Multiplication as a one-way function

- given 2 prime numbers p and q, calculate $n = p \cdot q$
  this requires little effort even for large numbers
- given n is the product of 2 prime numbers p and q determine p and q
  this is much more difficult: checking all possible prime numbers and seeing if they divide n (it's really slow for large numbers)
  there's no fast algorithm for this

based on the public and private Key system. It uses different theorems:

Fermat's little theorem

V1: if p is prime then $a^p \equiv a \pmod p$

V2: if p is prime and a is not divisible by p then $a^{p-1} \equiv 1 \pmod p$
you can also use this one to determine the inverse of a number to determine
$a^{p-2} \cdot a \equiv a^{p-1} \equiv 1 \pmod p$.

In RSA we want to take n, product of two prime numbers, Eulero did the necessary generalization

Eeulero's theorem

V1: for $a \quad n \in \mathbb{N} \quad n > 1$ and $a \mid gcd(a, n) = 1 \Rightarrow a^{\varphi(n)} \equiv 1 \pmod n$
$\varphi(n) = \| \{ x \in \mathbb{N} \mid x < n, gcd(x, n) = 1 \} \|$

V2: $n \in \mathbb{N} \mid n = pq$ p, q prime, $p \nmid a, q \nmid a$ then:
$a^{(p-1)(q-1)} \equiv 1 \pmod n$

RSA

• Generating a public and private Key:

- Choose two prime numbers p and q
- Determine $n = p \cdot q$ and $m = (p-1)(q-1)$
- Choose e so that $1 < e < m$ and $\gcd(e, m) = 1$
- Determine $d = e^{-1} \pmod m$
- We don't need p and q anymore
- (n, e) is the public key
- d is the private key

• Encipher a message

An wants to send B to Bert, he gives An his public Key (n, e)
then the encrypted message is $C = B^e \pmod n$
Bert recieves the message C. If $c > n$ An has to divide the message in pieces.
the size of the pieces is called block size

• Decipher a message

Bert recieves C and can decipher it with d, he's the only one that has d
he recalculates $B = C^d \pmod n$ (possibly for every piece of c)

• Why does it work?

$(B^e)^d \pmod n = B^{ed} \pmod n$ (since $d = e^{-1}$, so ed can be written as a multiple of m, $+1$)

C^d

$= B^{1+km} \pmod n$ (because $ed \equiv 1 \pmod m$)

$= B \cdot (B^m)^k \pmod n$

$= B \cdot 1^k \pmod n$ (because of Eulero's theorem)

$= B \pmod n$

the power of RSA

we use both one-way functions we have seen previously

- RSA can be cracked if p and q can be easily found from n, because then m is easy to determine and with m d is also easy to find.
That is why it's important to have p and q big prime numbers (f.e. 200 digits each) in that case it becomes really hard.
- from $C = B^e \pmod n$, given C, e, n directly determening B is impossible since it is precisely the discrete e-th root of the power, for wich we mentioned there is no fast algorithm for a large n.