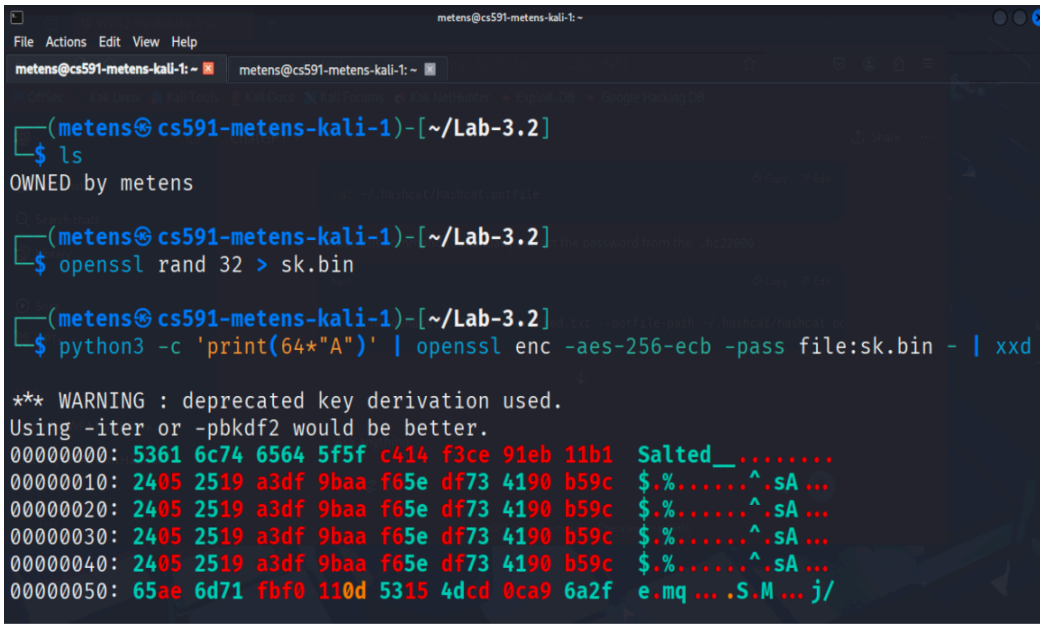
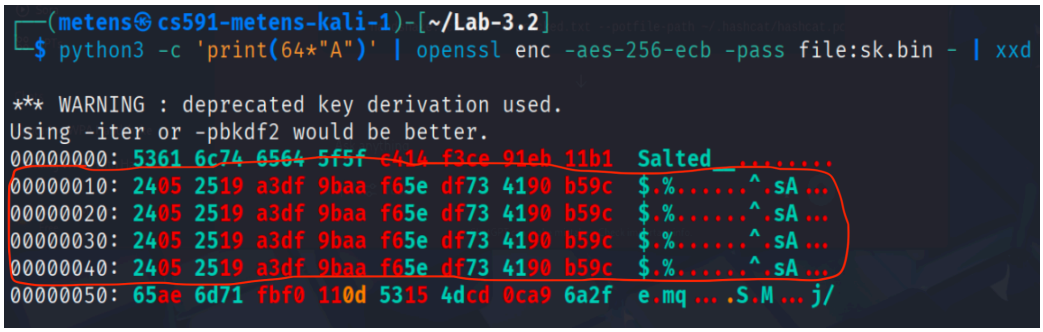
Lab 3.2

By: Nathan Metens (metens@pdx.edu)

Task 1: Symmetric encryption





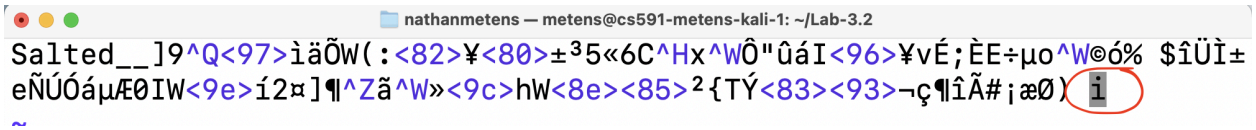
The repeated pattern of bytes in the ciphertext is “2405 2519 a3df 9baa f65e df73 4190 b59c”. Given the byte pattern, we can separate them into their bytes: “24 05 25 19 a3 df 9b aa f6 5e df 73 41 90 b5 9c”. The block size in bits of each repeated line is 16 bytes long. Since one byte is 8 bits, we have 8 * 16 = 128 total bits.

