REAKING RSA

EXP.NO: 4

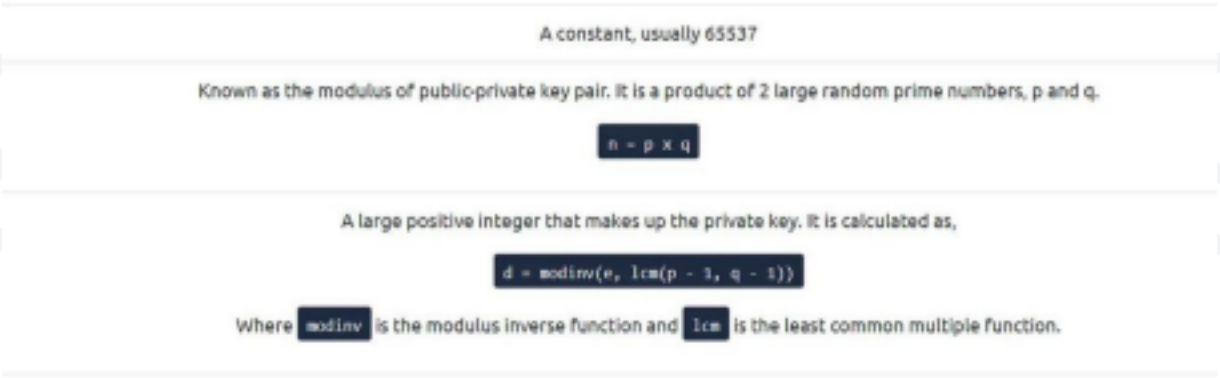
AIM:

Breaking RSA in TryHackMe Using Fermat's Factorization Algorithm

The goal is to break an RSA encryption challenge in TryHackMe by factoring the modulus N using Fermat's Factorization Algorithm. This method works best when the two prime factors p and q are close to each other, meaning their difference is small. Once p and q are found, the private key and decrypt messages can be found.

A brief overview of RSA

The security of RSA relies on the practical difficulty of factoring the product of two large prime numbers, the "factoring problem". RSA key pair is generated using 3 large positive integers –



(e, n) are public variables and make up the public key. d is the private key and is calculated using p and q. If we could somehow factorize n into p and q, we could then be able to calculate d and break RSA. However, factorizing a large number is very difficult and would take some unrealistic amount of time to do so, provided the two prime numbers are **randomly** chosen.

Fermat's Factorization Algorithm Mathematical Basis:

RSA uses a modulus N calculated as:

$N=p\times q$

N = p x q

where p and q are prime numbers.

If p and q are close, they can be rewritten as:

$p=(a-b)$, $q= (a+b)$

where a is the midpoint between p and q, and b is the offset.

Rearranging, we get:

$N=(a-b)(a+b)=a^2-b^2$

which can be rewritten as:

$a^2-N=b^2$

Thus, the problem reduces to finding an integer a such that a^2-N is a perfect square.

ALGORITHM:

1. **Find an initial estimate of aa:**

$$\text{aa} = [\sqrt{\text{NN}}]$$

(Round up the square root of NN).

2. **Iterate until a^2-N is a perfect square:**
   - Compute $b^2=a^2-N$
   - Check if $b^2$is a perfect square.
   - If it is, set $\text{aa} = \sqrt{\text{aa}^2}$
   - Compute p=a−b and q=a+b.

3. **Verify p and q by checking if p×q=N**

4. **Use p and q to compute φ(N) and the private key d:**

$$\varphi(N)=(p-1)(q-1)$$

$$d=e^{-1} \bmod \varphi(N)$$

using the Extended Euclidean Algorithm.

5. **Decrypt the ciphertext using:**

$$M=C^d \bmod N$$

When Fermat's Factorization Works Well:

