

## 18: PUBLIC KEY CIPHERS

### 18.1 RSA Ciphers

The Rivest – Shamir – Adelman (RSA) ciphers are our first example of public key ciphers. They are very widely used and rely on the difficulty in factoring products  $N = pq$  of two large primes.

To make an RSA cipher, I first choose two large primes  $p, q$  and set  $N = pq$ . (Typically these are primes with at least 800 binary bits.) Then I choose an exponent  $e$  randomly with  $e$  coprime to  $\varphi(N) = (p - 1)(q - 1)$ . I can always ensure that  $(e, \varphi(N)) = 1$  by taking  $e$  as a prime number larger than both  $p$  and  $q$ . This number  $e$  is called the *encrypting exponent*. Euclid's algorithm allows me to find integers  $d, k$  with

$$de - k\varphi(N) = 1$$

(in polynomial time). The integer  $d$  is called the *decrypting exponent*.

The public key will be  $(N, e)$ . The encrypting function is

$$a \mapsto a^e \pmod{N} .$$

The private key will be  $d$ . The decrypting function is

$$c \mapsto c^d \pmod{N} .$$

The Euler – Fermat theorem shows that

$$(a^e)^d \equiv a^{de} \equiv a^{k\varphi(N)+1} \equiv a \pmod{N} .$$

So the decrypting function is inverse to the encrypting function.

An enemy who intercepts a ciphertext needs to find the plaintext knowing only the public key. This appears to be tantamount to finding the factors  $p$  and  $q$  of  $N$  so that  $\varphi(N)$  can be computed.

**Theorem 18.1**    Security of the RSA ciphers

*Suppose that we have an algorithm to determine the private key for an RSA cipher when we are given the public key. Then the algorithm permits us to factorise products of two distinct primes.*

If we can do this in polynomial time, then we have a very effective way to break RSA ciphers but the theorem shows that this would also give us an equally effective way to factorise. No such algorithm is known.

*Proof:*

We are assuming that we have an algorithm that gives the decrypting exponent  $d$  in terms of  $N$  and  $e$ . Then  $a^{de} \equiv a \pmod{N}$  for every  $a \in \mathbb{Z}_N$ .

Write  $de - 1 = 2^s r$  for some odd integer  $r$ . Let  $\text{ord}_p(x)$  denote the order of an element  $x$  in the group  $\mathbb{Z}_p^\times$ . Set  $X = \{x \in \mathbb{Z}_N^\times : \text{ord}_p(x^r) \neq \text{ord}_q(x^r)\}$ . We will prove various properties of  $X$  that eventually prove our theorem.

## Lemma A

If  $x \in X$ , then we can factorise  $N$ .

*Proof:*

If  $x \in X$ , then  $y = x^r$  satisfies  $y^{2^s} = x^{2^s r} = x^{de-1} = 1$ , so  $\text{ord}_p(y)$  and  $\text{ord}_q(y)$  must be powers of 2. Suppose that  $\text{ord}_p(y) = 2^t < \text{ord}_q(y)$ . Then  $y^{2^t} \equiv 1 \pmod{p}$  but  $y^{2^t} \not\equiv 1 \pmod{q}$ . So

$$(y^{2^t} - 1, N) = p .$$

We could therefore use Euclid's algorithm to find one of the factors of  $N$ .

Thus, to factorise  $N$  when we have  $x \in X$ , we compute  $(y^{2^t} - 1, N)$  for  $t = 0, 1, 2, \dots, s$ . We know that it must be  $p$  for one of these choices and so we obtain a factor of  $N$ . □

Now we wish to count how many elements there are in  $X$ .

### **Lemma B**

*Let  $p$  be an odd prime with  $p - 1 = 2^u v$  for some natural numbers  $u, v$ .  
Then*

$$|\{x \in \mathbb{Z}_p^\times : \text{ord}_p(x^v) = c\}| \leq \frac{1}{2}(p - 1)$$

*for every possible order  $c$ .*

*Proof:*

Let  $\alpha$  be a primitive root modulo  $p$ . So  $\mathbb{Z}_p^\times$  is the cyclic group generated by  $\alpha$ . This certainly implies that  $\alpha^v$  has order  $2^u$  in  $\mathbb{Z}_p^\times$ . An element  $x \in \mathbb{Z}_p^\times$  is equal to  $\alpha^m$  for some natural number  $m$ . So we see that  $\text{ord}_p(x^v) = \text{ord}_p(\alpha^{mv}) = 2^u$  if and only if  $m$  is odd. Thus there are  $\frac{1}{2}(p - 1)$  elements  $x$  with  $\text{ord}_p(x^v) = 2^u$  while the remaining  $\frac{1}{2}(p - 1)$  elements have order  $2^w$  for some  $w < u$ . This means that

$$|\{x \in \mathbb{Z}_p^\times : \text{ord}_p(x^v) = c\}| \leq \frac{1}{2}(p - 1)$$

for every possible order  $c$ . □

## Lemma C

$$|X| \geq \frac{1}{2}(p-1)(q-1) = \frac{1}{2}\varphi(N).$$

*Proof:*

The Chinese remainder theorem Proposition 17.1 shows that  $|X|$  is

$$|\{(x, y) \in \mathbb{Z}_p^\times \times \mathbb{Z}_q^\times : \text{ord}_p(x^r) \neq \text{ord}_q(y^r)\}|.$$

For each  $y \in \mathbb{Z}_q^\times$  we have shown that

$$\{x \in \mathbb{Z}_p^\times : \text{ord}_p(x^r) \neq \text{ord}_q(y^r)\}$$

has at least  $\frac{1}{2}(p-1)$  elements.