Task 2: AES CBC mode

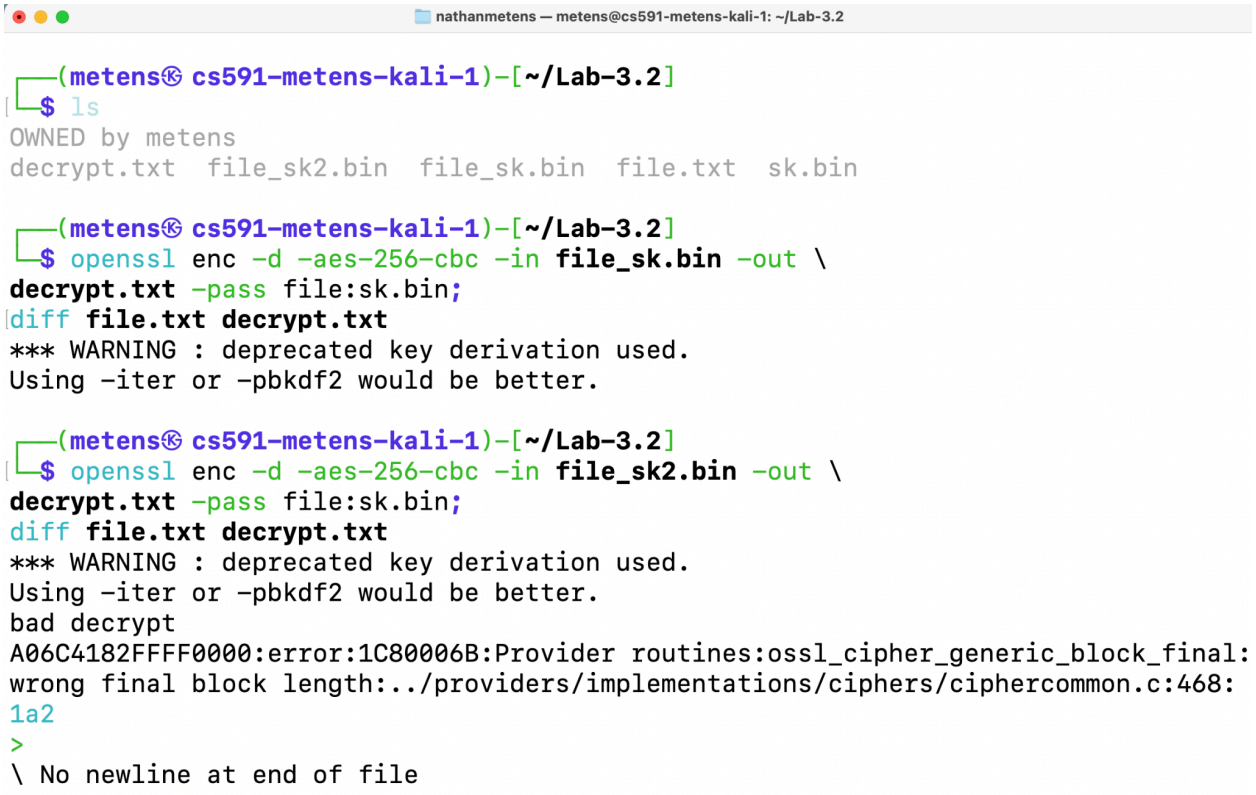
```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ python3 -c 'print(64*"A")' \
| openssl enc -aes-256-cbc -pass file:sk.bin - | xxd
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
00000000: 5361 6c74 6564 5f5f 21ff e912 9d50 805f  Salted__!....P._
00000010: fb1b 80e5 272a 2431 8839 dedc ad6f 5059  ....'*$1.9...oPY
00000020: 55d5 11da 637c ab5e 356c 3d54 3949 d3e3  U...c|.^5l=T9I..
00000030: 3318 9933 8e23 5720 4ef9 ef96 d996 ff1d  3..3.#W N.......
00000040: 3c67 2dbb 46ae 5d8f 28a9 fd80 ed46 58ec  <g-.F.].(....FX.
00000050: 58a3 f22e cde0 f7ef 9954 15c1 1bfe 8f84  X........T......

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ python3 -c 'print(64*"A")' > file.txt

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ openssl enc -aes-256-cbc -in file.txt -out file_sk.bin \
-pass file:sk.bin;
xxd file_sk.bin
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
00000000: 5361 6c74 6564 5f5f 0957 a209 07eb 1848  Salted__.W.....H
00000010: 7f8d bfe6 8523 e3ec c6a3 1781 abfd 53b7  .....#........S.
00000020: 8592 9e5e 793c 27c3 26cb 98ee 0fc4 5ee0  ...^y<'.&.....^.
00000030: 7b67 e3cf da51 cc09 b0eb d69b 443b fe48  {g...Q......D;.H
00000040: 7a33 5be5 de1b f677 b1ec 447d 9a22 456c  z3[....w..D}."El
00000050: 161e 9556 28e5 b6e4 0fb2 50df 0bdf d9c9  ...V(.....P.....

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ openssl enc -aes-256-cbc -in file.txt -out file_sk2.bin \
-pass file:sk.bin;
xxd file_sk2.bin
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
00000000: 5361 6c74 6564 5f5f 5d39 1197 ece4 d557  Salted__]9.....W
00000010: 283a 82a5 80b1 b335 ab36 4308 7817 d422  (:.....5.6C.x.."
00000020: fbe1 4996 a576 c93b c845 f7b5 6f17 a9f3  ..I..v.;.E..o...
00000030: 25a0 24ee dccc b165 d1da d3e1 b5c6 3049  %.$....e......0I
00000040: 579e ed32 a45d b61a e317 bb9c 6857 8e85  W..2.]......hW..
00000050: b27b 54dd 8393 ace7 b6ee c323 a1e6 d829  .{T........#...)

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$
```

