RSA Worksheet

Public key cryptography was suggested by Whitfield Diffie and Martin Hellman in 1976, but they were not able to come up with a system that used public keys. The RSA System is an example of public key cryptography. The RSA system was invented by Ronald Rivest, Adi Shamir, and Leonard Adleman. These three men figured out how to apply the concept of public keys in cryptography.

The RSA system has been used in many public places today. RSA is used in Internet Explorer, Netscape Navigator, and in some credit card transactions using the Internet. RSA has been a building block that other public key algorithms have built from.

Part I – A Simple Example

Step 1: The first step in the RSA System is picking two prime numbers (i.e. 3, 5, 7, 11, etc.). These numbers will normally be very large, but for this example, pick two small prime numbers (*p* and *q*).

Prime 1 (*p*) = _________

Prime 2 (*q*) = _________

These numbers should be very large in "real life" systems because the entire security depends on these prime numbers. If an intruder can factor *n* to get *p* and *q*, then the system integrity is compromised.

Step 2: Now form *n* by multiplying *p* and *q* together. $p * q$ must be greater than 25. If they are not, you will have to pick larger values of *p* and *q*.

$n = p * q =$ _________

Step 3: Now form *m* by multiplying *p*-1 and *q*-1 together.

$m = (p - 1) * (q - 1) =$ _________

Step 4: We need to pick a valid RSA encryption exponent. A valid RSA encryption exponent is relatively prime with m. In other words, the greatest common divisor of *a* and *m* must be a number. You can test this in Maple by using the *igcd* command.

$a =$ _________

Step 5: Suppose we want to send the message "ASU" to our receiver. First we will convert the characters in "ASU" to numbers. Each number in the alphabet is assigned a number sequentially (ex. A = 0, B = 1, C = 2, D = 3, …, Z = 25). So our string is now 01820 (Since A = 0, S = 18, and U = 20). Now raise these numbers to *a (mod n)*.

$0^a =$ _________ *(mod n)*

$18^a =$ _________ *(mod n)*

$20^a =$ _________ *(mod n)*

If you combine these numbers you will have the ciphertext. This is the text that is sent to the receiver, along with *a* and *n*.

Step 6: The receiver chooses a prime *b* so that *a*b* = 1 *(mod m)*. This is the decryption exponent that will be used to convert the ciphertext back to plaintext.

b = _________

Step 7: Now the receiver takes the received ciphertext and breaks it up for each character as before. This time, the numbers are raised to the power *b (mod n)*. The resulting text is the plaintext numbers, which can then be converted to letters.

_________ ^{b} = _________ *(mod n)*

_________ ^{b} = _________ *(mod n)*

_________ ^{b} = _________ *(mod n)*

Part II – An Example with “Large” Primes in Maple

From Applications of Abstract Algebra with Maple

```
> p := 40004334421200745800048572956 3;
```

```
> q := 400043344212007458000485729483;

> n := p*q;
    n := 160034677248326680947854891973148610386893801001169513805929

> m := (p-1)*(q-1);
    m := 160034677248326680947854891972348523698469786085168542346884

> a := 1009876890098767901 09;
                                a := 100987689009876790109

> igcd(a, m);
                                          1

> message := `cancelmissionwaitforneworders`;
                          message := cancelmissionwaitforneworders


> to_number := proc(mess)
    local sl, cn, sn, ii, ntable;
    ntable := table(['a'=0, 'b'=1, 'c'=2, 'd'=3, 'e'=4,
                     'f'=5, 'g'=6, 'h'=7, 'i'=8, 'j'=9, 'k'=10,
                     'l'=11, 'm'=12, 'n'=13, 'o'=14, 'p'=15,
                     'q'=16, 'r'=17, 's'=18, 't'=19, 'u'=20,
                     'v'=21, 'w'=22, 'x'=23, 'y'=24, 'z'=25]):
    sl := length(mess);
    cn := 0;
    for ii from 1 to sl do
        sn := ntable[substring(mess, ii..ii)]:
        cn := 100*cn + sn:
    od:
    RETURN(cn):
end:



> plaintext := to_number(message);
   plaintext := 200130204111208181808141322000819051417130422141703041718

> ciphertext := plaintext &^ a mod n;
  ciphertext := 364462939989036865324727789709171856100168737798418831394 9

> igcdex(a, m, 'b', 'y');
                                          1
```

```
> b := b mod m;
      b := 118197061499210406607786132423703550040103749990948373777393

> a*b mod m;
                                        1

> plaintext := ciphertext &^ b mod n;
    plaintext := 200130204111208181808141322000819051417130422141703041718

> to_letter := proc(num)
    local cs, cn, sl, a, b, c, d, e, f, g, h, i, j, k,
          l, m, n, o, p, q, r, s, t, u, v, w, x, y, z,
          ltable, ans;
    ltable := table([0=a, 1=b, 2=c, 3=d, 4=e, 5=f, 6=g,
                     7=h, 8=i, 9=j, 10=k, 11=l, 12=m,
                     13=n, 14=o, 15=p, 16=q, 17=r, 18=s,
                     19=t, 20=u, 21=v, 22=w, 23=x, 24=y,
                     25=z]);
    cn := num;
    sl := floor(trunc(evalf(log10(cn)))/2) + 1:
    ans := ``;
    for i from 1 to sl do
        cn := cn/100;
        cs := ltable[frac(cn)*100];
        ans := cat(cs, ans);
        cn := trunc(cn);
    od:
    RETURN(ans);
end:

> to_letter(plaintext);
                              cancelmissionwaitforneworders
```

Part III – An Example to Try With Maple

Pick two large primes and show how the message “Meet At Base” would be enciphered and deciphered in the RSA System. You may use Maple (or any other technology) and modify the example in Part II to assist you.

RSA Worksheet Information

Worksheet Goals

- To understand the mathematics behind the RSA cryptosystem (and public key systems as a whole).
- To realize why small prime numbers are not used in public key systems
- To learn how the RSA system is applied to everyday technologies of our world.

Required Background

- Modulo arithmetic is necessary (the amount necessary in this worksheet could be explained to someone briefly).
- Good Maple skills (need to be able to understand functions written in Maple). Some of the functions use simple loops and arrays. Basic computer science knowledge would be very helpful in understanding.
- Designed for a college undergraduate. The most limiting factor is Maple experience. Without using Maple, multiplying large primes by hand becomes tedious.