Therefore  $|X| \geq \frac{1}{2}(p-1)(q-1) = \frac{1}{2}\varphi(N).$

□

*Proof of Theorem 18.1 (continued)*

Choose an integer  $x$  randomly from  $\mathbb{Z}_N^\times$ . The probability that  $x \in X$  is at least  $\frac{1}{2}$ . When  $x \in X$  we know how to find a factor of  $N$ . When  $x \notin X$ , choose another random value for  $x$ . After  $k$  such random choices we will have found a factor of  $N$  with probability at least  $1 - \left(\frac{1}{2}\right)^k$ .

□

Note that the theorem shows that finding the private key is as hard as factoring  $N$ . There might well be other ways to decipher a particular ciphertext without finding the private key  $d$ . Rivest, Shamir and Adelman conjectured that any algorithm that allows us to decipher messages would also allow us to factorise  $N$ . However, this has not been proved.

## 18.2 Rabin Ciphers

The Rabin cipher is another public key cipher that relies on the difficulty of factoring products  $N = pq$  of two large primes. For this we need to consider finding square roots modulo a prime.

### Lemma 18.2

*Let  $p$  be a prime of the form  $4k - 1$ . If  $c \equiv a^2 \pmod{p}$  then  $a \equiv \pm c^k \pmod{p}$ .*

*Proof:*

If  $c \equiv a^2 \pmod{p}$ , then Fermat's little theorem gives

$$c^{2k} \equiv a^{4k} \equiv a^{(p-1)+2} \equiv a^2 \pmod{p} .$$

So  $c^k \equiv \pm a \pmod{p}$ . □

To make a Rabin cipher I first choose two large primes  $p, q$  of the form  $p = 4k - 1$  and  $q = 4m - 1$ . Set  $N = pq$ . Then I will create a cipher for the alphabet  $\mathbb{Z}_N^\times$ .

The public key will be  $N$  and the encrypting function is

$$a \mapsto a^2 \pmod{N} .$$

(Usually we restrict the alphabet so that  $(a, N) = 1$  and  $a > N^{1/2}$ .) The private key will be  $(p, q)$ .

Suppose that we have received a ciphertext  $c \equiv a^2 \pmod{N}$  and know the private key. Then Lemma 18.2 shows that

$$a \equiv \delta c^k \pmod{p} \quad \text{and} \quad a \equiv \varepsilon c^m \pmod{q}$$

where  $\delta, \varepsilon$  are each  $\pm 1$ . Find integers  $u, v$  with  $up + vq = 1$ . Then the Chinese remainder theorem shows that

$$a \equiv \delta c^k + up(\varepsilon c^m - \delta c^k) \pmod{N}.$$

All four of these possible values can occur and are distinct by Proposition 17.3. To decipher  $c$  we find all four square roots modulo  $N$  and choose the one that makes sense. Our messages should contain enough redundancy for only one of the four choices to make sense.

When we do not know the private key, breaking the Rabin cipher is as hard as factorising  $N$ .

**Theorem 18.3** Security of Rabin ciphers

*An algorithm to decipher the Rabin cipher gives an algorithm to factorise  $N$ .*

*Proof:*

An algorithm to decipher the Rabin cipher must give one of the four square roots of a ciphertext  $c$  modulo  $N$  which we obtain as  $c \equiv x^2 \pmod{N}$ .

Choose  $a \in \mathbb{Z}_N^\times$  at random. This algorithm gives a particular square root  $x$  of  $c \equiv a^2 \pmod{N}$ , so  $x^2 \equiv a^2 \pmod{N}$ . There are 4 distinct choices for  $a$  that give the same value for  $c$ . Two of these give  $x \equiv \pm a$  but the other two do not. For these other two we must have

$$x^2 - a^2 \equiv (x + a)(x - a) \equiv 0 \pmod{N}$$

but neither  $x + a$  nor  $x - a$  is divisible by  $N$ . Therefore,  $(N, x - a) \neq 1$ .

This means that, with probability  $\frac{1}{2}$ , we find a non-trivial factor  $(N, x - a)$  of  $N$ . If we repeat this  $r$  times, the chance of finding a factor of  $N$  is at least  $1 - \left(\frac{1}{2}\right)^r$ . □