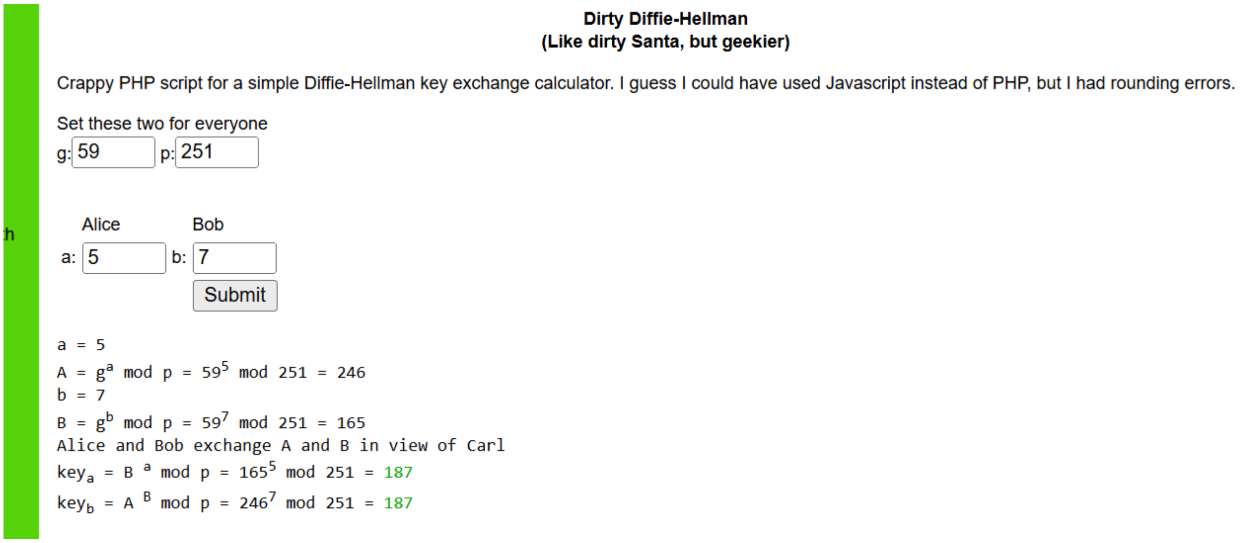
ASYMMETRIC CRYPTO

- Diffie - hellman key exchange (DHKE) - 1967 by Diffie and Hellman
  - Method of securely exchanging keys over a public channel with no prior knowledge of each other
  - Key can be used to encrypt subsequent communications using symmetric key cipher

- RSA (Rivest Shamir Adleman)
  - Generates two large random prime numbers, uses them to generate public + private key pairs for encryption, decryption, digital signature generation and digital signature verification

Task 1 - Deffie-hellman key exchange



Dirty Diffie-Hellman
(Like dirty Santa, but geekier)

Crappy PHP script for a simple Diffie-Hellman key exchange calculator. I guess I could have used Javascript instead of PHP, but I had rounding errors.

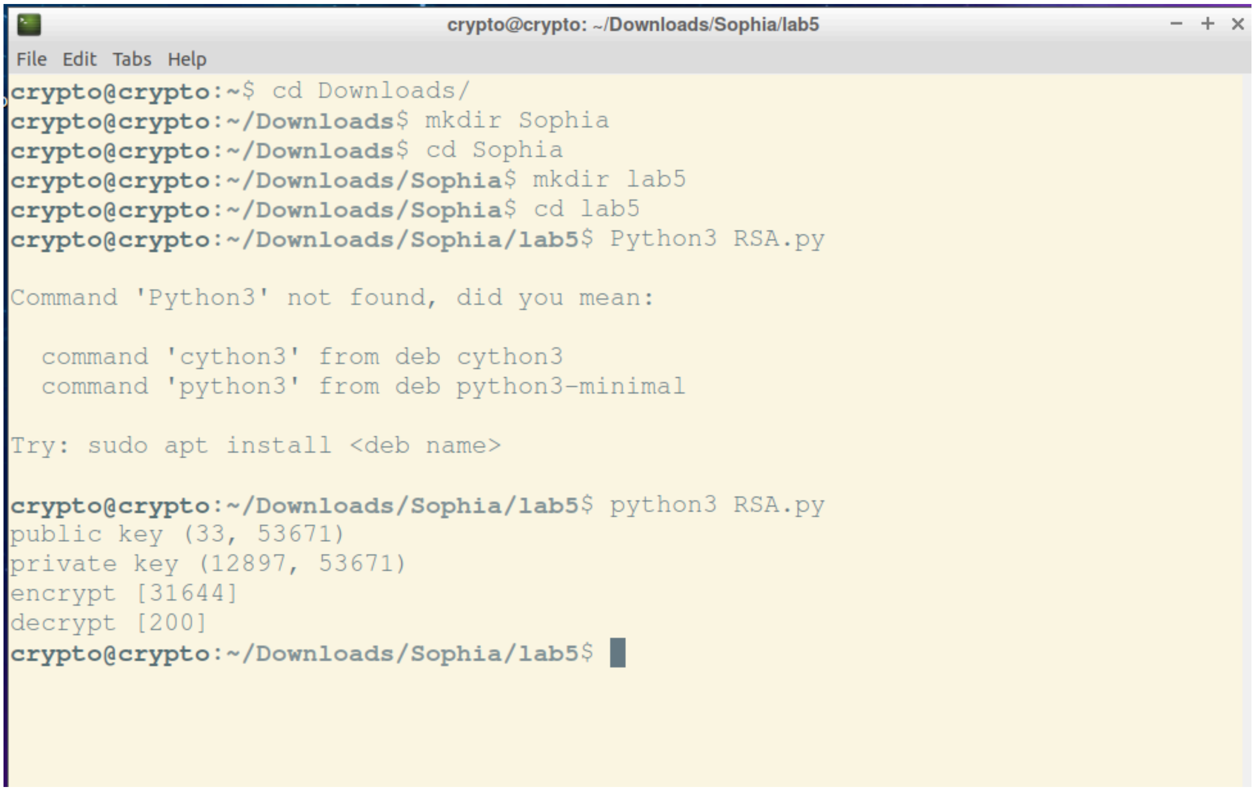
Set these two for everyone
g: 59 p: 251

Alice Bob
a: 5 b: 7
Submit

a = 5
A = g^a mod p = 59^5 mod 251 = 246
b = 7
B = g^b mod p = 59^7 mod 251 = 165
Alice and Bob exchange A and B in view of Carl
key_a = B ^{a} mod p = 165^5 mod 251 = 187
key_b = A ^{B} mod p = 246^7 mod 251 = 187

Task 2 - Manual RSA encryption and decryption

a. Deriving the private key



```
crypto@crypto:~$ cd Downloads/
crypto@crypto:~/Downloads$ mkdir Sophia
crypto@crypto:~/Downloads$ cd Sophia
crypto@crypto:~/Downloads/Sophia$ mkdir lab5
crypto@crypto:~/Downloads/Sophia$ cd lab5
crypto@crypto:~/Downloads/Sophia/lab5$ Python3 RSA.py

Command 'Python3' not found, did you mean:

  command 'cython3' from deb cython3
  command 'python3' from deb python3-minimal

Try: sudo apt install <deb name>

crypto@crypto:~/Downloads/Sophia/lab5$ python3 RSA.py
public key (33, 53671)
private key (12897, 53671)
encrypt [31644]
decrypt [200]
crypto@crypto:~/Downloads/Sophia/lab5$ 
```

Q: What are the numerical values of the private key and the public key?
public key (33, 53671)
private key (12897, 53671)

b. Encrypting a message

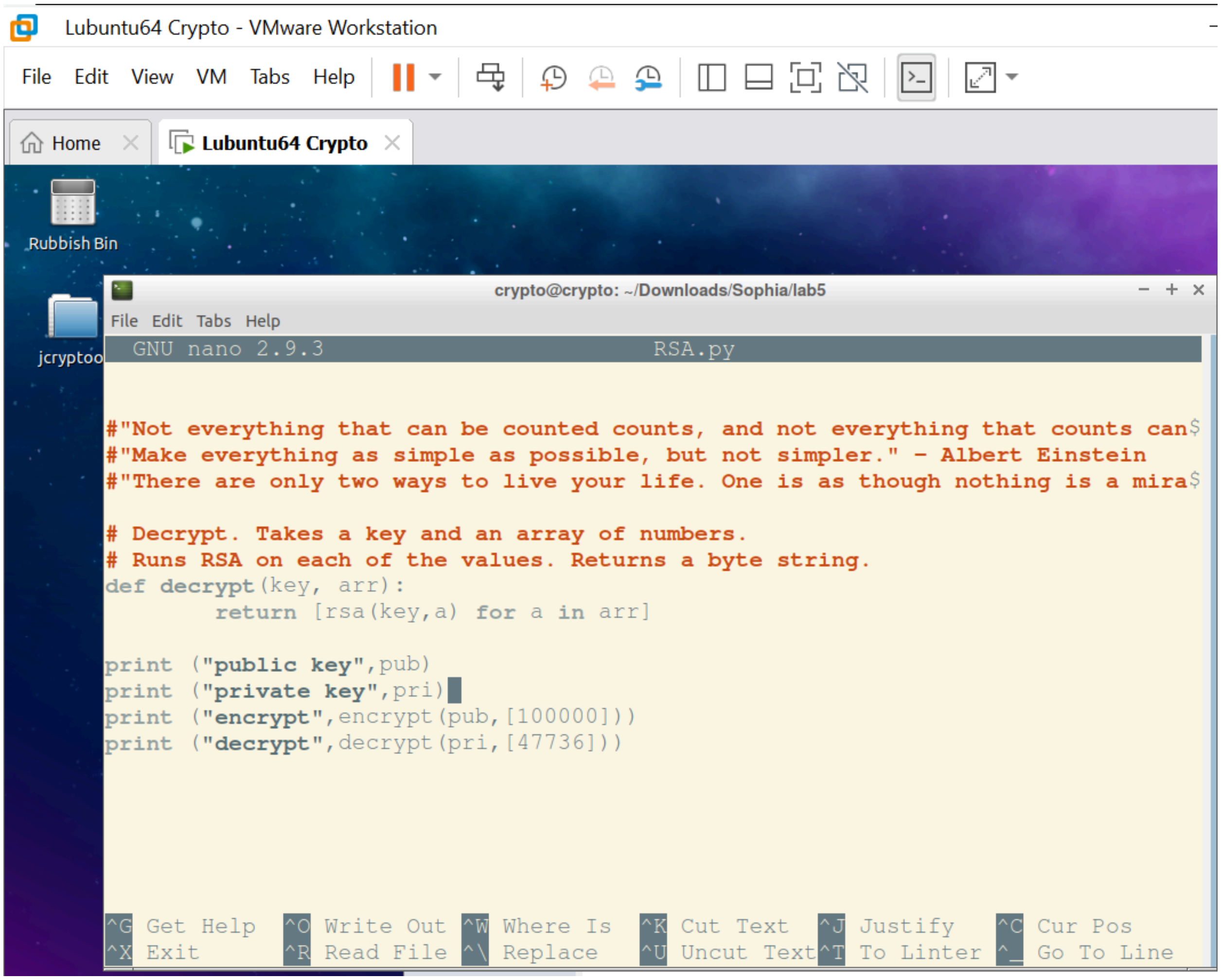
Q: What is the cipher-text of “100”?
encrypt [31644]

c. Decrypting a message

Q:What is the plain-text of C?
decrypt [200]

