Key Generation

1. choose two distinct large prime numbers p and q, each $b/2$ bits long.

2. Compute $n = pq$

n is either b bits or $b-1$ bits.

3. Compute $\varphi(n) = \varphi(p)\varphi(q) = (p-1)(q-1)$

Because $n = pq$ and p and q are coprime

Because p is prime thus has $p-1$ rel. prime numbers less than it. same for q.

4. Compute e s.t. $\gcd(e, \varphi(n)) = 1$

e usually $= 2^{16}+1 = 65{,}537$

e is public key (with known n)

5. Compute d s.t. $de \equiv 1 \pmod{\varphi(n)}$

d is kept as the private key (and keep n too)

(do not release p, q, or $\varphi(n)$, since those can be used to compute d.)

Encryption:

to encrypt $0 \leq m < n$,

m is being encrypted

$c \equiv m^e \pmod{n}$

c is the ciphertext

Decryption:

$$m = c^d \pmod n$$

Proof:

Fermat's little theorem:

$$a^{(p-1)} \equiv 1 \pmod p$$, if p is prime and p does not divide a.

We must show:

$$(m^e)^d \equiv m \mod (pq)$$, where p + q are distinct positive integers, and e + d satisfy $ed \equiv 1 \pmod{\varphi(n)}$, where $\varphi(n) = (p-1)(q-1)$

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

because p and q are coprime, by Chinese Remainder Th'm,

$$m^{ed} \equiv m \pmod p \quad \text{and} \quad m^{ed} \equiv m \pmod q$$

$\Updownarrow$ implies (iff) when q and p are coprime

$$m^{ed} \equiv m \pmod{pq}$$

Proof (con'd)

to prove $m^{ed} \equiv m \pmod{p}$:

$$m^{ed} = m^{k(p-1)(q-1)+1} \equiv m \pmod{p}$$

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

$$m \cdot (m^{p-1})^{k(q-1)} \equiv m \pmod{p}$$

<by FLT>

$$m \cdot 1^{k(q-1)} \equiv m \pmod{p}$$

$$m \equiv m \pmod{p}$$

Similarly for $m^{ed} \equiv m \pmod{q}$, ...

qed

Practical Matters

- when $m=0$, $m^e = 0$, thus no "encryption happens"
- when $m=1$, $m^e = 1$, thus no "encryption happens"
- in general, when $m < n^{1/e}$, no encryption happens because the "(mod n)" never comes into play, that is, $c = m^e$, thus m can be found by doing $\sqrt[e]{c} = m$
- when $m = n-1$,

$$m^e \pmod{n} = (n-1)^e \pmod{n}$$