
RSA
ALGORITHM

Криптографический алгоритм ассимитричного шифрования

RSA

Криптографический алгоритм ассимитричного шифрования

RSA

Функция Эйлера

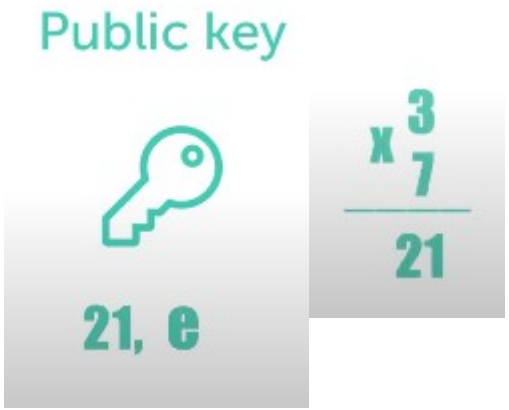
Ф = {P – 1} * {Q – 1} = {3 – 1} * {7 – 1} = 12

e:

1. Простое число ;

2. < ф [12] 2 3 5 7 11

3. Взаимно простое с [Ф]



- P

- Q

- mod

Криптографический алгоритм ассимитричного шифрования

{e, mod} = {5, 21}

RSA

Ф = 12

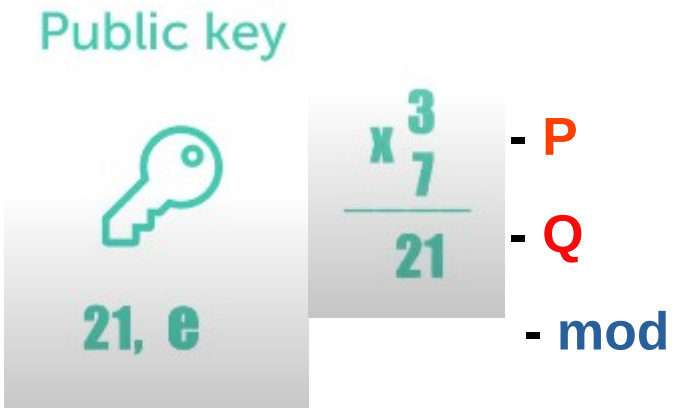
Функция Эйлера

Ф = {P – 1} * {Q – 1} = {3 – 1} * {7 – 1} = 12

e = 5:

1. Простое число ;