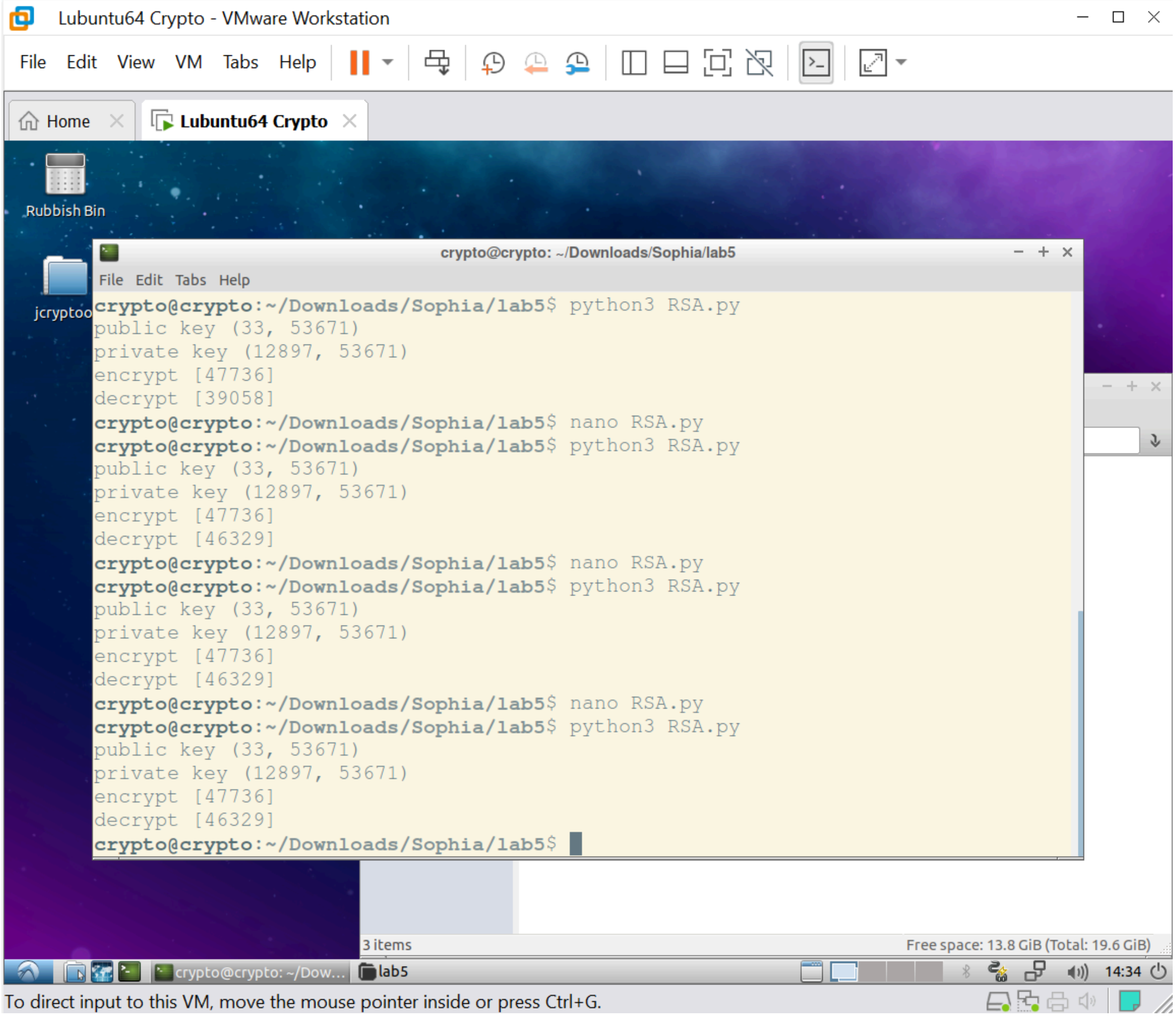
Q: Try to encrypt a big number, e.g., 100000, and then decrypt the cipher-text. Describe your finding and justify it. What is the biggest number that can be correctly encrypted and then decrypted?

- Maximum depends on product between the two prime numbers - 191 x 281 = 53671
  - 53671 is the limit for correctly encrypting and decrypting

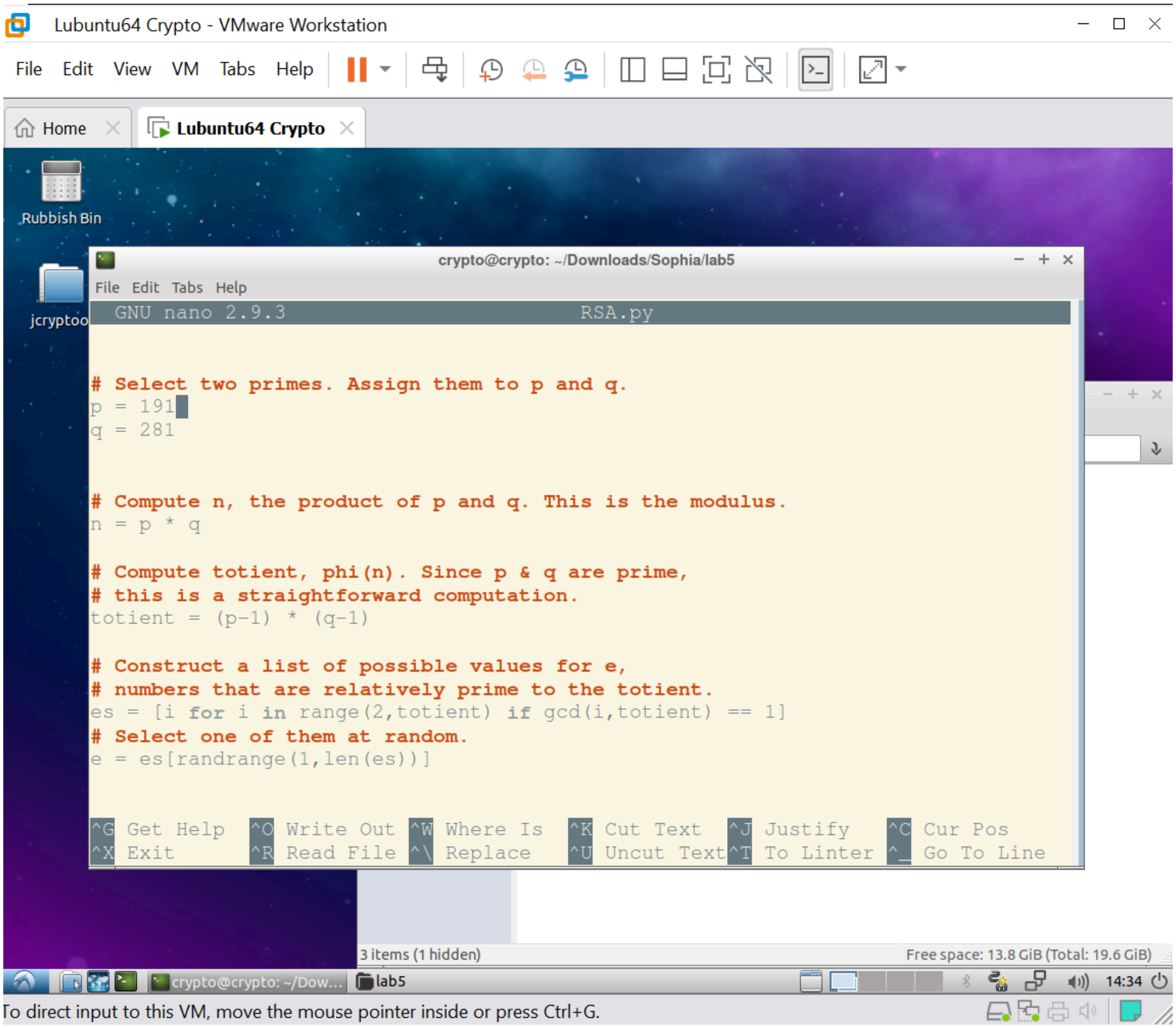


Lubuntu64 Crypto - VMware Workstation
File Edit View VM Tabs Help
Home
Lubuntu64 Crypto
Rubbish Bin
jcryptoo
crypto@crypto: ~/Downloads/Sophia/lab5
File Edit Tabs Help

```
  GNU nano 2.9.3                         RSA.py


#"Not everything that can be counted counts, and not everything that counts can$
#"Make everything as simple as possible, but not simpler." - Albert Einstein
#"There are only two ways to live your life. One is as though nothing is a mira$

# Decrypt. Takes a key and an array of numbers.
# Runs RSA on each of the values. Returns a byte string.
def decrypt(key, arr):
        return [rsa(key,a) for a in arr]

print ("public key",pub)
print ("private key",pri)
print ("encrypt",encrypt(pub,[100000]))
print ("decrypt",decrypt(pri,[47736]))



^G Get Help  ^O Write Out ^W Where Is  ^K Cut Text  ^J Justify   ^C Cur Pos
^X Exit      ^R Read File ^\ Replace   ^U Uncut Text^T To Linter ^_ Go To Line
```


Lubuntu64 Crypto - VMware Workstation
File Edit View VM Tabs Help
Home
Lubuntu64 Crypto
Rubbish Bin
jcryptoo
crypto@crypto: ~/Downloads/Sophia/lab5
File Edit Tabs Help
3 items
Free space: 13.8 GiB (Total: 19.6 GiB)
crypto@crypto: ~/Dow...
lab5
14:34
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

