

RSA

Cara kerja RSA :

1. Plain text (m) diubah menjadi bilangan bulat. Contoh :

m =

14373027140228241128488804738269643284877678466106311986085937
91957150817173014345647155343443375122070128550270138189134701
7436876486280948643039280253309

2. Untuk dikirim ke penerima, plaintext (m) harus dienkripsi (enkripsi RSA) menggunakan rumus : Cipher Text (C) = $m^e \bmod N$. Kalo di program seperti ini :

$C = \text{pow}(m, e, N)$

3. Kalo udah ke enkripsi, teks bisa langsung dikirim ke penerima

4. Jika penerima ingin membaca apa yang diterima, penerima harus melakukan decrypt / dekripsi ciphertext (C) nya dengan rumus : Plain Text (m) = $C^d \bmod N$. Kalo di program seperti ini :

$m = \text{pow}(C, d, N)$

5. Untuk mendapatkan d sebagai bahan xor Cipher Text (C) dapat dicari melalui rumus : $e \times d = 1 \bmod \phi_N$. Kalo di program buat nyari e atau d seperti ini :

$e = \text{pow}(d, -1, \phi_N)$

$d = \text{pow}(e, -1, \phi_N)$

6. Remind! N biasanya ganjil karena p dan q adalah prima lebih dari 2.

$N = p \times q$

7. Remind! $\phi_N = (p-1)(q-1)$

Contoh di PicoCTF - EVEN RSA CAN BE BROKEN

Soal :

```
~ % nc verbal-sleep.picoctf.net 53886
N: 23196540697482903028228567024613165150645457570416881572804709495819078328262
199796002561457836714868702633453549271711903229111508165781018217773437785726
e: 65537
ciphertext: 14373027140228241128488804738269643284877678466106311986085937919571
50817173014345647155343443375122070128550270138189134701743687648628094864303928
0253309
```

N =

2319654069748290302822856702461316515064545757041688157280470949581
90783282621997960025614578367148687026334535492717119032291115081657
81018217773437785726

e : 65537

cyphertext (C) =

1437302714022824112848880473826964328487767846610631198608593791957
1508171730143456471553434433751220701285502701381891347017436876486
280948643039280253309

Solusi dalam program python :

1. Nyari N berdasarkan rumus $N = p \times q \rightarrow N$ (genap) bisa diasumsikan antara p dan q salah satunya adalah prima genap yaitu 2 karena 2 adalah satu-satunya prima yang genap

Jika $p = 2$, maka q :

$$N = p \times q$$

$$N = 2 \times q$$

$$q = N / 2$$

```
>>> q = N // 2
```

q ini tujuannya untuk mencari nilai phi_N.

2. Kita cari phi_N nya dulu dengan rumus $\phi_N = (p-1)(q-1)$

$$\phi_N = (2-1)(q-1)$$

$$\phi_N = 1(q-1)$$

$$\phi_N = q-1$$

```
>>> phi_N = (2-1)(q-1)
```

3. nyari e atau d (karena e sudah ada di soal yang konteksnya adalah soal memberikan enkripsi, maka kita cari d untuk mendekripsi chipertext)

$$e \times d = 1 \bmod \phi_N$$

$$d = \text{pow}(e, -1, \phi_N)$$

```
d = pow(e, -1, phi_N)
```

4. Kita bisa lakukan dekripsi dengan menghasilkan sebuah variabel plaintext (m) dengan rumus $\text{PlainText (m)} = C^e \bmod N$

`m = pow(C, d, N)`

```
m = pow(c, d, N)
```

5. Karena udah dapet hasil dari variabel m yaitu plaintext tinggal bisa di print.

```
m = pow(c, d, N)
flag = long_to_bytes(m)
print(flag)
```

`b'picoCTF{tw0_1$_pr!m3de643ad5}'`

Info : `long_to_bytes()` berfungsi mengkonversi ASCII ke string.