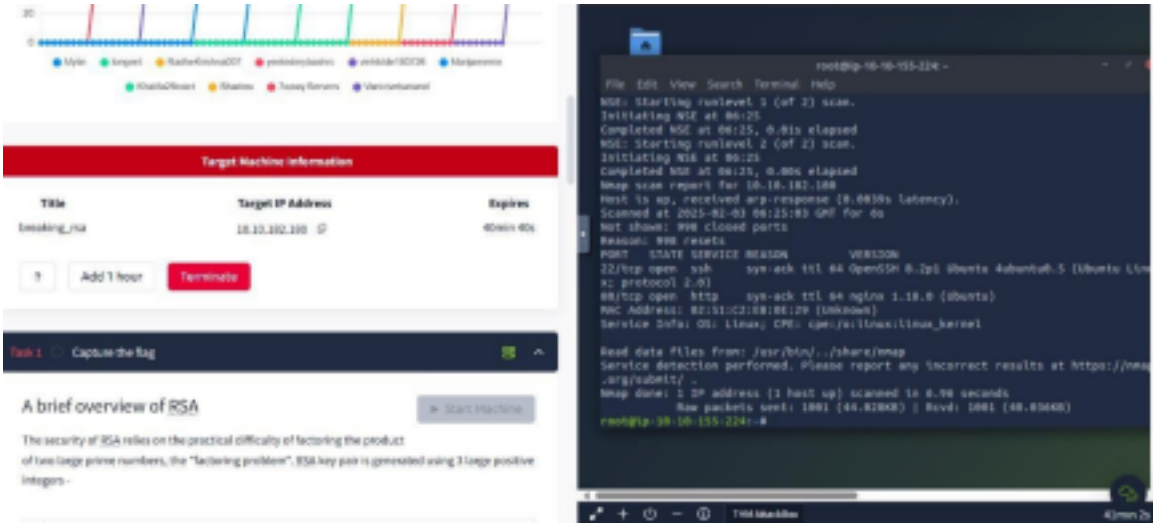
- **When p and q are close.**
- **For small or medium-sized RSA moduli.**
- **When the difference q - p is small, making b small.**

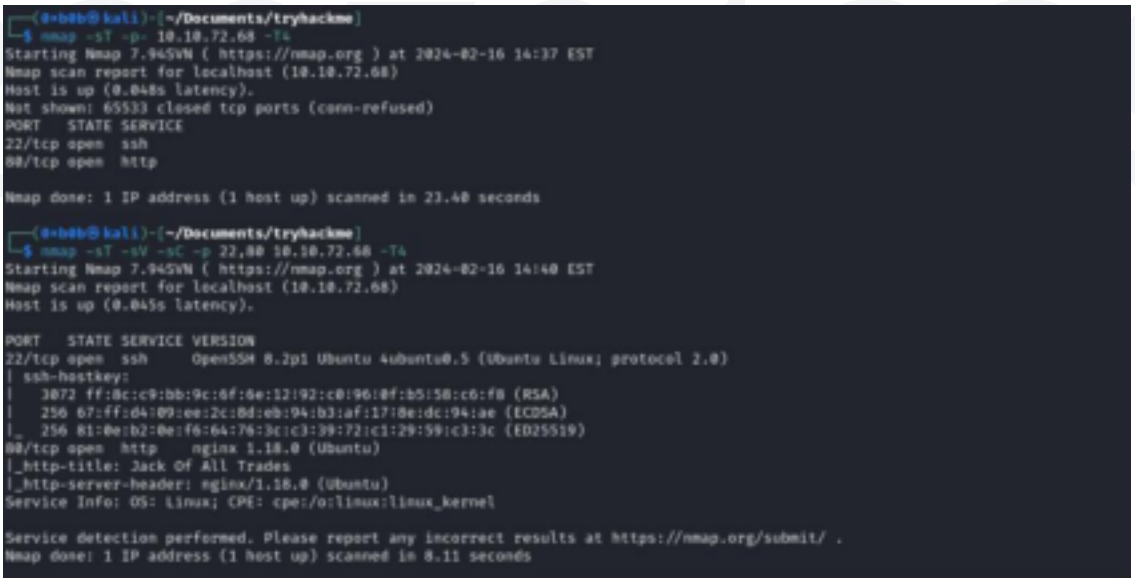
OUTPUT:

1. How many services are running on the box?

$ sudo nmap -sV -Pn -vvv -T3 10.10.182.180

Ans: 2





Q. 2 What is the name of the hidden directory on the web server? (without leading '/')

Ans: development

```
┌──(0×b0b㉿kali)-[~]
└─$ gobuster dir -u http://10.10.72.68 -w /usr/share/wordlists/dirb/big.txt
===============================================================
Gobuster v3.6
by OJ Reeves (@TheColonial) & Christian Mehlmauer (@firefart)
===============================================================
[+] Url:                     http://10.10.72.68
[+] Method:                  GET
[+] Threads:                 10
[+] Wordlist:                /usr/share/wordlists/dirb/big.txt
[+] Negative Status codes:   404
[+] User Agent:              gobuster/3.6
[+] Timeout:                 10s
===============================================================
Starting gobuster in directory enumeration mode
===============================================================
/development          (Status: 301) [Size: 178] [--> http://10.10.72.68/development/]
Progress: 20469 / 20470 (100.00%)
===============================================================
Finished
===============================================================
```