{e, mod} – Открытый ключ



- P

- Q

- mod

Криптографический алгоритм ассимитричного шифрования

Открытй ключ **RSA** **Закрытый ключ**

{e,mod}={5,21} **{d,mod}={17,21}**

X < mod{21} **2**

11 **2 в 17**

11 в 5 **131 072**

161 051 **131 072 % 21 = 11**

161 051 % 21 = 2 **131 072 – 6241 * 21 =11**

161 051 - 7669 * 21 = 2

Криптографический алгоритм ассимитричного шифрования

220

220 / 2

110 / 2

55 / 5

11 – простое число

221

На 2 не делится

На 3 не делится

На 5 не делится

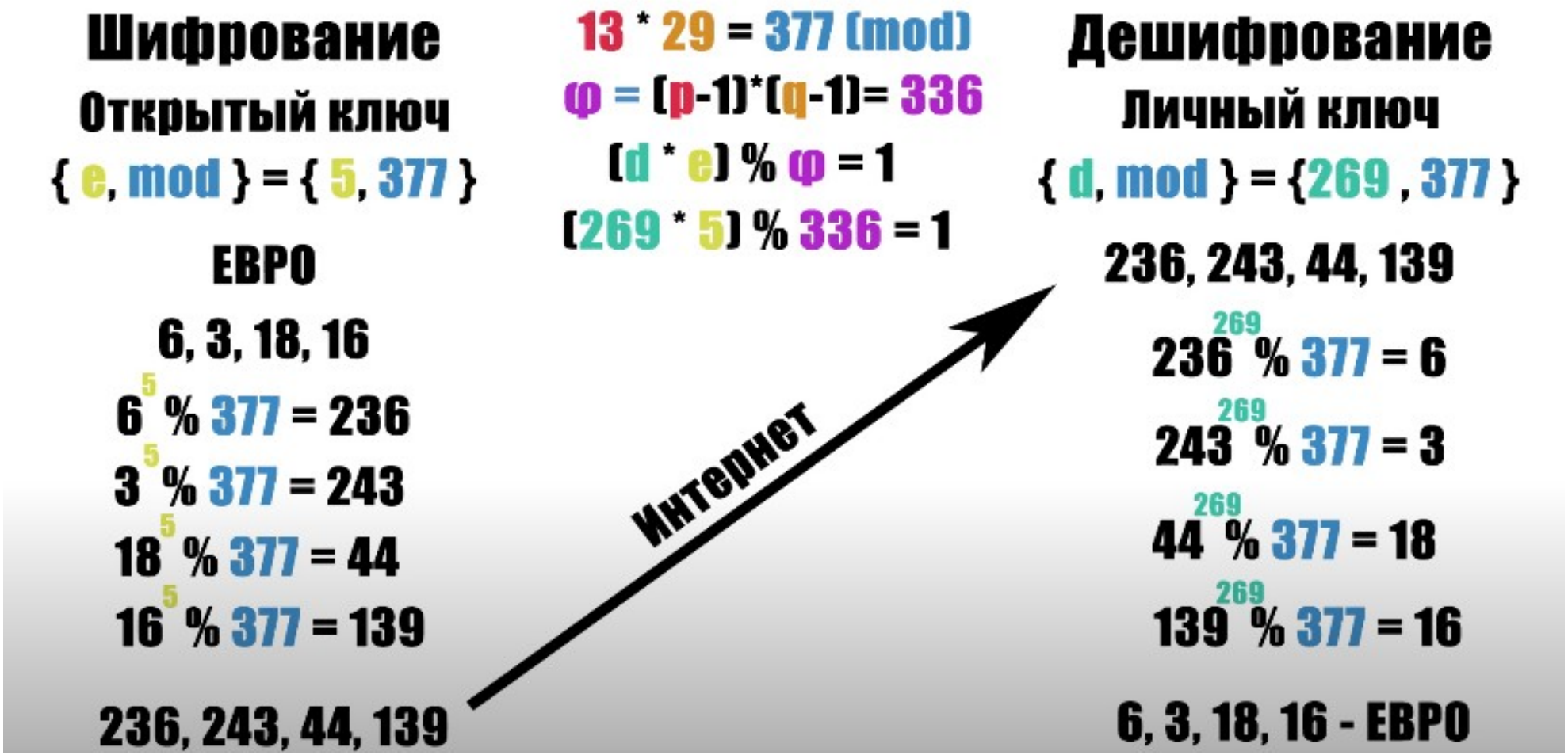
На 7 не делится

На 11 не делится

Лишь 13 дает ответ: 221 / 13 = 17

Просты множители 13 * 17(mod)

Криптографический алгоритм ассимитричного шифрования



Криптографический алгоритм ассимитричного шифрования

Проблемы RSA шифрование по буквам:

1. Частые коды пробелов позволяют разбить шифровку на слова

2. Легко вычеслить коды однобуквенных слов [а,в,р,л]

3. Недолгие пробелы вычесляются дополнительные буквы по коротким словам , типа “но”, “не”, “да”

4.По более длинным словам можно без труда восстоновить оставшиеся буквы.

5. Поэтому злоумышленнику не придется отгадывать ваши секретные ключи. Он вломает сообщение не зная их.

Криптографический алгоритм ассимитричного шифрования

Дополнительные алгоритмы шифрование

Каждая предыдущая часть сообщения начинает влиять на следующую

b:=[b+a]%mod

a – предыдущая часть сообщени; b-следующая

6.3.18.16

6-без изминений, т.к. (6+0)%377 = 6

(3+6)%377=9

(18+9)%377 = 27

(16+27)%377 = 43

6,9,27,46

b:=(b-a)%mod

Криптографический алгоритм ассимитричного шифрования