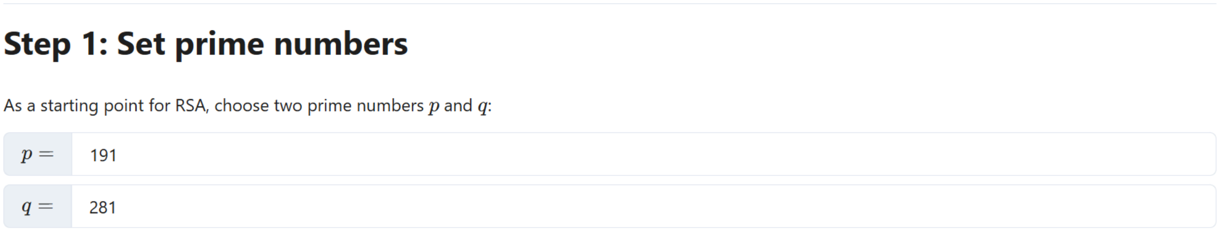
```
crypto@crypto:~/Downloads/Sophia/lab5$ python3 RSA.py
public key (33, 53671)
private key (12897, 53671)
encrypt [47736]
decrypt [39058]
crypto@crypto:~/Downloads/Sophia/lab5$ nano RSA.py
crypto@crypto:~/Downloads/Sophia/lab5$ python3 RSA.py
public key (33, 53671)
private key (12897, 53671)
encrypt [47736]
decrypt [46329]
crypto@crypto:~/Downloads/Sophia/lab5$ nano RSA.py
crypto@crypto:~/Downloads/Sophia/lab5$ python3 RSA.py
public key (33, 53671)
private key (12897, 53671)
encrypt [47736]
decrypt [46329]
crypto@crypto:~/Downloads/Sophia/lab5$ nano RSA.py
crypto@crypto:~/Downloads/Sophia/lab5$ python3 RSA.py
public key (33, 53671)
private key (12897, 53671)
encrypt [47736]
decrypt [46329]
crypto@crypto:~/Downloads/Sophia/lab5$
```



```
  GNU nano 2.9.3                          RSA.py


# Select two primes. Assign them to p and q.
p = 191
q = 281


# Compute n, the product of p and q. This is the modulus.
n = p * q

# Compute totient, phi(n). Since p & q are prime,
# this is a straightforward computation.
totient = (p-1) * (q-1)

# Construct a list of possible values for e,
# numbers that are relatively prime to the totient.
es = [i for i in range(2,totient) if gcd(i,totient) == 1]
# Select one of them at random.
e = es[randrange(1,len(es))]


^G Get Help   ^O Write Out  ^W Where Is   ^K Cut Text   ^J Justify    ^C Cur Pos
^X Exit       ^R Read File  ^\ Replace    ^U Uncut Text ^T To Linter  ^_ Go To Line
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

d. RSA key generation and encryption with an online tool



Step 1: Set prime numbers

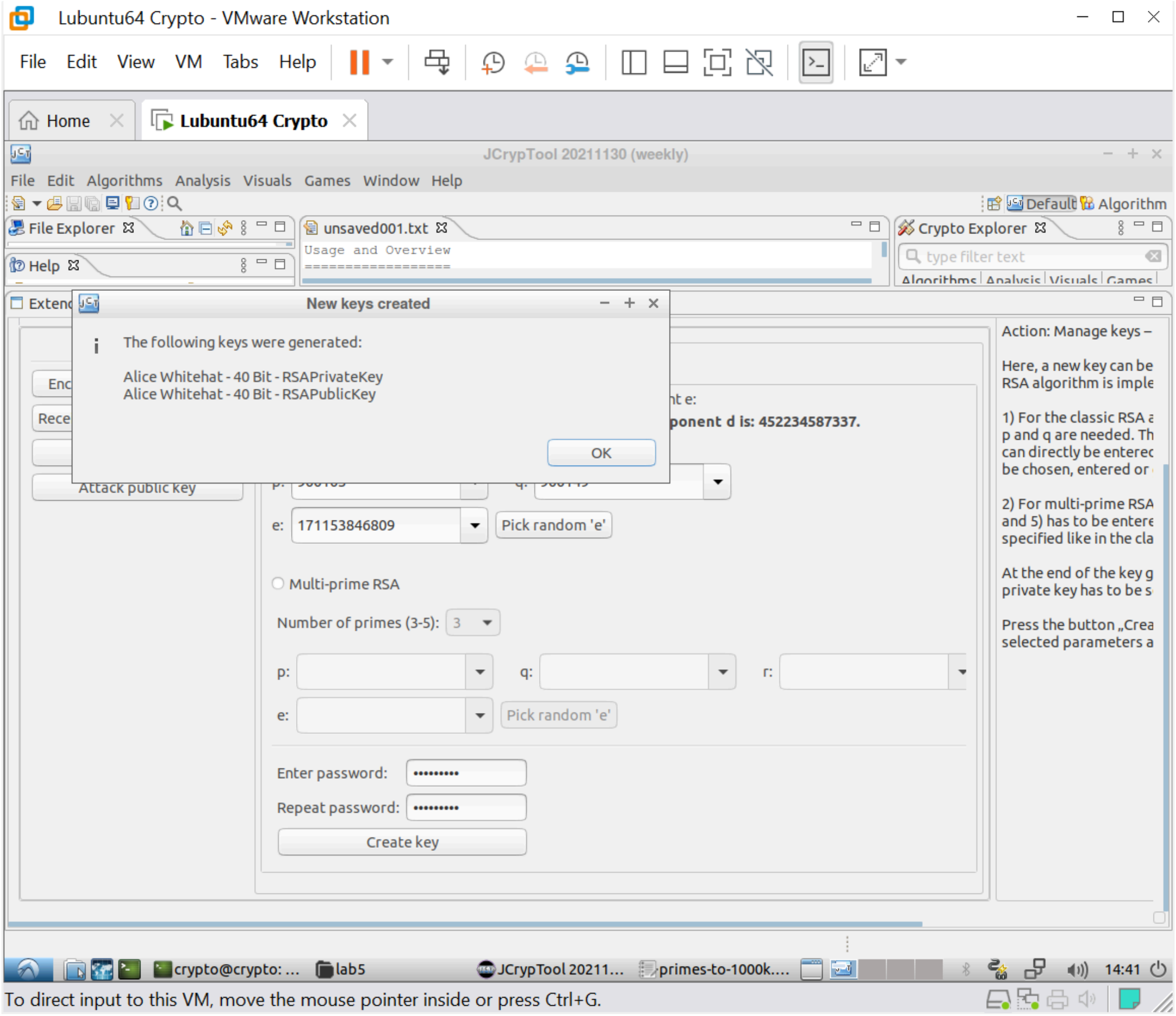
As a starting point for RSA, choose two prime numbers p and q:

$p =$ 191

$q =$ 281

Task 3 - Attack RSA keys

- Security of RSA based on how hard the Integer Factorisation Problem (IFP) is
  - Integer factorisation - the decomposition of a composite number into a product of smaller integers
  - RSA can be cracked through the exhaustive-search method when the number is large



```
Usage and Overview
==================
```

The following keys were generated:

Alice Whitehat - 40 Bit - RSAPrivateKey
Alice Whitehat - 40 Bit - RSAPublicKey

nt e:

ponent d is: 452234587337.

e: 171153846809

○ Multi-prime RSA

Number of primes (3-5): 3

p:

q:

r:

e:

Enter password: •••••••••

Repeat password: •••••••••

Action: Manage keys –

Here, a new key can be
RSA algorithm is imple

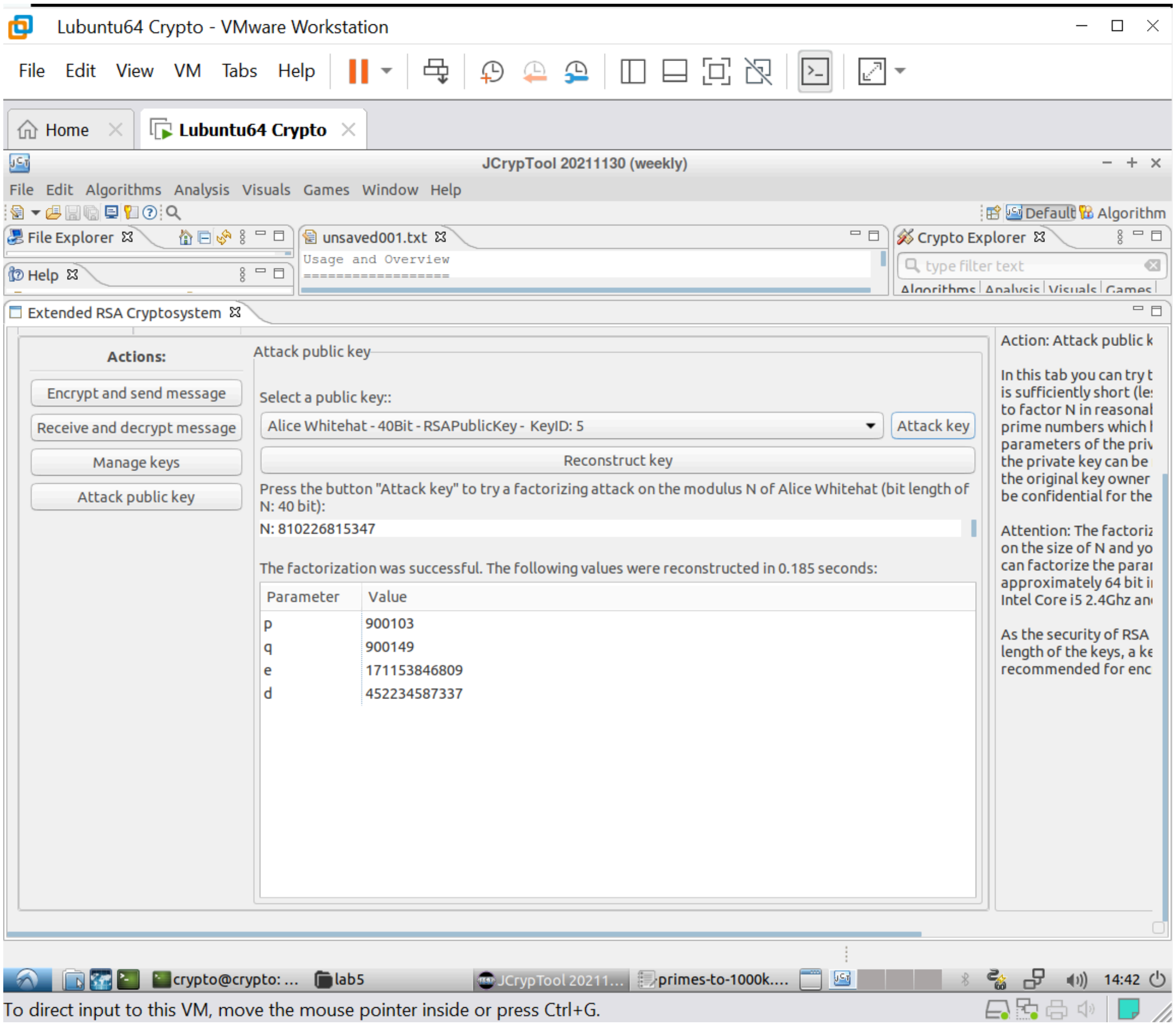
1) For the classic RSA a
p and q are needed. Th
can directly be enterec
be chosen, entered or

2) For multi-prime RSA
and 5) has to be entere
specified like in the cla

At the end of the key g
private key has to be s

Press the button „Crea
selected parameters a

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.


Lubuntu64 Crypto - VMware Workstation
File Edit View VM Tabs Help
Home
Lubuntu64 Crypto
JCrypTool 20211130 (weekly)
File Edit Algorithms Analysis Visuals Games Window Help
Default Algorithm
File Explorer
unsaved001.txt
Crypto Explorer
type filter text
Help
Algorithms Analysis Visuals Games
Extended RSA Cryptosystem
Actions:
Encrypt and send message
Receive and decrypt message
Manage keys
Attack public key
Attack key
Reconstruct key
crypto@crypto: ...
lab5
JCrypTool 20211...
primes-to-1000k....
14:42
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

