

Forum:[Hack Forums](#)

Thread: [\[Zykl0n's B Tutorial\] RSA : encryption, decryption and security - Part 1/3](#)

Error667 - October 24, 2012, 8:29 am

Hello guys,

I'm Zykl0n B, and I hope you will understand this writing. It took me a long time to make it understandable. So, take a pencil, a paper sheet and let's start.

Warning : This tutorial is based on advanced concepts in mathematics.

Overview

Rivest Shamir Adleman (RSA almost always abbreviated) is an asymmetric cryptographic algorithm. This means this algorithm is based on the use of a public key and a private key. This algorithm was described in 1977 by Ronald Rivest, Adi Shamir and Leonard Adleman but remained classified until 1997.

The first thing you have to know is that the public key is used to encrypt while the private key is used to decrypt.

The public key is a key that is accessible by anyone wishing to encrypt information, private key is itself dedicated to the person who created the key pair (private key + public one).

These two keys have mathematical values in common, but it is theoretically impossible to find the private key knowing public key.

T H E O R Y

A - Creation of the keys

Reminder : A prime number is a natural number that has exactly two distinct divisors (positive and integers) : 1 and itself.

1 - Let P and Q be two distinct primes.

2 - Note $N = P * Q$

N is used as the modulus for the public key and the private key.

But what is the modulus ? Well, the modulus is an integer found by doing the Euclidean division of 2 numbers.

Code EG : $8 \bmod 3 = 2$

$0 \bmod 1 = 1$

3 - Calculate the Euler totient of n noted $\phi(n)$. $\phi(n) = (p-1)(q+1)$

The Euler totient function is to determine for a number N (which is a positive integer), the number of positive integers (inferior or equal to N) whose are coprime with N.

Code To resume : $f(N) = X$ positive integers $< \text{ or } =$ to N, coprime with N.

We can see that if there was a curve to be plotted, it does not stop growing, which is quite logical.

Code To illustrate : $\phi(12) = 4$ because for the numbers from 1 to 12, only four numbers (1, 5, 7, 11) are coprime with 12.

Reminder : Two numbers are coprime if their GDC (Greatest Common Divisor) is equal to 1. To do it manually, the Euclidean division can be used.

Is 10 coprime with 12 ?

Answer : --> $12/10$: the rest is 2

--> $10/2 = 5$, the rest is 0.

10 is not coprime with 12.

4 - Choose e , an integer coprime with $\phi(n)$ with : $1 < e < \phi(n)$.

5 - According to the 'theorem of Bachet Méziriac', As e is prime with $\phi(n)$, there is an integer d such that $ed = 1 \bmod[\phi(n)]$.

THEOREM :

Given two integers A and B both non-zero, if D is the GDC of A and B then there exist two integers X and Y such that: $(X.A) + (Y.B) = d$.

SOME EXPLANATIONS ;

Lets change the letters : A is becoming e

B is becoming $\phi(n)$

Furthermore, we know that e and $\phi(n)$ are coprime : GDC is 1.

$$(X.e) + (Y.\phi(n)) = 1$$

The integer d is the decryption exponent.

So, the pair (modulus, decryption exponent) is the private key, whereas the pair (modulus, encryption exponent) is the public key.

These pairs are respectively noted (n, d) and (n, e) .

B - Encryption of the message

Note : As you guess, the entire message must be converted to numbers with RSA. We'll see later how to do.

Let M be the message : $0 \leq M < n$.

The encrypted message C would be : $C = M^e \bmod n$

C - Decryption of the message

One can decrypt the message only by using the private key d .

The equation is : $M = C^d \pmod{n}$

We note that to encrypt a message, we have just to know e and n . But to decrypt it, we must know d and n .

U S E

A - Creation of the keys

$$1 - P = 7 ; Q = 13$$

$$2 - N = 91$$

$$3 - \phi(n) = (7-1)(13-1) \\ = 72$$

$$4 - e = 11$$

$$5 - ed = 1 \pmod{\phi(n)} \\ 11d = 1 \pmod{72} \\ d = 59$$

Public key is $(91, 11)$. Private key is $(91, 59)$.