Q.3 What is the length of the discovered RSA key? (in bits)

To determine the length in bits of the public we can issue the following command:

```
┌──(0×b0b㉿kali)-[~/Documents/tryhackme/breaking-rsa]
└─$ ssh-keygen -l -f id_rsa.pub
2048 SHA256:DIqTDIhboydTh2QU6i58JP+5aDRnLBPT8GwVun1n0Co no comment (RSA)
```

Ans: 4096

Q.4 What are the last 10 digits of n? (where 'n' is the modulus for the public-private key pair) Ans: 1225222383

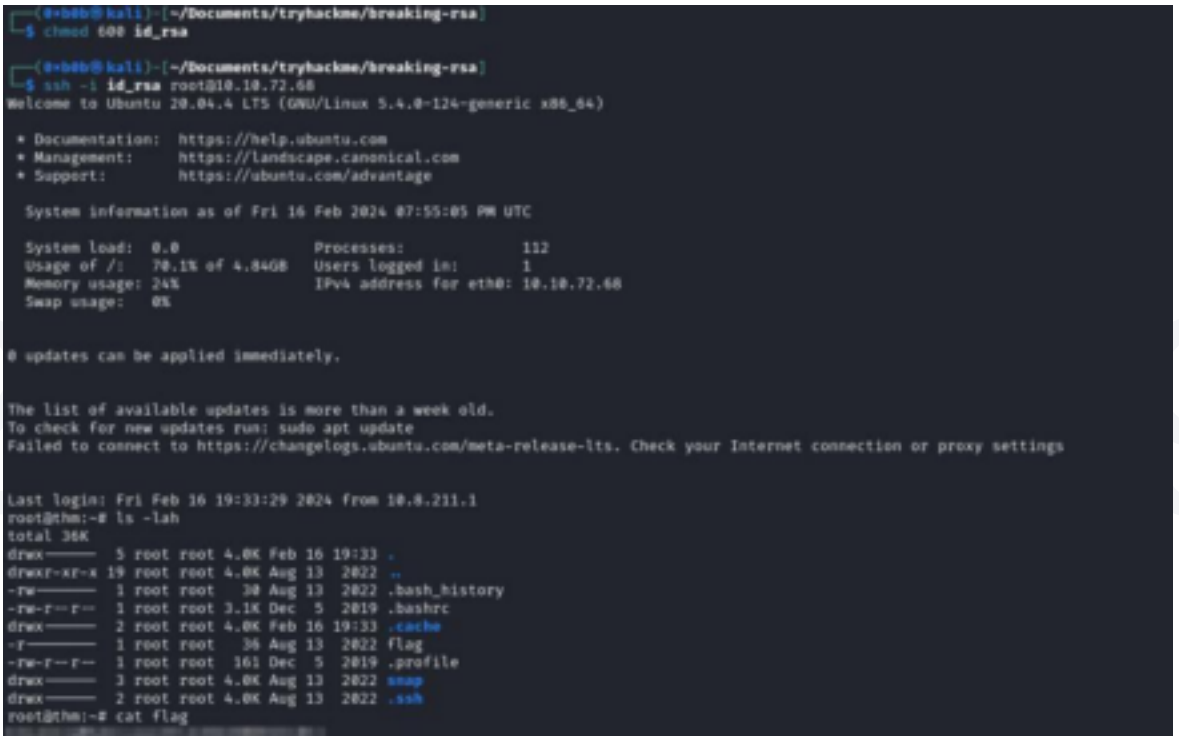
```
┌──(0×b0b㉿kali)-[~/Downloads]
└─$ python
Python 3.11.7 (main, Dec  8 2023, 14:22:46) [GCC 13.2.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> from Crypto.PublicKey import RSA
>>> f = open("id_rsa.pub","r")
>>> key = RSA.importkey(f.read())
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
AttributeError: module 'Crypto.PublicKey.RSA' has no attribute 'importkey'. Did you mean: 'importKey'?
>>> key = RSA.importKey(f.read())
>>> print key.n
  File "<stdin>", line 1
    print key.n
          ^^^^^^^^^^
SyntaxError: Missing parentheses in call to 'print'. Did you mean print( ... )?
>>> print(key.n)
[redacted]
>>> print(key.e)
65537
```