```
Usage and Overview
==================
```

Attack public key

Select a public key::

Alice Whitehat - 40Bit - RSAPublicKey - KeyID: 5

Press the button "Attack key" to try a factorizing attack on the modulus N of Alice Whitehat (bit length of N: 40 bit):

N: 810226815347

The factorization was successful. The following values were reconstructed in 0.185 seconds:

Parameter	Value
p	900103
q	900149
e	171153846809
d	452234587337

Action: Attack public k

In this tab you can try t
is sufficiently short (le:
to factor N in reasonal
prime numbers which l
parameters of the priv
the private key can be
the original key owner
be confidential for the

Attention: The factoriz
on the size of N and yo
can factorize the parai
approximately 64 bit ii
Intel Core i5 2.4Ghz an

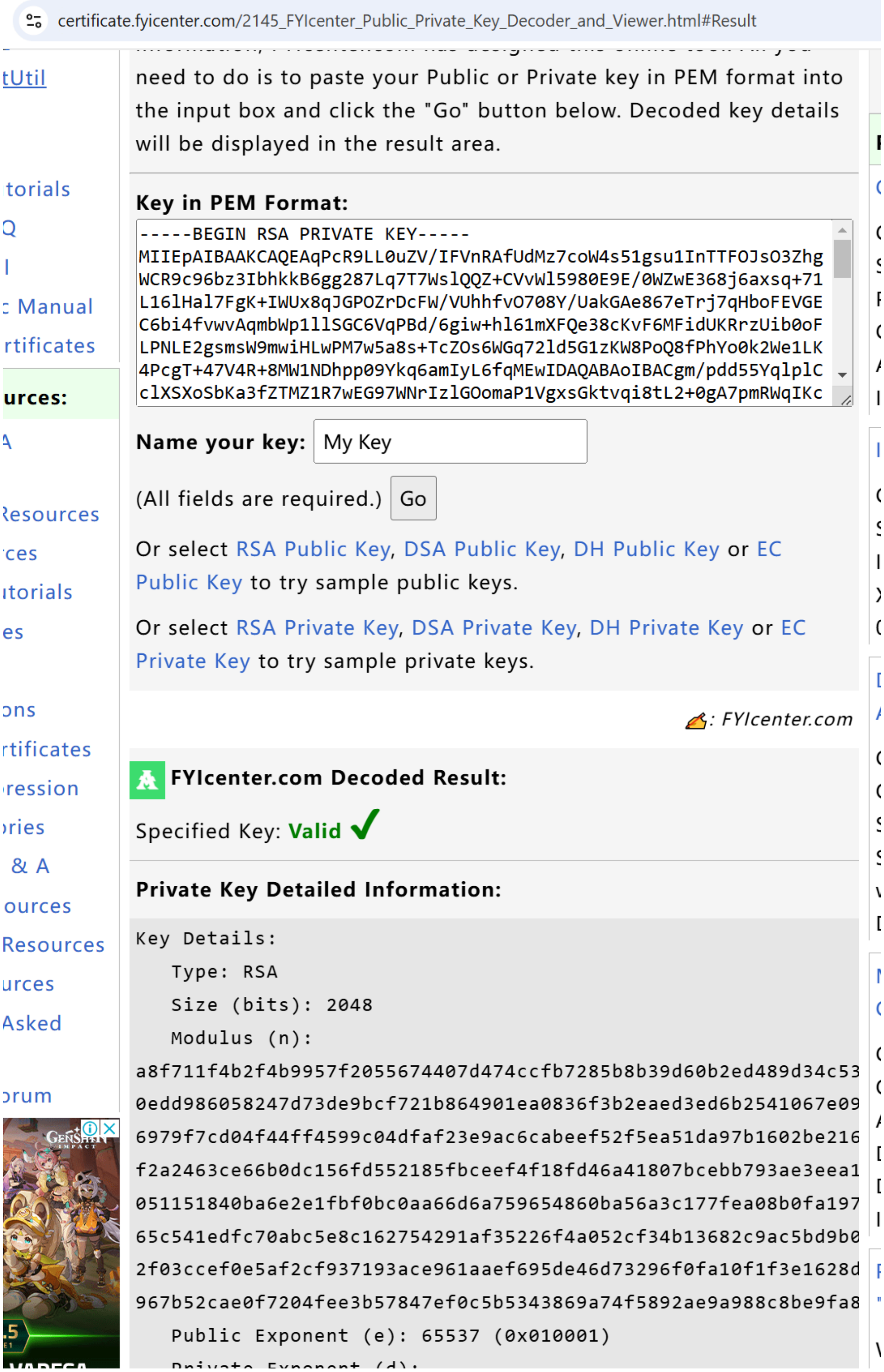
As the security of RSA
length of the keys, a ke
recommended for enc

Q: To factor a number N using exhaustive-search, i.e., try factors in 1, 2, 3......N, how many numbers at most should be tried?

Maximum numbers to check is N - 1, ensures it doesn’t leave a remainder: finds factors

Task 4 - RsA encryption and decryption

a. Generate RSA private and public key pair



need to do is to paste your Public or Private key in PEM format into the input box and click the "Go" button below. Decoded key details will be displayed in the result area.

Key in PEM Format:

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAqPcR9LL0uZV/IFVnRAfUdMz7coW4s51gsu1InTTFOJsO3Zhg
WCR9c96bz3IbhkkB6gg287Lq7T7WslQQZ+CVvWl5980E9E/0WZwE368j6axsq+71
L16lHal7FgK+IWUx8qJGPOZrDcFW/VUhhfvO708Y/UakGAe867eTrj7qHboFEVGE
C6bi4fvwvAqmbWp1llSGC6VqPBd/6giw+hl61mXFQe38cKvF6MFidUKRrzUib0oF
LPNLE2gsmsW9mwiHLwPM7w5a8s+TcZOs6WGq72ld5G1zKW8PoQ8fPhYo0k2We1LK
4PcgT+47V4R+8MW1NDhpp09Ykq6amIyL6fqMEwIDAQABAoIBACgm/pdd55YqlplC
clXSXoSbKa3fZTMZ1R7wEG97WNrIzlGOomaP1VgxsGktvqi8tL2+0gA7pmRWqIKc
```

Name your key: My Key

(All fields are required.)

Or select RSA Public Key, DSA Public Key, DH Public Key or EC Public Key to try sample public keys.

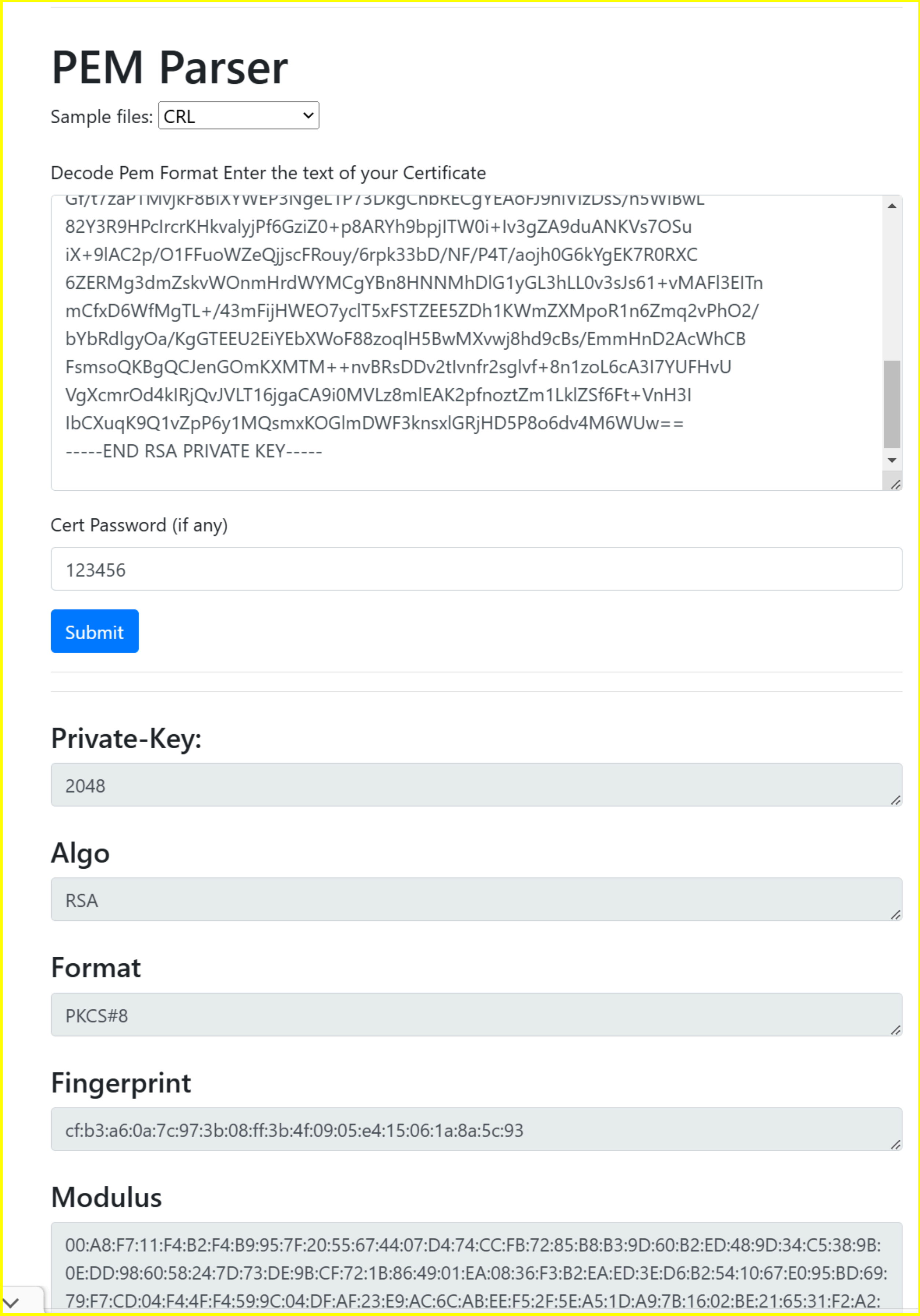
Or select RSA Private Key, DSA Private Key, DH Private Key or EC Private Key to try sample private keys.