The result of each encryption is different from the last because the salt is randomly changed for each new encryption. The “Salted__” part of each new hash takes up 8 of the total 16 bytes of data in the first line. The remaining 8 bytes are for the salt. So, “5361 6c74 6564 5f5f 5d39 1197 ece4 d557 Salted__]9.....W” means that “5361 6c74 6564 5f5f” == “53 61 6c 74 65 64 5f 5f” and with ascii translation, it means “S a l t e d _ _”. This is 8 bytes. Then after the second underscore, the salt is “5d39 1197 ece4 d557” == “5d 39 11 97 ec e4 d5 57” == “] ‘ . . . . . 9”. The dots in the salt are characters not found in the ASCII table.

For the decryption of the file, I ran the command, and then once I got the expected result, I changed the file_sk2.bin so that it wouldn't correctly decrypt to see what would happen:

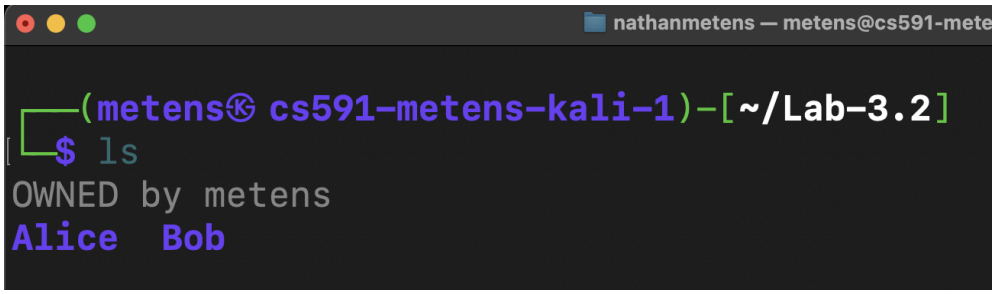


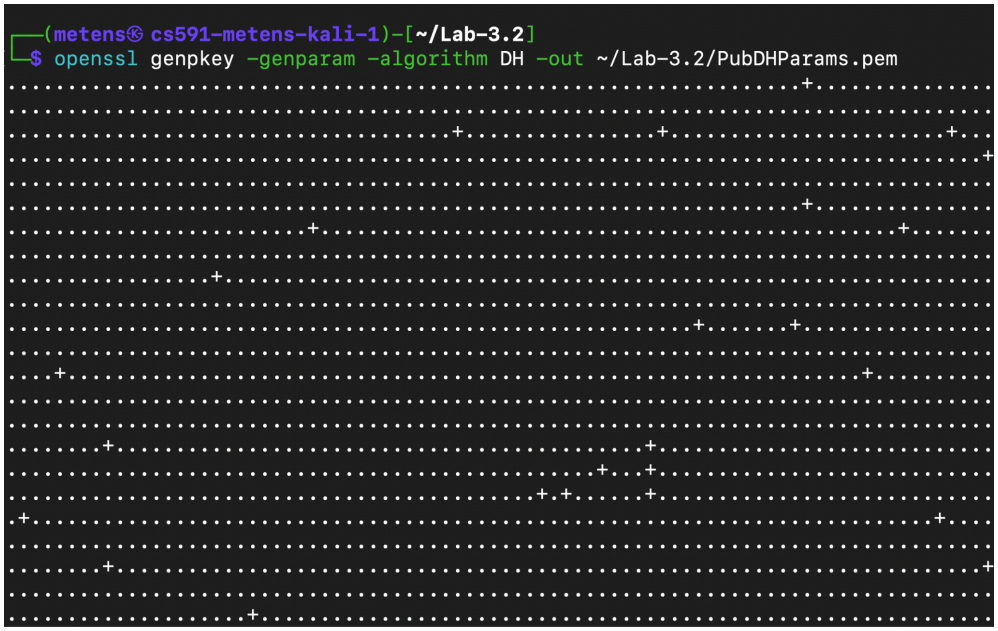
I added an i at the end of the file…

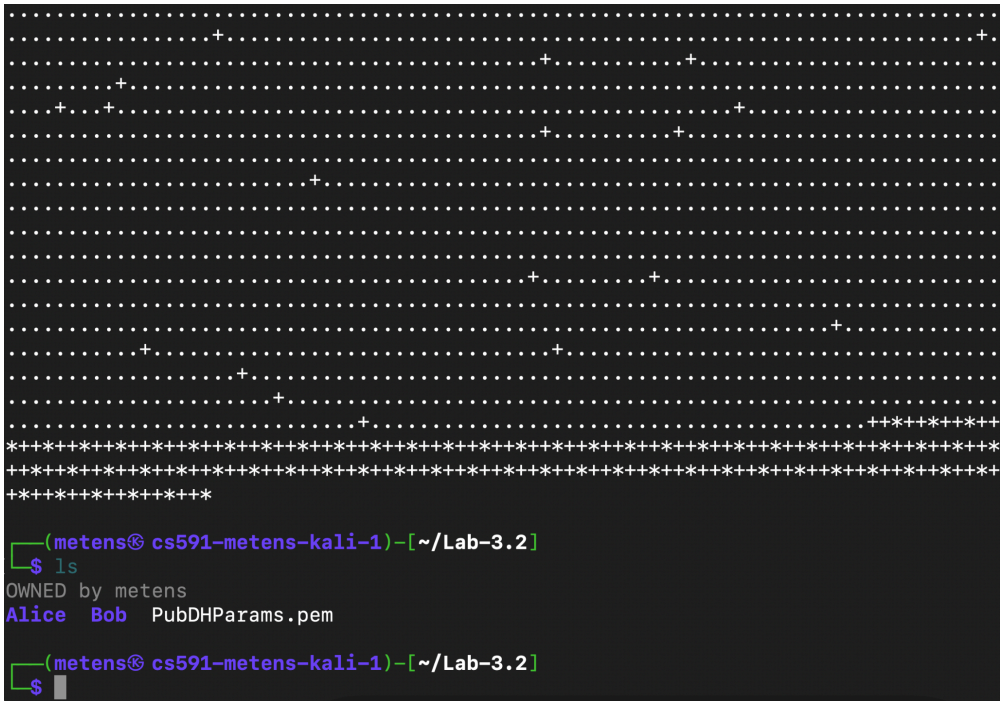


The result of the first decryption was as expected; there was no difference between file.txt and decrypt.txt. However, with the modified file, I encountered a “bad decrypt” error.

