RSA Encryption

One issue with Diffie-Hellman is that you can't use it to send an intended message to the recipient. Notice that neither Alice nor Bob ever get to "choose" the secret key that they share. Both contribute, but this system couldn't be used to encrypt an actual message.

So, even after this was out in the public realm, people still wanted to know if a full cryptosystem could be done publicly…

Ron Rivest - young professor at MIT
Adi Shamir - Weitzman Institute, then MIT
Len Adleman - USC

1. The way this works is that a user generates both public keys and private keys for themselves, without communicating with anyone.

2. User posts the public keys.

3. Anyone can use the public keys to send the user a message. That ciphertext can only be read by the user and no one else.

Key is that knowing the public keys doesn't give information about the private keys or how to "undo" the operation done by the public keys, so anyone can send the message, but the user who created the keys is the only person who can read the message.

1. User pick two large prime numbers, p and q, both are secret keys.
2. User calculates $n = pq$, n is a public key.
3. User picks a random integer e, $1 < e < \varphi(n)$, where $\gcd(e, \varphi(n)) = 1$. e is a public key.
4. User calculates $d = e^{-1} \bmod \varphi(n)$. (Note: $de \equiv 1 \bmod \varphi(n)$.) d is the private key.

User posts (n, e) as the public keys.
The only necessary private key is d.
But, p, q and $\varphi(n)$ must also be kept secret. If someone found out any one of these three things, they could use that information to very quickly calculate d.

How to Send a Message

Message must be an integer greater than 1 and less than n.

To encrypt, just calculate $C = M^e \bmod n$. Send C as your ciphertext.

To decrypt, the user will calculate $C^d = M \bmod n$.

To prove that this works:

$C^d = (M^e)^d = M^{ed} = M^{k\varphi(n)+1} = M^{k\varphi(n)}M^1 = (M^{\varphi(n)})^k M^1$

What do we know about $M^{\varphi(n)} \bmod n$? What does Euler's Theorem tell us about this value? You may assume that gcd(M, n) = 1 in answering this question.

$C^d = (M^e)^d = M^{ed} = M^{k\varphi(n)+1} = M^{k\varphi(n)}M^1 = (M^{\varphi(n)})^k M^1 \equiv 1^k M^1 \equiv M \bmod n$

Why is it that knowing p, q or phi(n) divulges the value of d?

Case 1: You find out p. Then you calculate q = n/p. Once you know p and q, then you calculate phi(n) = (p-1)(q-1). Then you can calculate $d = e^{-1} \bmod \text{phi}(n)$.

Case 2: You find out q. Then you calculate p = n/q. Once you know p and q, then you calculate phi(n) = (p-1)(q-1). Then you can calculate $d = e^{-1} \bmod \text{phi}(n)$.

Case 3: You find out phi(n)…that is all you need. You know e already and you know n, so just do $d = e^{-1} \bmod \text{phi}(n)$.

Incidentally, if you know phi(n) and n, you can also recover p and q:

n = pq
phi(n) = (p-1)(q-1)

Here is an example:

91 = pq
72 = (p-1)(q-1)
72 = pq - p - q + 1
72 = 91 - p - q + 1
72 = 92 - p - q
p + q = 20

$pq = 91$
$p(20-p) = 91$
$20p - p^2 = 91$
$p^2 - 20p + 91 = 0$
$(p - 7)(p-13) = 0$, ta da!
But with large integers you wouldn't know how to factor this, but you could do the quadratic equation…

<u>In general</u>
$Phi(n) = n - p - q + 1$
$p + q = n + 1 - phi(n)$
$pq = n$
$p\ (n+1-phi(n) - p) = n$
$(n+1-phi(n))p - p^2 = n$
$p^2 - (n+1-phi(n))p + n = 0$

From here, use the quadratic formula to solve for p.