
**A Guide To RSA**
**by**
**Robert Yates.**

Topics



# Introduction

Welcome to my latest paper which is on the subject of RSA.
My inspiration for this paper comes from the fact that until recently i had no knowledge of the RSA scheme and i want to share with you everything i learnt and present it in a way which would of benefited me more if i had of had this paper now.

This paper also assumes the reader has little knowledge of math and is not intended to be patronising however it might be difficult to follow if you are just reading it, i would recommend taking a pen and paper and following all calculations to make sure you really under each step :)

# What is RSA

RSA is an algorithm for a public key encryption system, i.e this is an asymmetric cipher, one person has a private key and gives out a public key, anyone who has the public key can encrypt some message or data then only the person with the private key can read this message, not even the person who encoded the original message with the public key can now decode this. The idea behind this is to prevent “man in the middle attacks”.

The algorithm was described by Ron **R**ivest, Adi **S**hamir and Leonard **A**dlerman in 1977, however its believed that two British mathematicians in the British Military had also discovered this system earlier on but due their occupations this had to remain secret.

# Mod-Exponentiation

We start by explaining Mod-Exponentiation, better yet lets split that up and explain mod and exponentiation briefly. When talking mathematically, mod is an operation which gives you the remainder of a division and exponentiation is an operation where you take a number and raise it to the power of another, written often like 2^2 or $2^2$.

So, for example, 2^2 is equal to 4, and (2^2) mod 3 is equal to the remainder of the integer division of 4 by 3: thus, (2^2) mod 3=1.

So, mod-exponentiation is one of the key operations you need to know also when doing RSA, the operation is as follows,

*x^y mod z*

It is also known as PowMOD or PMOD, i highly recommend you get a good calculator with this function.

In the case of RSA we usually say that

x = plain/ciphered_text

y = key

and z is our modulus.

The result of PMOD is our wanna-be ciphertext.

Lets take a look at this:

Key = 5, Modulus = 50, encode numbers 1 to 5

1^5 mod 50 = 1

2^5 mod 50 = 32

3^5 mod 50 = 43

4^5 mod 50 = 24

5^5 mod 50 = 25

. ..

looks go so far, but what about this

11^5 mod 50 = 1 ?

11 is the same as encoding 1, so this would make proper decryption of ciphertext '1' impossible, you would not be able to know if you were decrypting the number 11 or the number 1.

I hope this is clear so far, i am currently explaining how something in theory would be encrypted with RSA and the rules in which it must be done, decryption and key pair generation will be explained further on, but first we have to look deeper into what the RSA algo really is, and explain how a good key pair and modulus can be generated for RSA.

## Euler's Theorem

Ok Euler's Theorem is as follows,

$$x^{\varphi(z)} \bmod z = 1$$

Now at this point it will seem like for everything i explain i have to explain something else until your brain implodes on information, but as i would say, don't panic :-).

Let us examine each part of the theorem one by one.

φ(z) is Euler's Totient function, which means when z is a positive integer then the result is the number of positive integers less than or equal to z that are Coprime to n.

Coprime? Thought you would ask that, ok so basically x and y are considered to be Coprime when they have no common factor but 1, same can be said if their greatest common divisor is 1, for example

8 and 7 are Coprime but 14 and 7 are not since they can both be divided by 7.

So, the φ(z) Totient function is simply the number (the count) of the Coprimes to z. The special characteristic of prime numbers is the fact that the Totient of any prime number -say N- is always equal to N-1, as you can quickly understand by looking the example below:

Take number 7: (prime)

-->1) 1 is relatively prime to 7

-->2) 2 is relatively prime to 7: 7/2 = 3.5

-->3) 3 is relatively prime to 7: 7/3 = 2.3

-->4) 4 is relatively prime to 7: 7/4 = 1.7

-->5) 5 is relatively prime to 7: 7/5 = 1.4

-->6) 6 is relatively prime to 7: 7/6 = 1.1

phi -the reading of the sign φ- is then N-1=>6

Take number 6: (not prime)

-->1) 1 is relatively prime to 6

-->2) 2 is NOT relatively prime to 6: 6/2 = 3, so 2*3=6!

-->3) 3 is NOT relatively prime to 6: 6/3 = 2, so 3*2=6!

-->4) 4 is relatively prime to 6: 6/4 = 1.5

-->5) 5 is relatively prime to 6: 6/5 = 1.2

phi is NOT N-1, but 3.

(it might seem obvious, but we use only integer math for this stuff)

Ok so lets get back to the Euler's Theorem

$$x^{\varphi(z)} \bmod z = 1 \pmod z$$

this actually means that if Z is a prime then φ(z) would always equal z-1, which we use later in our new equation.

Imagine the number 2, with a Z of '7'. This theorem says that 2 raised to 6 and divided by 7 can ONLY have a remainder of 1.

Now think: we know that if X=Y, then X*X=Y*Y (or, alternatively, we always know that if X=Y then X*X=Y*X). What does this means?

$$x^{\varphi(z)} \bmod z * x^{\varphi(z)} \bmod z = 1 \pmod z * 1 \pmod z$$

or, with simple numbers,

$$2^6 \bmod 7 * 2^6 \bmod 7 = 1 \bmod 7 * 1 \bmod 7$$

At this point, we can group things together on left and right (2 numbers with the same base and different exponent multiplied each other are just the same number with the exponent added, 2^**2** * 2^**3** = 2^(**2**+**3**) ):

2^6 mod 7 * 2^6 mod 7  is  2^(6+6=6*2=12) mod 7

obviously, 1*1 is 1, so we obtain the following:

x^(2*φ(z)) mod z = 1*1 (mod z)

And so? So, we can easily see that the above result will be true no matter which number we multiply phi and 1 by:

x^(3*φ(z)) mod z = 1*1*1 (mod z)

x^(4*φ(z)) mod z = 1*1*1*1 (mod z)

x^(5*φ(z)) mod z = 1*1*1*1*1 (mod z)

x^(W*φ(z)) mod z = 1*1*1*..up to W times (mod z)

...so? What we know till now is that we have a math formula that makes our X (our hypothetical text) raised to any multiplier of φ(z) returns 1 in modular arithmetic. Not very useful, until we try to multiply each side by x and see:

x^(W*φ(z)) mod z * x = 1 (mod z) * x

-->x^(W*φ(z)+1) mod z = x (mod z)

So, we have a math relation that, given an x raised to any integer multiple of totient and then added 1 RETURNS ITSELF!