B - Encryption of the message

Let say the message we would like to encrypt is PASSWORD. As I said before, this must be converted in integers to make the algorithm work.

We can use an ASCII table, which you probably know if you're working with XSS sometimes, to convert each character to decimal (as we could use OCTAL to convert, while it's an integer).

CHAR/DECIMAL :

P is 80
A is 65
S is 83
S is 83
w is 87
O is 79
R is 82
D is 68

That's great : all these decimal are smaller than n .

Now, it is highly advisable to convert each decimal in binary. But to avoid having to do tedious calculations, I'll keep this form.

The decrpyted message is : 8065838387798268

Now, let's encrypt it.

$$C = M^e \pmod{n}$$

$$C = 8065838387798268^{11} \pmod{91}$$

$$C = 9400364107053231309682443579252936578186534197931040402400118717510362202565222163748661207187771303770343666448864073199593507317716615352680105346439437176514909978316767232 \pmod{91}$$

C - Decryption of the message

$$M = C^d \pmod{n}$$

$$M = (9400364107053231309682443579252936578186534197931040402400118717510362202565222163748661207187771303770343666448864073199593507317716615352680105346439437176514909978316767232 \pmod{91})^{(59)} \pmod{91}$$

RELOADED - With an easier integer.

Message is 2.

$$\begin{aligned} C &= 2^{11} \pmod{91} \\ &= 2048 \pmod{91} \\ &= 46 \end{aligned}$$

--

Sometimes, there're ways to get C and avoid overflows. By decomposing, we obtain :

$$\begin{aligned} C &= (2^8 \cdot 2^2 \cdot 2^1) \pmod{91} \\ &= ((2^8 \pmod{91}) \cdot (2^2 \pmod{91}) \cdot (2^1 \pmod{91})) \pmod{91} \end{aligned}$$

We have :

$$\begin{aligned} 2^8 \pmod{91} &= 74 \\ 2^2 \pmod{91} &= 4 \\ 2^1 \pmod{91} &= 2 \end{aligned}$$

So :

$$\begin{aligned} 2^{11} \pmod{91} &= 74 \cdot 4 \cdot 2 \pmod{91} \\ &= 46 \end{aligned}$$

--

$$\begin{aligned} M &= 46^{(59)} \pmod{91} \\ &= 2 \end{aligned}$$

End of the first part of this writing.

I will post soon a tutorial on security and attack methods (theory and practice), which will also contain advanced mathematics. On the other hand, I'll make an ebook that will be available on my twitter account.

Follow me on twitter: <https://twitter.com/BZyklon>

Regards,

LolZ Gym - October 24, 2012, 8:31 am

See kids, you need math for cracking algorithms.

Nice job OP !

Error667 - October 25, 2012, 10:45 am

Thanks you to have posted the first.

betona - October 25, 2012, 5:44 pm

nice post worth calling hq, on phone thats why reply is short...

thmaster100 - October 25, 2012, 5:49 pm

feel's like I am in "structure de machine" class all over again ^^ <<< I study this shit in french and I never stop loving it ^^

Error667 - October 26, 2012, 1:22 pm

Again, thanks. I'll probably publish this first part of the ebook tonight on twitter.

TheDubWubCub - October 26, 2012, 3:51 pm

Very nice tutorial! Good job man! :D

richie079 - October 26, 2012, 6:30 pm

I have no idea what you just said?!?! Now i know it may be half 2 in the morning, but even at midday this would still fry my tiny mind :/ DAMN! you hot girl in row 2 of my maths class for distracting me, damn you to heck!

Good job though!! clearly a lot of work went in to that

DDDDDEEEEXXXX - October 30, 2012, 7:24 pm

Studied that in school, Euclide's algorithm /typo/ and modular arithmetic + the usage of RSA encryption

Error667 - January 29, 2013, 9:38 am

Yes, this's studied in "special math" in some countries, but their methods are often very different to explain.

G Dragon - January 29, 2013, 1:39 pm

thank you for sharing !

35fxckingchar