FYIcenter.com Decoded Result:

Specified Key: **Valid** ✓

Private Key Detailed Information:

```
Key Details:
   Type: RSA
   Size (bits): 2048
   Modulus (n):
a8f711f4b2f4b9957f2055674407d474ccfb7285b8b39d60b2ed489d34c53
0edd986058247d73de9bcf721b864901ea0836f3b2eaed3ed6b2541067e09
6979f7cd04f44ff4599c04dfaf23e9ac6cabeef52f5ea51da97b1602be216
f2a2463ce66b0dc156fd552185fbceef4f18fd46a41807bcebb793ae3eea1
051151840ba6e2e1fbf0bc0aa66d6a759654860ba56a3c177fea08b0fa197
65c541edfc70abc5e8c162754291af35226f4a052cf34b13682c9ac5bd9b0
2f03ccef0e5af2cf937193ace961aaef695de46d73296f0fa10f1f3e1628d
967b52cae0f7204fee3b57847ef0c5b5343869a74f5892ae9a988c8be9fa8
   Public Exponent (e): 65537 (0x010001)
```



PEM Parser

Sample files: CRL

Decode Pem Format Enter the text of your Certificate

```
Gf/t7zaPTMvjkF8BlXYWEP3NgeLTP73DkgChbRECgYEAoFJ9hIVIzDsS/h5WIBwL
82Y3R9HPcIrcrKHkvalyjPf6GziZ0+p8ARYh9bpjITW0i+Iv3gZA9duANKVs7OSu
iX+9lAC2p/O1FFuoWZeQjjscFRouy/6rpk33bD/NF/P4T/aojh0G6kYgEK7R0RXC
6ZERMg3dmZskvWOnmHrdWYMCgYBn8HNNMhDlG1yGL3hLL0v3sJs61+vMAFl3EITn
mCfxD6WfMgTL+/43mFijHWEO7yclT5xFSTZEE5ZDh1KWmZXMpoR1n6Zmq2vPhO2/
bYbRdlgyOa/KgGTEEU2EiYEbXWoF88zoqlH5BwMXvwj8hd9cBs/EmmHnD2AcWhCB
FsmsoQKBgQCJenGOmKXMTM++nvBRsDDv2tIvnfr2sglvf+8n1zoL6cA3I7YUFHvU
VgXcmrOd4kIRjQvJVLT16jgaCA9i0MVLz8mlEAK2pfnoztZm1LklZSf6Ft+VnH3I
IbCXuqK9Q1vZpP6y1MQsmxKOGlmDWF3knsxlGRjHD5P8o6dv4M6WUw==
-----END RSA PRIVATE KEY-----
```

Cert Password (if any)

123456

Private-Key:

2048

Algo

RSA

Format

PKCS#8

Fingerprint

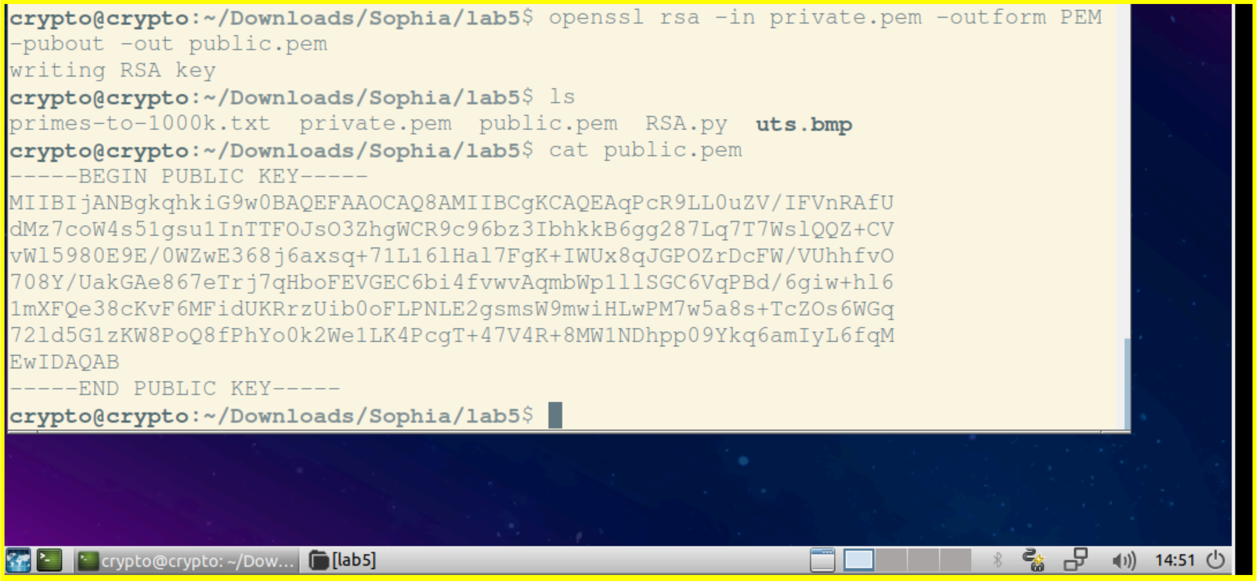
cf:b3:a6:0a:7c:97:3b:08:ff:3b:4f:09:05:e4:15:06:1a:8a:5c:93

Modulus

```
00:A8:F7:11:F4:B2:F4:B9:95:7F:20:55:67:44:07:D4:74:CC:FB:72:85:B8:B3:9D:60:B2:ED:48:9D:34:C5:38:9B:
0E:DD:98:60:58:24:7D:73:DE:9B:CF:72:1B:86:49:01:EA:08:36:F3:B2:EA:ED:3E:D6:B2:54:10:67:E0:95:BD:69:
79:F7:CD:04:F4:4F:F4:59:9C:04:DF:AF:23:E9:AC:6C:AB:EE:F5:2F:5E:A5:1D:A9:7B:16:02:BE:21:65:31:F2:A2:
```

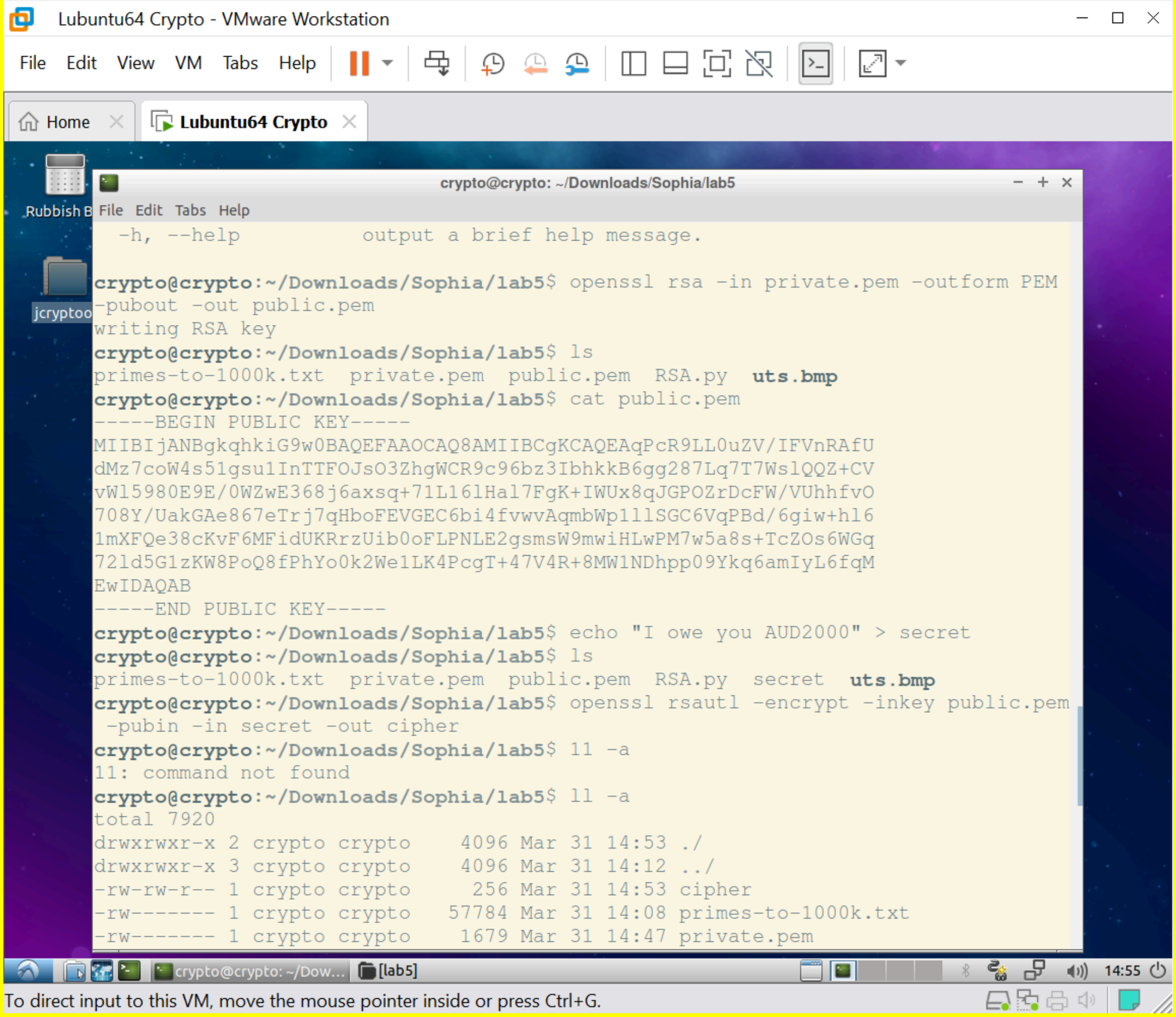
Q: Try the following tools and confirm that you can recover RSA components.