Since we know that the totient has a special effect when we evaluate it on prime numbers, it is equal to the prime less one, we can write that

X^(z-1) mod z = 1

and

X^(z-1)*X mod z = 1*X (mod z)

which equals to X^(z) mod z = X mod z. In numbers it is:

2^(7-1) mod 7 = 1

and

2^(7) mod 7 = 2

It is nice, but not yet useful for encryption. But we know, at least, a relation that given a base, an exponent and a modulus return itself.

Let's dig more. Think about a number built by two prime numbers, what will its totient be? Interesting enough, the totient of a number built by two prime numbers p and q will be a multiplication of the totient of the two numbers,(p-1)*(q-1). so the totient φ of the number p*q is

$$\varphi(p*q) = \varphi(p)*\varphi(q) = (p-1)*(q-1)$$

i.e. lets take 2 prime numbers p(83) and q(53):

p*q = 4399

p-1*q-1 = 4264

φ(p*q) = p-1*q-1= 4264

Now, if we replace the 'z' used above with the number 'p*q' (4399) so that:

$$X^{\varphi(p*q)} \bmod (p*q) = 1 \bmod (p*q)$$

we know that the totient φ(p*q) is equal to (p-1)*(q-1) because p and q are primes, so we have that

$$x^{(p-1)*(q-1)} \bmod (p*q) = 1$$

or with numbers:

$$X^{4264} \bmod 4399 = 1 \pmod{4399}$$

with q and p that are prime numbers, 83 and 53.

Lets make an example:

10^4264 mod 4399 = 1

20^4264 mod 4399 = 1

20^4264 mod 4399 = 1

now comes an interesting part from all the mathematics above.
So, from all above the mathematics it is so that

$$X^{\varphi(z)}*X \bmod z = 1*X \pmod z$$

equal to

$$X^{\varphi(z)+1} \bmod z = 1*X \pmod z$$

Now, if we place $z = p*q$ and so $\varphi(z) = \varphi(p*q) = (p-1)(q-1)$ that

$$X^{(p-1)(q-1)} *X \bmod z = 1*X \pmod z$$

which can be expressed as a +1 to the exponent of X, so **(p-1)(q-1)+1**.

$$X^{ (p-1)*(q-1)+1 } \bmod z = X \bmod Z$$

Thus, if we add 1 to the prior exponent, 4264 that is our p-1*q-1, look what happens:

$10^{4265} \bmod 4399 = 10$

$20^{4265} \bmod 4399 = 20$

$20^{4265} \bmod 4399 = 30$

What is very interesting of the new formula is that you can discover the number 4265 only if you can FACTORIZE the modulus 4399 in order to obtain the p and q needed to evaluate $(p-1)(q-1)+1$

# RSA Algorithm

At this point we are ready to show how RSA is formed from these equations, the above formula is important because currently we see the plain text (10,20,30) is only again retrieved when raised to the power of 4265 and using the modulus of 4399.

10^4265 mod 4399 = 10

we can now split this into 2 stages, an encryption stage and a decryption stage.

We need a encoding key(e) and a decoding key(d).

We start with E, E should be **coprime** to (p-1*q-1)

The security of RSA does not depend on E, we can check the primes against (p-1*q-1) until we find one. 3, 5, 7 etc (the security of RSA depends on how hard it is factoring the number we choose as modulus).

3 is coprime to 4265, lets choose 3 as our E.

In order to calculate D we must do an inverse mod operation like so

E^-1 mod (p-1*q-1). Inverse mod can be found on any good calculator and is often shown as **M^-1** i am using Hexprobe Multibyte Calculator

so 3^-1 mod 4265 is 2843(d)

now take 3 as our encoding key(e) and 2843 as our decoding key(d).

The RSA rules are now

C = P^E mod m

P = C^D mod m

P = Plain Text

C = Cipher Text

Lets test this and encode the number 1000

1000^E(3) mod m(4399) = C(1724)

1000 encrypts to 1724, lets decode it now.

1724^D(2843) mod m(4399) = P(1000)

as you can see it decodes back, however the P must be less than M for this to work which is why large primes are used.

## RSA Security

So i have my keys e,d,m and i want to exchange some secret information with someone, i send them my m and e.

The other person has m(4399) and e(3) they encode 1000 and send me back 1724, now i can decode this with d(2843) anyone listening in the middle only has C,E,M witht his information they could begin an attack which would involve the following.

M = 4399

P*Q = 4399

E*D =  4265

E=3

D=? (2843)

The strength relies on the ability on the attacker being able to factor M back to P and Q, if M was a 2048bit number this could take years.

If you found these document useful or use it for any teaching purposes, please let me know.

+[yates]

03/JAN/2008

*Email: cm9iZXJ0LnlhdGVzQHJldmVyc2UtZW5naW5lZXJpbmcuaW5mbw*

*(base64 encoded)*