Contents

1 Integer Factorisation

Any integer can be expressed as the multiplication of a list of prime numbers. The longer the value, the harder (and slower) this gets. *Semi-primes* (product of two **primes**) are the hardest numbers to factor

2 RSA

- RSA is the most common method for general public key cryptography
- It provides both encryption and/or authentication
- RSA provides us with two keys:
  - Public (e, n). e is usually a small number, d is a much larger number n is a **very large semi-prime number**
  - Private d
- The values e and d are mathematically linked such that:

$$M^e = C \,(\text{mod } n) \tag{1}$$
$$C^d = M \,(\text{mod } n) \tag{2}$$

- They are inverses of one another, when used as exponents mod n

2.1 Euler Totient Function

- Integers a and b are *relatively prime* if they **do not share a divisor** (except 1)
- The Euler totient ϕ is the integers from 1 to n-1 that are relatively prime with n
- The **totient value** of a prime p is simply p-1
- For two primes multiplied together its (p-1)(q-1)

2.2 RSA key generator

- Choose two large primes, p and q, then calculate n = pq
- Select a value e that is relatively prime with the totient of n

$$p = 17, q = 11 \tag{3}$$
$$n = p * q = 187 \tag{4}$$
$$\phi(n) = (p-1) * (q-1) = 160 \tag{5}$$
$$e = one\ of\ 3, 6, 7, 11 = 7 \tag{6}$$

- p, q, $\phi(n)$ - Private
- e, n - Public
- Calculate a multiplicative inverse to e: d, where

$$(e * d) \,\text{mod}\ \phi(n) = 1$$

- This is easily achieved if we know $\phi(n)$, but not otherwise
- Now we have a public key e,n and a private key d

2.3 Why is RSA secure

- We need to know $\phi(n)$ to find d
- Finding this is extremely hard, for example we could factor n into p and q

2.4 Using RSA

- The keys (e, n) and (d) are reversible either can be used for encryption, and the other used for decryption
- Everyone knows the public key, only the owner knows the private key
- This leads us to two very useful use cases for RSA:
  - Encryption only the owner can read
  - Signing that must have been performed by the owner

3 Digital Signatures

- Authentication codes provide integrity, **but dont guarantee the sender**
- Public-key encryption allows us to **sign documents**
- We can use a trusted third party in order to **verify the ownership of a public key**
- Bob then knows he has Alices genuine key, not an imposter
- Can also be self signed
- An important part of Transport Layer security (TLS)

3.1 X.509 v3

- Organised by Public Key Infrastructure (PKI)
- The standard for digital certificates holds information on the type, subject and issuer
- The issuer will usually be a **trusted third party**, like Verisign or Globalsign

3.2 Certificate issuance

- Server (server.com) has a certificate containing their public key, which they want people to trust
- They go to a certificate authority (CA), who after doing ID checks, sign the certificate with their private key
- The server can supply digital signatures using the public key, backed by the certificate when requested, e.g. during a TLS handshake
- To verify the trust in the Server.com certificate, we need to examine the signing certificate
- In many cases, the chain involves multiple certificates
- Chains always end in a root certificate, **located on your machine**

Reference section

placeholder