 1^{k\phi(p)} \equiv 1 \pmod q.
\]
In other words, \( a^{k\phi(n)} - 1 \) is divisible by \( q \). Consequently 
\[
a^{k\phi(n)+1} \equiv a \pmod q.
\]
Thus \( a^{\phi(n)+1} - a \) is divisible by both \( p \) and \( q \) so it is divisible by \( pq \). QED

Computing \( a^k \pmod n \) efficiently (to encrypt messages)

We need to be efficient since computing \( a^k \) directly. For instance \( 12^{15} \) is too large to compute on most calculators. The idea is to observe that if you have computed \( b \pmod n \), then it is easy to compute \( b^2 \pmod n \). To use this observation write \( k \) as a sum of powers of 2, that is, in base 2. For instance, to compute \( 12^{15} \pmod 6 \) write \( 15 = 2^3 + 2^2 + 2^1 + 2^0 = \text{base 2} 1111 \). Then 
\[
12^{15} \equiv 12^{(2^3)} \cdot 12^{(2^2)} \cdot 12^{(2^1)} \cdot 12^{(2^0)}.
\]
Notice that each of the factors on the right side is the square of the factor to its right; for instance \( 12^{(2^2)} = [12^{(2^1)}]^2 \), so, beginning from the final factor on the right, one can efficiently compute the successive factors \pmod 6. As an exercise, carry this out on a small calculator – where computing \( 12^{15} \) directly would be impossible.

The following is a recipe that carries out this procedure to compute \( a^k \pmod n \) efficiently. It is straightforward to make this into a computer program.

\[
x = 1 \text{ (initialize the answer } x \text{)}
\]

while \( k > 0 \) repeat:

- \( e = 0 \) if \( k \) is even, \( e = 1 \) if \( k \) is odd, so \( e = k - 2[k/2] \) (here \([k/2]\) means the largest integer in \( k/2 \), so \([5/2] = 2\) and \([6/2] = 3\)).
- If \( e = 1 \), replace \( x \) by \( ax \) and reduce \( \pmod n \) (if \( e = 0 \) do nothing).
- Replace \( a \) by \( a^2 \) and reduce this \( \pmod n \).
- Replace \( k \) by \( (k - e)/2 \), that is, drop the unit digit in the binary expansion of \( k \) and shift the remaining digits one place to the right.

When done (so \( k = 0 \)), then \( x \equiv a^k \pmod n \), as desired. You might find it interesting to ponder how this implements the procedure; I’d use it to compute both \( 12^{15} \pmod 6 \) and \( 12^{13} \pmod 6 \) on a hand calculator.
Alice → Bob (by Rivest, Shamir, & Adelman)

Task: Alice wants to send a message to Bob, say in a letter, but wants to keep its contents a secret from anyone along the way who might steal the letter and read it. She uses public key cryptography. This relies on the widely believed but unproved assumption that it is difficult to factor a large number (say 200 digits) that is the product of two large primes.

Public, known to everyone: \((n, e) = \text{Bob's public key, where}\)

- \(n = p \times q\), where \(p\) and \(q\) are primes known only to Bob.
- \(e\): satisfying \(e < n\) and relatively prime to \(\phi(n) = (p-1)(q-1)\). \(e\) is the public exponent.

An essential ingredient here is that there is a trusted repository for public keys. If you look there, the keys you get will be valid.

Private, known only to Bob:

- The above primes \(p\) and \(q\).
- The private exponent \(d\) with the property that \(ed - 1\) is divisible by \((p-1)(q-1)\), that is, \(ed \equiv 1 \pmod{\phi(n)}\), which is equivalent to \(ed = k\phi(n) + 1\) for some integer \(k\).

Example 4: \(p = 23, q = 97\) so \(n = pq = 2231\)

\((p-1)(q-1) = 22 \times 96 = 2112\) so say \(e = 5\).

We want \(ed - 1 = k(p-1)(q-1)\) for some \(k\), that is, \(5d = 1 + 2112 \times k\). \(k = 2\) works so \(d = 4225/5 = 845\) is OK.

Example 5: \(p = 97, q = 109\) so \(n = pq = 10573\) and \((p-1)(q-1) = 96 \times 108 = 10368\) so say \(e = 11\).

We want \(ed - 1 = k(p-1)(q-1)\) for some \(k\), that is, \(11d = 1 + 10368 \times k\). \(k = 9\) works so \(d = 8483\) is OK.

For those who know more algebra, since

\[ed \equiv 1 \pmod{\phi(n)},\]

\(d\) is the multiplicative inverse of \(e\) and can always be found using the Euclidean algorithm.
**Alice Encrypts the message for Bob:**

Say the message has been transformed into an integer \( 0 \leq M < n \) (if the message is longer than \( n \) digits, then first break it into smaller parts, each of which has less than \( n \) digits). Her encrypted message is:

\[
m \equiv M^e \pmod{n} \quad \text{(trapdoor function)}.
\]

**Bob Decrypts the message:** He computes \( m^d \mod{n} \).

**Claim:** \( m^d = M \), so Bob has recovered Alice’s message.

**Proof:** Since \( m = M^e \), then \( m^d \mod{n} \equiv M^{ed} \mod{n} \). But \( d \) was chosen so that \( ed \equiv 1 \mod{\varphi(n)} \). Consequently \( ed = k\varphi(n) + 1 \) for some integer \( k \). Thus by the Corollary

\[
M^{ed} = M^{k\varphi(n)+1} \equiv M \pmod{n}.
\]

**Trapdoor Functions for Private Communication**

The above encryption/decryption procedure satisfies the criteria proposed earlier by Diffie and Hellman (1976).

- It will change any positive integer \( x \) into a unique positive integer \( y \).
- It has an inverse that changes \( y \) back to \( x \).
- Efficient algorithms exist to compute both the forward function and its inverse.
- If only the function and its forward algorithm are known, it is computably infeasible to discover the inverse algorithm.

**Digital Signatures:**

Alice wants to send her “signature” to Bob to send her some money. The signature is not secret. Bob wants to know that:

1. The signature has not been tampered with.
2. It really is from Alice.

**Procedure:**

Alice makes a digital signature \( s \equiv S^d \mod{n} \) where \((n, d)\) are Alice’s own private key and \( S < n \) is her public signature.

She sends both \( s \) and \( S \) to Bob.

Bob computes \( x \equiv s^e \mod{n} \), where \((n, e)\) are Alice’s public key. If \( x = S \), then he is assured the message is both authentic and from Alice.

**Proof:**

\[
x \equiv s^e \pmod{n} \equiv S^{ed} \pmod{n} \equiv S \pmod{n}
\]