Cyclic RSA Attack

Andrew Bellenir, Joshua Hulst, Dr. Mostafa El-Said (advising)

RSA Explained

Two hypothetical people, Alice and Bob, want to communicate without letting Eve read their messages, even though Eve is capable of seeing all network communication. To do so, they must use an encryption algorithm, such as RSA.

RSA is a form of public cryptography where each person has two keys, a public key and a private key. As the names imply, the public key is accessible by anyone, while the private key is kept to oneself. To send a message to Bob, Alice would use Bob's public key to encrypt the message. Bob would then use his private key to decrypt the message. This way anyone can send Bob an encrypted message, but Bob is the only one capable of reading messages sent to him.

Key Generation

RSA requires two random prime numbers, p and q, to generate the keys. For our example we will use:

$$p=3$$
$$q=17$$

To compute the keys, the following steps are taken:

$n=p*q$ $\qquad 3*17=51$

$\Phi(n)=(p-1)*(q-1)$ $\qquad (3-1)*(17-1)=32$

e is coprime to $\Phi(n)$ such that:

$1<e<\Phi(n)$ $\qquad e=3$

$d*e=1+k*\Phi(n)$ $\qquad d*3=1+k*32$

$\qquad\qquad d=11$

e and n make up our “Public Key”, d and n make our “Private Key”

Message Passing

To encrypt a message we use: $m^e \bmod (n)$

To decrypt we use: $c^d \bmod (n)$

Encryption Example

Alice encrypts a message, “5” using Bob's public key: e=3, n=51

$5^3 \bmod (51)=23$

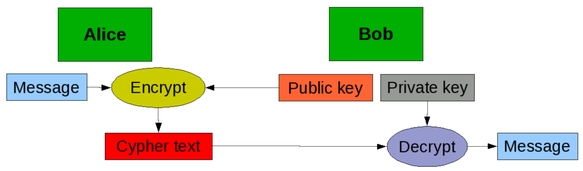
So Alice would send Bob the encrypted message “23”

Decryption Example

Bob would then decrypt the message using his private key: d=11, n=51

$23^{11} \bmod 51=5$

This encryption scheme works as long as the key values have been calculated correctly and are long enough to handle the message



Attack Code

Below is output from a cyclical attack with a key 20 bits long and a message 8 bits long. It took 9875 cycles before the message was decrypted.

```
File  Edit  View  Terminal  Go  Help
jhulst@t61:~/rsa> java RSATest run=1 key=20
keysize: 20 msglength: 8
Message is: 83
cycled 9875 times (m=83)

jhulst@t61:~/rsa>
```

Similarly, the attack below uses a 40 bit key and an 8 bit message. The encryption was cracked after 9516592 cycles.

```
File  Edit  View  Terminal  Go  Help
jhulst@t61:~/rsa> java RSATest run=1 key=40
keysize: 40 msglength: 8
Message is: 200
.....
cycled 9516592 times (m=200)

jhulst@t61:~/rsa>
```

Finally, a 60 bit encryption was not able to be cracked after 10000001 cycles.

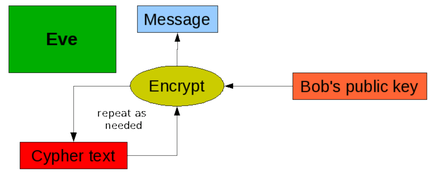
```
File  Edit  View  Terminal  Go  Help
jhulst@t61:~/rsa> java RSATest run=1 key=60
keysize: 60 msglength: 8
Message is: 228
.....
Could not be computed after 10000001 cycles

cycled 10000001 times (m=391939359521834190)

jhulst@t61:~/rsa>
```

Cyclic Attack

If Alice and Bob do not take care in determining their keys, however, and Eve intercepts the encrypted message sent by Alice in the above example, “23”, she could attempt to crack it using a cyclic attack. This type of attack involves continuously re-encrypting the message until the original comes back again:

$\text{encrypt}(23)=23^3 \bmod (51)=29$
$\text{encrypt}(29)=29^3 \bmod (51)=11$
$\text{encrypt}(11)=11^3 \bmod (51)=5$ (this is the original message encrypted by Alice)
$\text{encrypt}(5)=5^3 \bmod (51)=23$ (this is the same encryption calculation originally done by Alice)

When the cycling re-encryption results in the original intercepted value (23), Eve may go back one step in her calculations to recover the decrypted message. What Eve encrypted to a value of 23 must be what Alice encrypted to a value of 23. Thus the code is broken and Eve can read “secure” messages between Alice and Bob.



Protecting Against Cyclic Attacks

Given enough time, the cyclic attack will always be able to break messages sent using the RSA encryption algorithm. Fortunately for those wishing to communicate over secure channels, an astronomical amount of time is required for this attack to break secure implementations RSA. What makes a secure implementation so secure?

•Strong Primes

- It has been suggested that choosing “Strong Primes” for p and q increases the number of cycles required to break the encryption:
  - p is a strong prime if p-1 and p+1 both have large factors, t and w
  - t-1 and t+1 have a large factor
  - w-1 and w+1 have a large factor
- In our analysis with small key size (10-70 bit) we saw no correlation between key size and cycles required to crack.

•Large Primes

- In our tests, keys greater than 60 bits were not crackable using a cyclic attack in 24 hours
- As RSA currently uses keys that are 1024 bits or higher, it would take many years to crack
- This type of attack is not deemed feasible with current hardware