Task 3: Diffie-Hellman key exchange







```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ cat ~/Lab-3.2/PubDHParams.pem
-----BEGIN DH PARAMETERS-----
MIIBDAKCAQEAprzaXr/5bOTDPcA/RtCKGH9MPKFd6LYyjwPKA3J+RQ0AIfJ3VYE8
ruvvOM99D8+aSGxKJRM0z09eEx0TeaTW/9EP0aFwdCQBcnYzwmeCnX7nqV6Dey5H
XaXwX6Po69OvS6/UptNozJi2ZKOl/FiiPwLirKWBDhSYJcOo3XKr0Hl8MUL2VX7P
IeLOF2OLGNMMf+4tbke8yWilrwwHbsKmtyDqnAiHtnjAePEnjBWQZL+bepXnmgMG
9VWzc6rwV4UBiESE7Ts6PbP9EtFY+0hL4xWRVV7DqTeqiZygc5raCspuCJwjEwmt
R7GigzrOuYetnukJyWIQMb6/EbSO19VqXwIBAgICAOE=
-----END DH PARAMETERS-----

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ openssl pkeyparam -in ~/Lab-3.2/PubDHParams.pem -text -noout
DH Parameters: (2048 bit)
P:
    00:a6:bc:da:5e:bf:f9:6c:e4:c3:3d:c0:3f:46:d0:
    8a:18:7f:4c:3c:a1:5d:e8:b6:32:8f:03:ca:03:72:
    7e:45:0d:00:21:f2:77:55:81:3c:ae:eb:ef:38:cf:
    7d:0f:cf:9a:48:6c:4a:25:13:34:cf:4f:5e:13:1d:
    13:79:a4:d6:ff:d1:0f:d1:a1:70:74:24:01:72:76:
    33:c2:67:82:9d:7e:e7:a9:5e:83:7b:2e:47:5d:a5:
    f0:5f:a3:e8:eb:d3:af:4b:af:d4:a6:d3:68:cc:98:
    b6:64:a3:a5:fc:58:a2:3f:02:e2:ac:a5:81:0e:14:
    98:25:c3:a8:dd:72:ab:d0:79:7c:31:42:f6:55:7e:
    cf:21:e2:ce:17:63:8b:18:d3:0c:7f:ee:2d:6e:47:
    bc:c9:68:a5:af:0c:07:6e:c2:a6:b7:20:ea:9c:08:
    87:b6:78:c0:78:f1:27:8c:15:90:64:bf:9b:7a:95:
    e7:9a:03:06:f5:55:b3:73:aa:f0:57:85:01:88:44:
    84:ed:3b:3a:3d:b3:fd:12:d1:58:fb:48:4b:e3:15:
    91:55:5e:c3:a9:37:aa:89:9c:a0:73:9a:da:0a:ca:
    6e:08:9c:23:13:09:ad:47:b1:a2:83:3a:ce:b9:87:
    ad:9e:e9:09:c9:62:10:31:be:bf:11:b4:8e:d7:d5:
    6a:5f
G:    2 (0x2)
recommended-private-length: 225 bits
```

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ cd Bob

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ openssl genpkey -paramfile ../PubDHParams.pem -out BobPrivDHKey.pem;