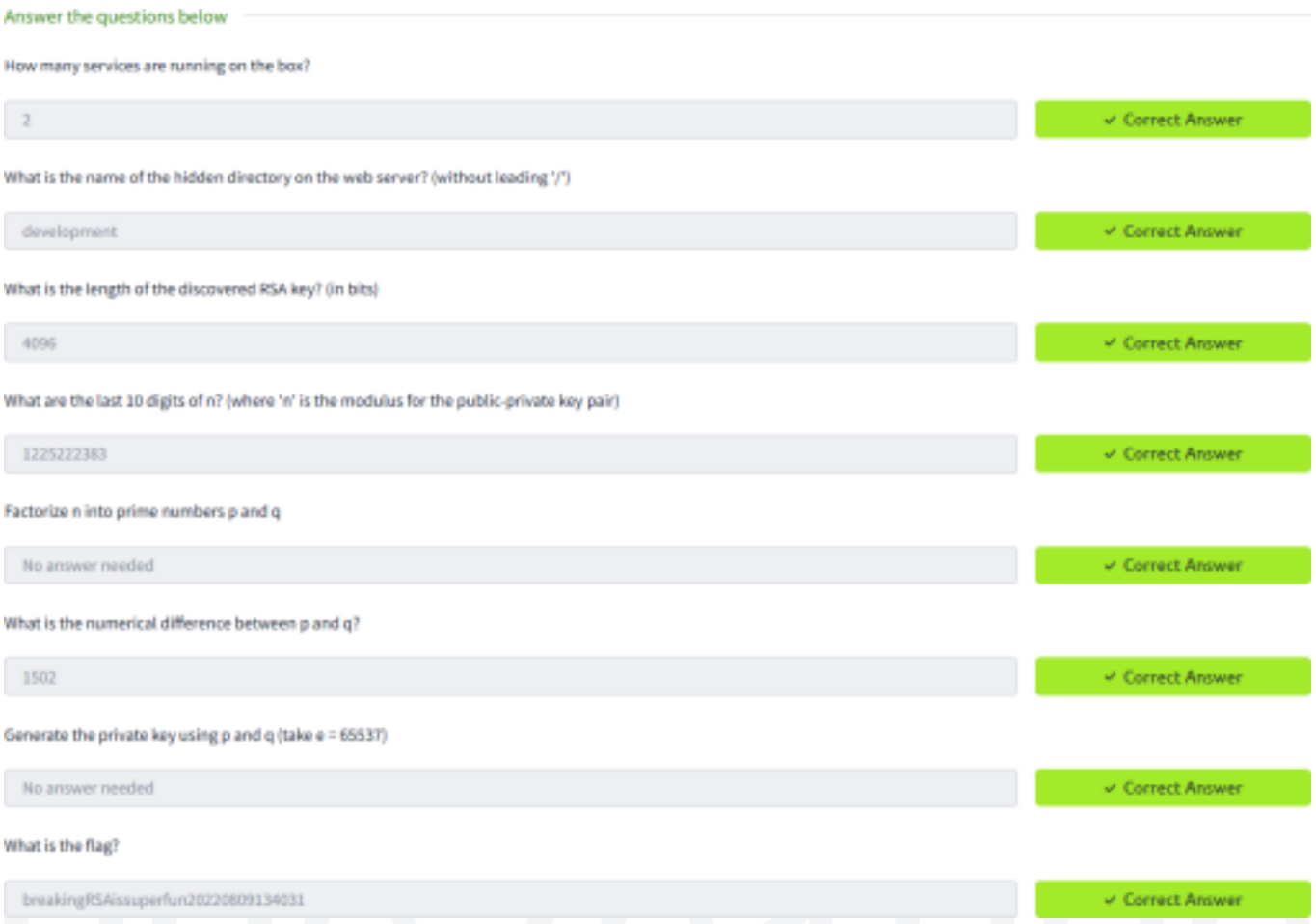

Q.5 What is the numerical difference between p and q?

Ans: 1502

Q.6 What is the flag?

Ans: breakingRSAissuperfun20220809134031





RESULT:

Thus, Breaking RSA in TryHackMe is Completed Successfully