Yes i can recover the RSA components



```
crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsa -in private.pem -outform PEM
-pubout -out public.pem
writing RSA key
crypto@crypto:~/Downloads/Sophia/lab5$ ls
primes-to-1000k.txt  private.pem  public.pem  RSA.py  uts.bmp
crypto@crypto:~/Downloads/Sophia/lab5$ cat public.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqPcR9LL0uZV/IFVnRAfU
dMz7coW4s51gsu1InTTFOJsO3ZhgWCR9c96bz3IbhkkB6gg287Lq7T7WslQQZ+CV
vWl5980E9E/0WZwE368j6axsq+71L16lHal7FgK+IWUx8qJGPOZrDcFW/VUhhfvO
708Y/UakGAe867eTrj7qHboFEVGEC6bi4fvwvAqmbWp1llSGC6VqPBd/6giw+hl6
1mXFQe38cKvF6MFidUKRrzUib0oFLPNLE2gsmsW9mwiHLwPM7w5a8s+TcZOs6WGq
72ld5G1zKW8PoQ8fPhYo0k2We1LK4PcgT+47V4R+8MW1NDhpp09Ykq6amIyL6fqM
EwIDAQAB
-----END PUBLIC KEY-----
crypto@crypto:~/Downloads/Sophia/lab5$
```

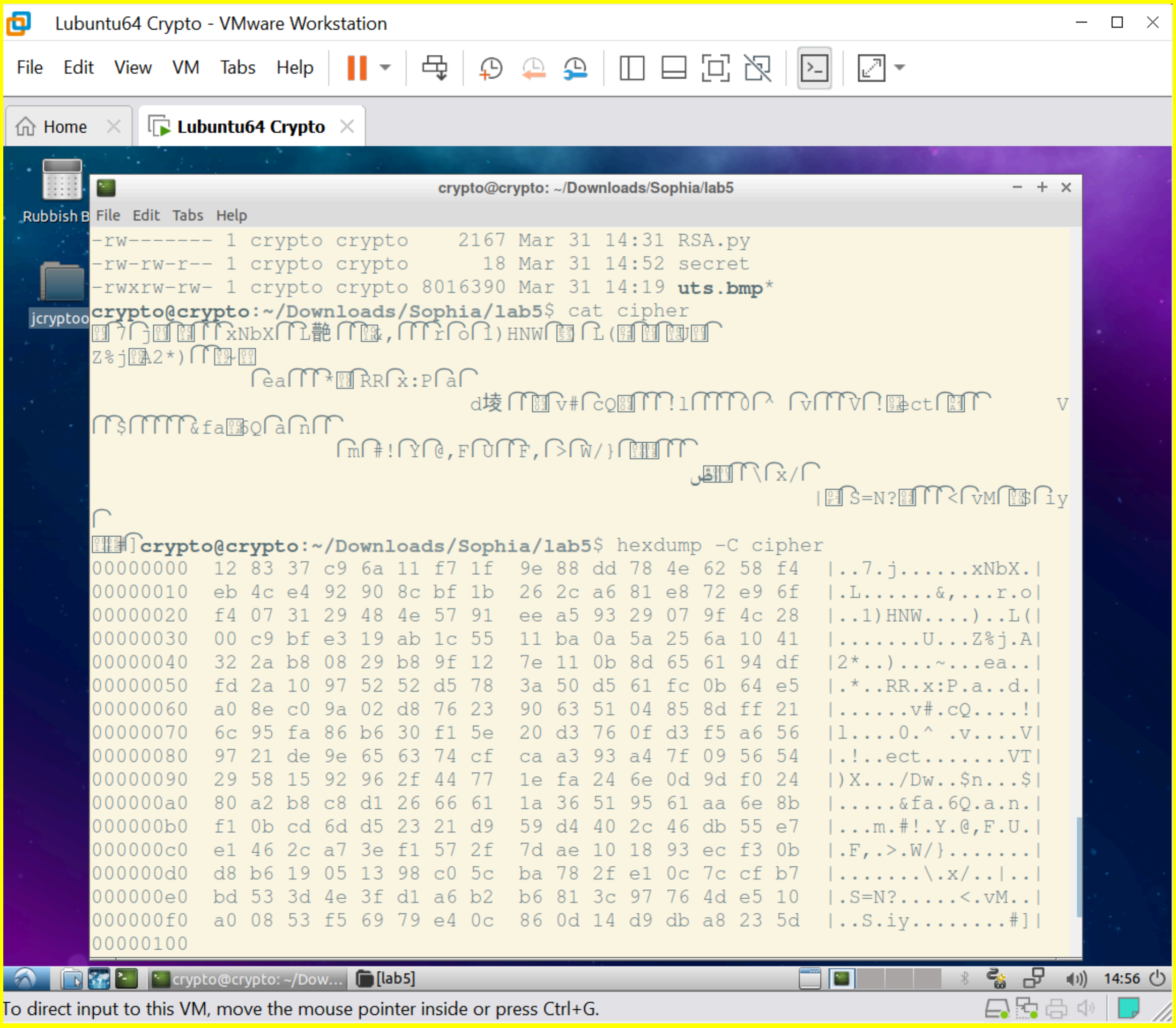
b. Encrypt a small file with the public key and decrypt with the private key



Lubuntu64 Crypto - VMware Workstation
File Edit View VM Tabs Help
Home
Lubuntu64 Crypto
Rubbish B
jcryptoo
crypto@crypto: ~/Downloads/Sophia/lab5
File Edit Tabs Help
crypto@crypto: ~/Dow...
[lab5]
14:55

