Experiment-2

Cryptanalysis of RSA

Date: 21-08-24

AIM

Implementation of Cryptanalysis using RSA.

PROCEDURE

Step-1: Install VMWare and host Kali Linux.

Step-2: Login to Kali Linux and open Terminal and run commands.

Step-3: Use Hexadecimal to decimal convertor.

Step-4: Use factordb.com to find the factors for the decimal value.

Step-5: Write an exploit in python and get the plain text.

SOURCECODE

*open vmware, go to kali open terminal.
*For opl moving directory to desktop
-(kali㉿kali)-[~]
cd Desktop
* In Desktop create folder rsa. copy thege 3 files enc.txt exploit.py pubke -y.pem in folder.

*→ cd rsa
→ (kali㉿kali)-[~/Desktop/rsa]
19
enc.txt exploit.py pubkey.pem

*open enc.txt file

$ cat enc.txt

[illegible]

→ open pubkey.pem

$cat pubkey.pem

—BEGIN PUBLIC KEY—

MGQWDBYJkoZIhvCNAQEBBQADUWAWUAJJAMA
3k/b+So2EMjj8Ro47t5FdLO6WHMMvWUPOIZOli
-Pu638kF8/QjRaOaJGMFHR1MTnG5J9v5/JzV
UjHTB3/uNJMovyyOOZQOWIDAQAB

— END PUBLIC KEY—

*The Below command is used to display the details of public RSA key stored in PEM File Format.

$ openssl rsa -pubin -inform PEM -text -noout < pubkey.pem

RSA Public-key: (576 bit)

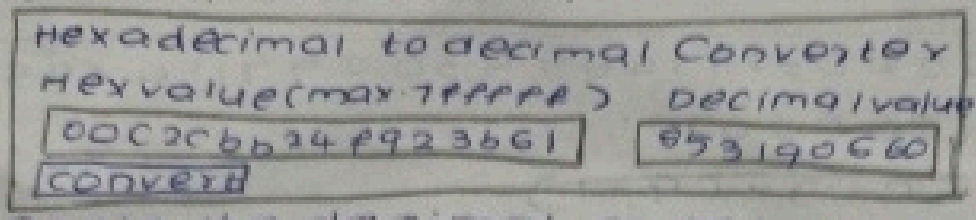
Modulus:

00:c2:cb:b2:4f:db:f9:23:b6:1a:68:c3:f1:1a:38:
96:de:45:74:b3:ba:58:73:0c:bd:65:09:38:86:4e:
22:23:ee:06:70:4a:17:cf:d0:ed:16:b4:68:91:a6:
14:74:75:99:39:c6:e4:99:af:e7:f2:59:55:48:c7:
40:1a:7f:b8:d2:4c:d1:5c:b2:36:4c:d0:a3

Exponent: 65537 (0x10001)

*copy the hexadecimal decimal code into a notepad as n value as it is a hexadecimal we can convert it into decimal for gaining the plaintext.

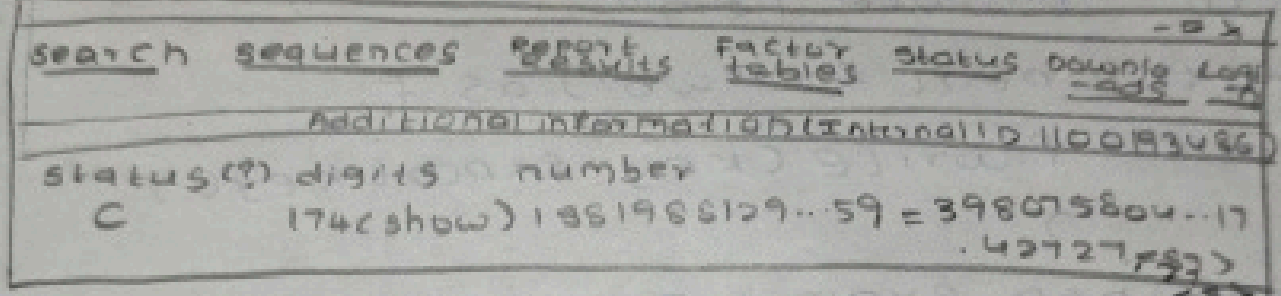
Hexadecimal to decimal converter



Paste the decimal code in the notepad as n value.

→ need to factorize n.

→ So goto website factordb.com click search, poste decimal value of n.



*Create a exploit.py

$ touch exploit.py

→ To install pycryptodome:

$ pip install pycryptodome

* copy the code in the exploit.py file and paste it

n = 188198812920607963869723946
16504398071635633794173827007
633564229888594152

e = 65537

p = 3980750864240649373971255
00556864911990643623452
67084068518975794638895
61

q = 4727721461074353025362307

OUTPUT

```
19730482246329146953029711 4645
9852171130520711
phi - n = (p-1)*(q-1)
d = inverse (e, phi-n)
key = RSA.construct (n, e, p, d, q))
fn = "private.pem"
with open( fn, "wb") as f:
        f.write (key.exportkey()
        )
```

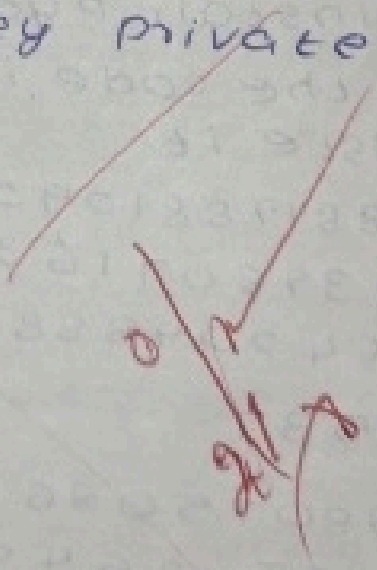
→ python exploit.py

→ to decrypt the text

openssl pkeyutl -decrypt -in enc.txt -out dec.txt -inkey private.pem

→ $ cat dec.txt

RSA is easy.



Experiment-2

Cryptanalysis of RSA

Date: 13 - 8 - 24

AIM

Implementation of Cryptanalysis using RSA.

PROCEDURE

Step-1: Install VMWare and host Kali Linux.

Step-2: Login to Kali Linux and open Terminal and run commands.

Step-3: Use Hexadecimal to decimal convertor.

Step-4: Use factordb.com to find the factors for the decimal value.

Step-5: Write an exploit in python and get the plain text.

SOURCECODE

Steps

1. create a folder name 'rsa' in kali linux desktop
2. open browser in github download three files. enc.txt exploit.py pubkey.pem
3. Copy those three files and paste in rsa folder.

4. Now open cmd in kali and type commands

→ $ cd Desktop

→ $ cd rsa

→ $ ls

It lists all the files as follows.

enc.txt exploit.py pubkey.pem

→ $ cat pubkey.pem

It is used to access the content of the file.

→ $ openssl rsa -pubin -inform PEM -text -noout < pubkey.pem

→ $ touch exploit.py

→ $ pip install pycryptodome

→ $ python exploit.py

→ $ openssl pkeyutl -decrypt -in enc.txt -out dec.txt -inkey private.pem

→ $ cat dec.txt

It displays output as

RSAisEasy