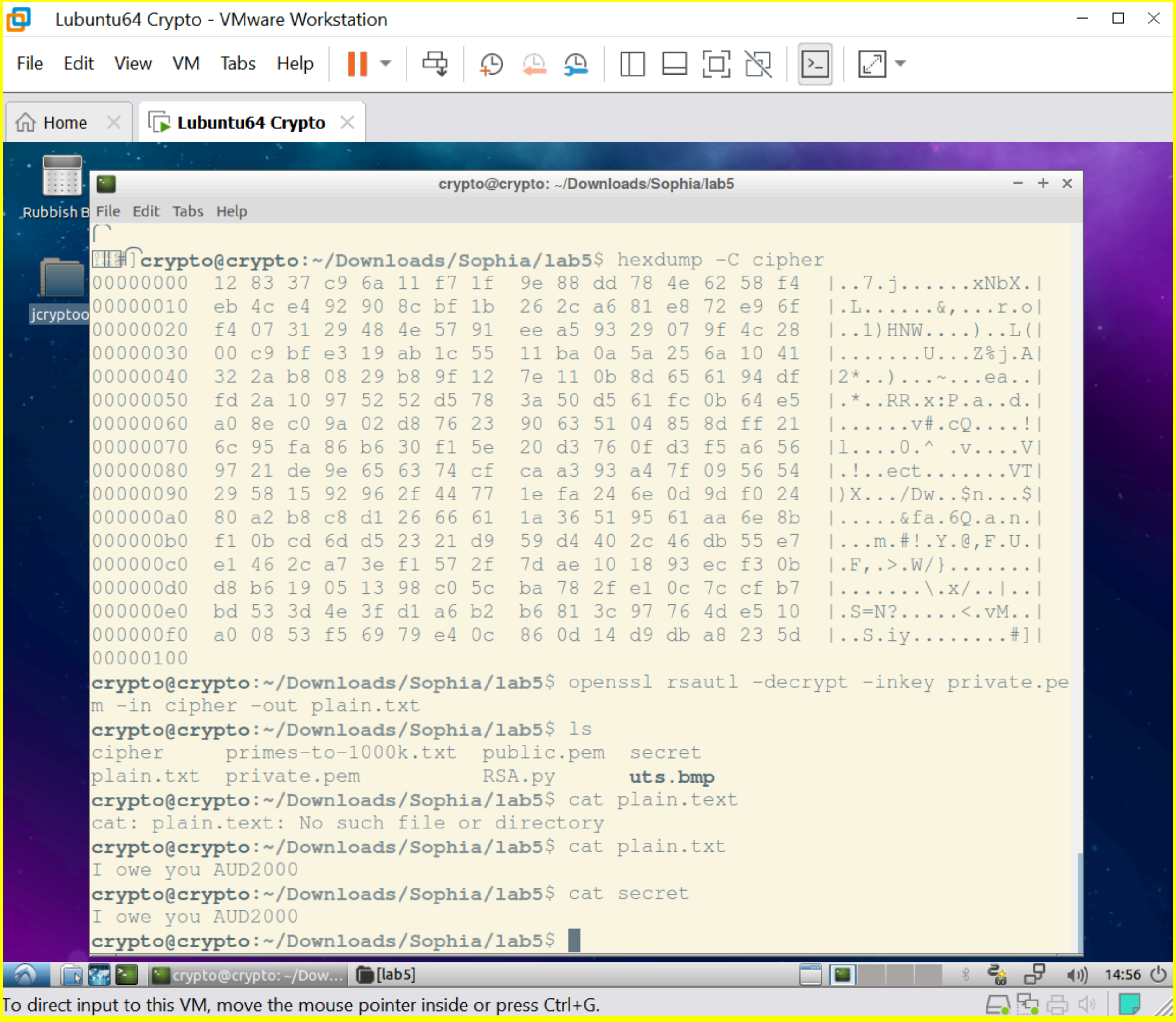
```
  -h, --help           output a brief help message.

crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsa -in private.pem -outform PEM
-pubout -out public.pem
writing RSA key
crypto@crypto:~/Downloads/Sophia/lab5$ ls
primes-to-1000k.txt  private.pem  public.pem  RSA.py  uts.bmp
crypto@crypto:~/Downloads/Sophia/lab5$ cat public.pem
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEAqPcR9LL0uZV/IFVnRAfU
dMz7coW4s51gsu1InTTFOJsO3ZhgWCR9c96bz3IbhkkB6gg287Lq7T7WslQQZ+CV
vWl5980E9E/0WZwE368j6axsq+71L161Hal7FgK+IWUx8qJGPOZrDcFW/VUhhfvO
708Y/UakGAe867eTrj7qHboFEVGEC6bi4fvwvAqmbWp111SGC6VqPBd/6giw+hl6
1mXFQe38cKvF6MFidUKRrzUib0oFLPNLE2gsmsW9mwiHLwPM7w5a8s+TcZOs6WGq
721d5G1zKW8PoQ8fPhYo0k2We1LK4PcgT+47V4R+8MW1NDhpp09Ykq6amIyL6fqM
EwIDAQAB
-----END PUBLIC KEY-----
crypto@crypto:~/Downloads/Sophia/lab5$ echo "I owe you AUD2000" > secret
crypto@crypto:~/Downloads/Sophia/lab5$ ls
primes-to-1000k.txt  private.pem  public.pem  RSA.py  secret  uts.bmp
crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsautl -encrypt -inkey public.pem
 -pubin -in secret -out cipher
crypto@crypto:~/Downloads/Sophia/lab5$ 11 -a
11: command not found
crypto@crypto:~/Downloads/Sophia/lab5$ ll -a
total 7920
drwxrwxr-x 2 crypto crypto    4096 Mar 31 14:53 ./
drwxrwxr-x 3 crypto crypto    4096 Mar 31 14:12 ../
-rw-rw-r-- 1 crypto crypto     256 Mar 31 14:53 cipher
-rw------- 1 crypto crypto   57784 Mar 31 14:08 primes-to-1000k.txt
-rw------- 1 crypto crypto    1679 Mar 31 14:47 private.pem
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.


Lubuntu64 Crypto - VMware Workstation
File Edit View VM Tabs Help
Home
Lubuntu64 Crypto
Rubbish B
jcryptoo
crypto@crypto: ~/Downloads/Sophia/lab5
File Edit Tabs Help
crypto@crypto: ~/Dow...
[lab5]
14:56

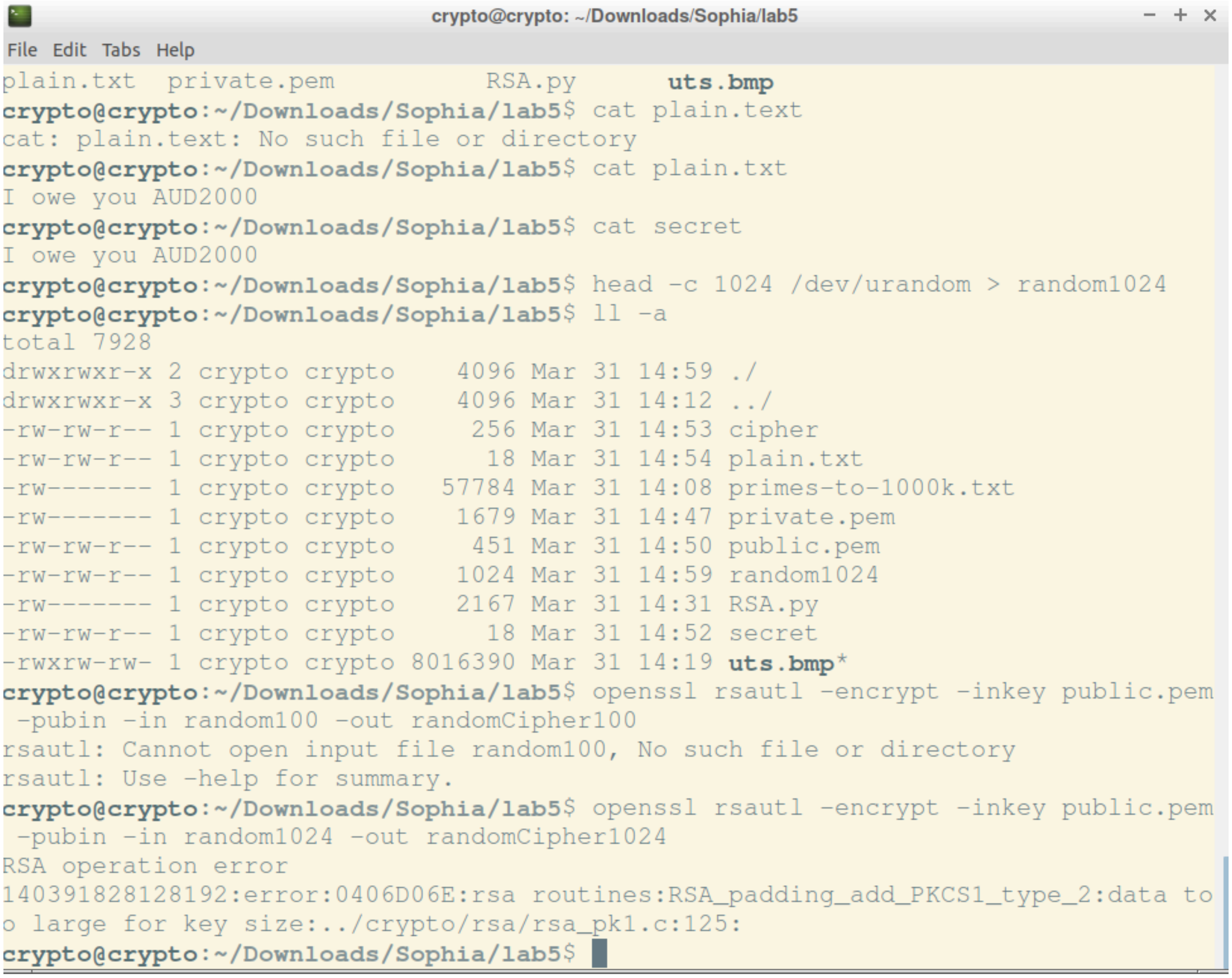
```
-rw------- 1 crypto crypto    2167 Mar 31 14:31 RSA.py
-rw-rw-r-- 1 crypto crypto      18 Mar 31 14:52 secret
-rwxrw-rw- 1 crypto crypto 8016390 Mar 31 14:19 uts.bmp*
crypto@crypto:~/Downloads/Sophia/lab5$ cat cipher
� 7� j� � ��xNbX��L艷��� &,���r�o�1)HNW� �L(� � �U�
Z%j�A2*)�� � ~�
             �ea���*� RR�x:P�a�
                                  d埈�� v#�cQ���!1����0�^ �v���V�!�ect� ��       V
��$����&fa�6Q�a�n��
                      �m�#!�Y�@,F�U��F,�>�W/}�� ��
                                                    ض� �\�x/�
                                                              |� S=N?���<�vM� $�iy
�
� #]crypto@crypto:~/Downloads/Sophia/lab5$ hexdump -C cipher
00000000  12 83 37 c9 6a 11 f7 1f  9e 88 dd 78 4e 62 58 f4  |..7.j......xNbX.|
00000010  eb 4c e4 92 90 8c bf 1b  26 2c a6 81 e8 72 e9 6f  |.L......&,...r.o|
00000020  f4 07 31 29 48 4e 57 91  ee a5 93 29 07 9f 4c 28  |..1)HNW....)..L(|
00000030  00 c9 bf e3 19 ab 1c 55  11 ba 0a 5a 25 6a 10 41  |.......U...Z%j.A|
00000040  32 2a b8 08 29 b8 9f 12  7e 11 0b 8d 65 61 94 df  |2*..)...~...ea..|
00000050  fd 2a 10 97 52 52 d5 78  3a 50 d5 61 fc 0b 64 e5  |.*..RR.x:P.a..d.|
00000060  a0 8e c0 9a 02 d8 76 23  90 63 51 04 85 8d ff 21  |......v#.cQ....!|
00000070  6c 95 fa 86 b6 30 f1 5e  20 d3 76 0f d3 f5 a6 56  |l....0.^ .v....V|
00000080  97 21 de 9e 65 63 74 cf  ca a3 93 a4 7f 09 56 54  |.!..ect.......VT|
00000090  29 58 15 92 96 2f 44 77  1e fa 24 6e 0d 9d f0 24  |)X.../Dw..$n...$|
000000a0  80 a2 b8 c8 d1 26 66 61  1a 36 51 95 61 aa 6e 8b  |.....&fa.6Q.a.n.|
000000b0  f1 0b cd 6d d5 23 21 d9  59 d4 40 2c 46 db 55 e7  |...m.#!.Y.@,F.U.|
000000c0  e1 46 2c a7 3e f1 57 2f  7d ae 10 18 93 ec f3 0b  |.F,.>.W/}.......|
000000d0  d8 b6 19 05 13 98 c0 5c  ba 78 2f e1 0c 7c cf b7  |.......\.x/..|..|
000000e0  bd 53 3d 4e 3f d1 a6 b2  b6 81 3c 97 76 4d e5 10  |.S=N?.....<.vM..|
000000f0  a0 08 53 f5 69 79 e4 0c  86 0d 14 d9 db a8 23 5d  |..S.iy........#]|
00000100
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.


Lubuntu64 Crypto - VMware Workstation
File
Edit
View
VM
Tabs
Help
Home
Lubuntu64 Crypto
Rubbish B
jcryptoo
crypto@crypto: ~/Downloads/Sophia/lab5
File
Edit
Tabs
Help
crypto@crypto: ~/Dow...
[lab5]
14:56
To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

```
crypto@crypto:~/Downloads/Sophia/lab5$ hexdump -C cipher
00000000  12 83 37 c9 6a 11 f7 1f  9e 88 dd 78 4e 62 58 f4  |..7.j......xNbX.|
00000010  eb 4c e4 92 90 8c bf 1b  26 2c a6 81 e8 72 e9 6f  |.L......&,...r.o|
00000020  f4 07 31 29 48 4e 57 91  ee a5 93 29 07 9f 4c 28  |..1)HNW....)..L(|
00000030  00 c9 bf e3 19 ab 1c 55  11 ba 0a 5a 25 6a 10 41  |.......U...Z%j.A|
00000040  32 2a b8 08 29 b8 9f 12  7e 11 0b 8d 65 61 94 df  |2*..)...~...ea..|
00000050  fd 2a 10 97 52 52 d5 78  3a 50 d5 61 fc 0b 64 e5  |.*..RR.x:P.a..d.|
00000060  a0 8e c0 9a 02 d8 76 23  90 63 51 04 85 8d ff 21  |......v#.cQ....!|
00000070  6c 95 fa 86 b6 30 f1 5e  20 d3 76 0f d3 f5 a6 56  |l....0.^ .v....V|
00000080  97 21 de 9e 65 63 74 cf  ca a3 93 a4 7f 09 56 54  |.!..ect.......VT|
00000090  29 58 15 92 96 2f 44 77  1e fa 24 6e 0d 9d f0 24  |)X.../Dw..$n...$|
000000a0  80 a2 b8 c8 d1 26 66 61  1a 36 51 95 61 aa 6e 8b  |.....&fa.6Q.a.n.|
000000b0  f1 0b cd 6d d5 23 21 d9  59 d4 40 2c 46 db 55 e7  |...m.#!.Y.@,F.U.|
000000c0  e1 46 2c a7 3e f1 57 2f  7d ae 10 18 93 ec f3 0b  |.F,.>.W/}.......|
000000d0  d8 b6 19 05 13 98 c0 5c  ba 78 2f e1 0c 7c cf b7  |.......\.x/..|..|
000000e0  bd 53 3d 4e 3f d1 a6 b2  b6 81 3c 97 76 4d e5 10  |.S=N?.....<.vM..|
000000f0  a0 08 53 f5 69 79 e4 0c  86 0d 14 d9 db a8 23 5d  |..S.iy........#]|
00000100
crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsautl -decrypt -inkey private.pe
m -in cipher -out plain.txt
crypto@crypto:~/Downloads/Sophia/lab5$ ls
cipher     primes-to-1000k.txt  public.pem  secret
plain.txt  private.pem          RSA.py      uts.bmp
crypto@crypto:~/Downloads/Sophia/lab5$ cat plain.text
cat: plain.text: No such file or directory
crypto@crypto:~/Downloads/Sophia/lab5$ cat plain.txt
I owe you AUD2000
crypto@crypto:~/Downloads/Sophia/lab5$ cat secret
I owe you AUD2000
crypto@crypto:~/Downloads/Sophia/lab5$ 
```

c. Size limitation of RSA encryption



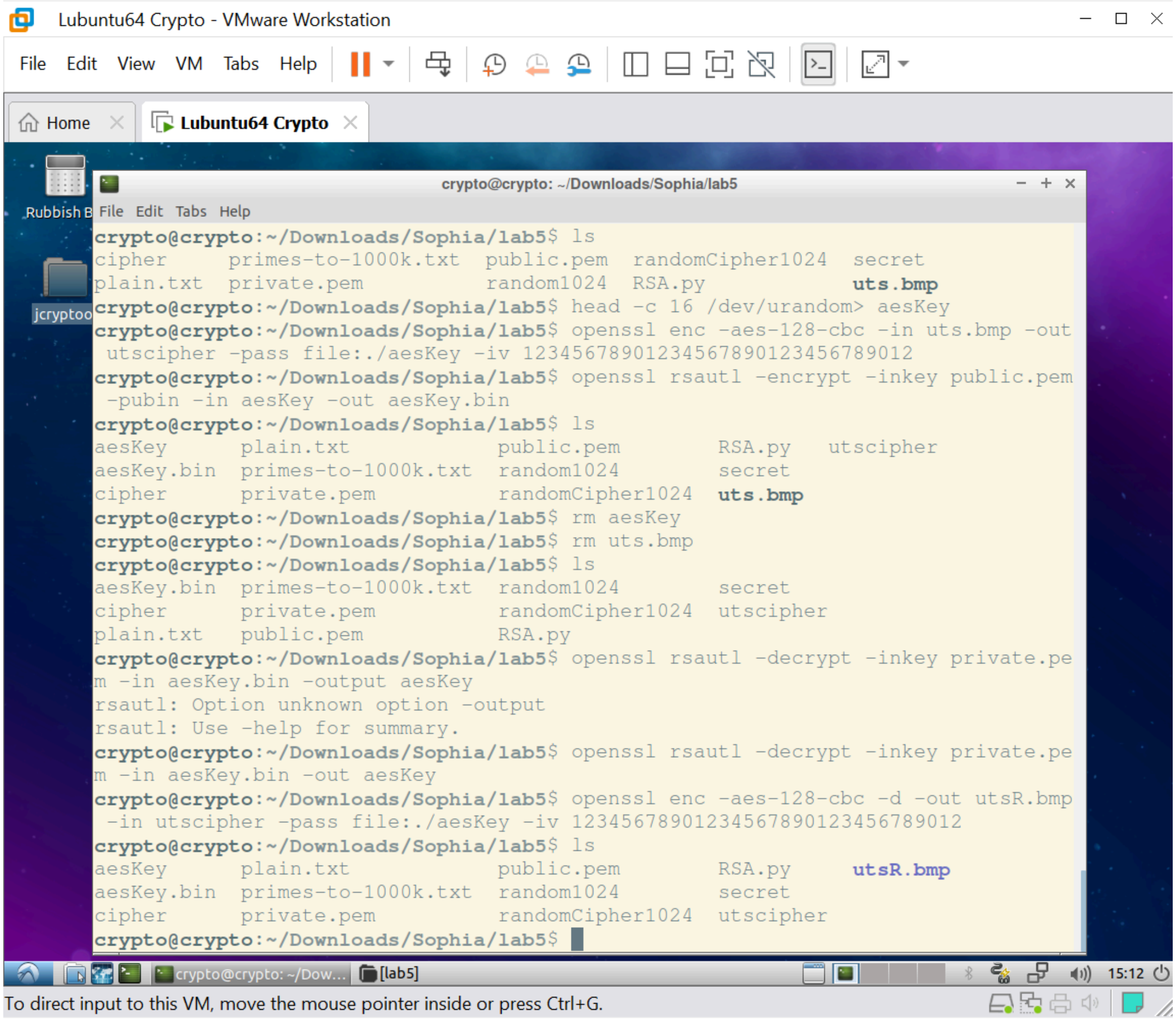
```
plain.txt  private.pem          RSA.py      uts.bmp
crypto@crypto:~/Downloads/Sophia/lab5$ cat plain.text
cat: plain.text: No such file or directory
crypto@crypto:~/Downloads/Sophia/lab5$ cat plain.txt
I owe you AUD2000
crypto@crypto:~/Downloads/Sophia/lab5$ cat secret
I owe you AUD2000
crypto@crypto:~/Downloads/Sophia/lab5$ head -c 1024 /dev/urandom > random1024
crypto@crypto:~/Downloads/Sophia/lab5$ ll -a
total 7928
drwxrwxr-x 2 crypto crypto    4096 Mar 31 14:59 ./
drwxrwxr-x 3 crypto crypto    4096 Mar 31 14:12 ../
-rw-rw-r-- 1 crypto crypto     256 Mar 31 14:53 cipher
-rw-rw-r-- 1 crypto crypto      18 Mar 31 14:54 plain.txt
-rw------- 1 crypto crypto   57784 Mar 31 14:08 primes-to-1000k.txt
-rw------- 1 crypto crypto    1679 Mar 31 14:47 private.pem
-rw-rw-r-- 1 crypto crypto     451 Mar 31 14:50 public.pem
-rw-rw-r-- 1 crypto crypto    1024 Mar 31 14:59 random1024
-rw------- 1 crypto crypto    2167 Mar 31 14:31 RSA.py
-rw-rw-r-- 1 crypto crypto      18 Mar 31 14:52 secret
-rwxrw-rw- 1 crypto crypto 8016390 Mar 31 14:19 uts.bmp*
crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsautl -encrypt -inkey public.pem
 -pubin -in random100 -out randomCipher100
rsautl: Cannot open input file random100, No such file or directory
rsautl: Use -help for summary.
crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsautl -encrypt -inkey public.pem
 -pubin -in random1024 -out randomCipher1024
RSA operation error
140391828128192:error:0406D06E:rsa routines:RSA_padding_add_PKCS1_type_2:data to
o large for key size:../crypto/rsa/rsa_pk1.c:125:
crypto@crypto:~/Downloads/Sophia/lab5$ 
```

Q: How many bytes at maximum can be encrypted with 2048-bit key RSA? Why?

245 bytes

- RSA encrypts data in blocks and is based on the size of the key
- An RSA key can handle data blocks that are 2048 bits - 8 bits in a byte
  - 2048 bits / 8 bits = 256 bytes - 10 bytes (for padding) = 245 bytes

d. Protect big files


Lubuntu64 Crypto - VMware Workstation
File Edit View VM Tabs Help
Home
Lubuntu64 Crypto
Rubbish B
jcryptoo
crypto@crypto: ~/Downloads/Sophia/lab5
File Edit Tabs Help
crypto@crypto: ~/Dow...
[lab5]
15:12

```
crypto@crypto:~/Downloads/Sophia/lab5$ ls
cipher     primes-to-1000k.txt  public.pem  randomCipher1024  secret
plain.txt  private.pem          random1024  RSA.py            uts.bmp
crypto@crypto:~/Downloads/Sophia/lab5$ head -c 16 /dev/urandom> aesKey
crypto@crypto:~/Downloads/Sophia/lab5$ openssl enc -aes-128-cbc -in uts.bmp -out
 utscipher -pass file:./aesKey -iv 12345678901234567890123456789012
crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsautl -encrypt -inkey public.pem
 -pubin -in aesKey -out aesKey.bin
crypto@crypto:~/Downloads/Sophia/lab5$ ls
aesKey      plain.txt            public.pem        RSA.py   utscipher
aesKey.bin  primes-to-1000k.txt  random1024        secret
cipher      private.pem          randomCipher1024  uts.bmp
crypto@crypto:~/Downloads/Sophia/lab5$ rm aesKey
crypto@crypto:~/Downloads/Sophia/lab5$ rm uts.bmp
crypto@crypto:~/Downloads/Sophia/lab5$ ls
aesKey.bin  primes-to-1000k.txt  random1024        secret
cipher      private.pem          randomCipher1024  utscipher
plain.txt   public.pem           RSA.py
crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsautl -decrypt -inkey private.pe
m -in aesKey.bin -output aesKey
rsautl: Option unknown option -output
rsautl: Use -help for summary.
crypto@crypto:~/Downloads/Sophia/lab5$ openssl rsautl -decrypt -inkey private.pe
m -in aesKey.bin -out aesKey
crypto@crypto:~/Downloads/Sophia/lab5$ openssl enc -aes-128-cbc -d -out utsR.bmp
 -in utscipher -pass file:./aesKey -iv 12345678901234567890123456789012
crypto@crypto:~/Downloads/Sophia/lab5$ ls
aesKey      plain.txt            public.pem        RSA.py     utsR.bmp
aesKey.bin  primes-to-1000k.txt  random1024        secret
cipher      private.pem          randomCipher1024  utscipher
crypto@crypto:~/Downloads/Sophia/lab5$ 
```

To direct input to this VM, move the mouse pointer inside or press Ctrl+G.

Q: Why are the private keys in asymmetric ciphers often generated rather than specified?

- Randomness + uniqueness - unpredictable for security = increases key strength
- Prevents key from getting leaked