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ openssl pkey -in BobPrivDHKey.pem -text -noout
DH Private-Key: (2048 bit)
private-key:
    66:16:95:d7:e1:e9:dc:44:f5:30:06:84:0c:b9:ac:
    0e:6a:2d:24:17:05:ff:2d:90:88:7e:9f:6f:84:55:
    17:b8:23:95:05:5a:90:7d:37:ac:e7:94:74:10:ea:
    2c:94:8e:b9:c1:31:85:9c:f8:aa:cf:32:e9:43:54:
    0b:f8:a9:d0:f6:f2:67:7c:e1:43:4b:05:f7:a7:47:
    14:63:e9:c5:9e:c5:2e:31:c4:b2:64:2d:66:31:83:
    d9:e5:1b:f8:ce:64:4b:a0:58:f6:ce:23:a6:b7:5c:
    0b:b0:07:61:6c:87:38:f2:8b:47:00:79:eb:68:99:
    34:45:b2:f9:9c:37:1d:23:82:13:df:9c:20:35:95:
    93:aa:87:26:86:ab:d5:2e:14:0c:a4:14:e8:38:71:
    73:bd:f8:5a:35:db:bf:10:83:75:8f:88:5b:61:15:
    b5:ff:7b:9c:3d:c9:97:7e:07:20:71:67:d6:81:46:
    e6:eb:f6:0e:bf:51:50:07:ae:c2:8f:f3:2f:48:10:
    31:2f:36:f7:c5:8c:26:fb:02:68:e6:e2:1a:fa:27:
    0d:d0:4b:ca:65:a1:93:76:53:34:06:71:51:56:cb:
    f9:3c:63:67:ae:02:1d:fe:1c:b7:d5:50:ca:30:dd:
    a2:66:c3:b8:9b:39:97:e0:96:d0:a1:88:35:6c:b2:
    04
public-key:
    6b:98:14:85:39:f0:40:96:0a:a7:48:de:6a:54:5e:
    cf:cf:0f:4d:dc:ad:71:29:fc:01:4e:bd:4a:f7:0e:
    79:21:80:b5:7d:c3:a2:47:b7:81:d4:7f:a8:34:a7:
    a5:23:2f:03:db:43:c5:40:45:8a:09:24:22:f5:45:
    9e:be:b4:43:86:9f:22:e8:1b:2c:df:89:16:d9:eb:
    57:5c:29:20:b3:01:68:ce:c6:7c:27:13:f6:03:94:
    1e:34:5c:b7:43:26:45:12:e4:6b:c1:e8:c8:5e:6f:
```

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ openssl pkey -in BobPrivDHKey.pem -pubout -out ../BobPubDHKey.pem;

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ openssl pkey -pubin -in ../BobPubDHKey.pem -text -noout
DH Public-Key: (2048 bit)
public-key:
    6b:98:14:85:39:f0:40:96:0a:a7:48:de:6a:54:5e:
    cf:cf:0f:4d:dc:ad:71:29:fc:01:4e:bd:4a:f7:0e:
    79:21:80:b5:7d:c3:a2:47:b7:81:d4:7f:a8:34:a7:
    a5:23:2f:03:db:43:c5:40:45:8a:09:24:22:f5:45:
    9e:be:b4:43:86:9f:22:e8:1b:2c:df:89:16:d9:eb:
    57:5c:29:20:b3:01:68:ce:c6:7c:27:13:f6:03:94:
    1e:34:5c:b7:43:26:45:12:e4:6b:c1:e8:c8:5e:6f:
    df:85:fc:08:ba:01:5b:97:26:48:63:3b:b4:8f:52:
    6b:81:32:1a:5e:b5:75:67:7c:1c:5b:1a:f7:6d:52:
    b0:64:c0:40:55:cf:eb:a7:cf:d2:b2:bd:bf:2c:1f:
    fa:22:5a:1d:4a:40:70:40:ce:c5:99:4e:a3:ff:3f:
    2f:d3:d1:a2:53:ac:71:7f:0b:17:3c:0d:b0:0f:92:
    b9:02:3d:99:9f:80:4a:a7:30:63:7f:d9:6f:c7:ec:
    7d:9b:6b:d3:1c:66:56:51:2b:28:10:55:cc:3d:73:
    af:b8:ad:a5:9d:db:1b:a0:11:27:6d:c4:94:ab:08:
    22:d5:95:00:7f:b6:96:4f:de:25:a3:8b:71:85:b4:
    7f:ba:8b:62:e9:82:10:d6:85:e6:44:75:a3:95:0d:
    5a
P:
    00:a6:bc:da:5e:bf:f9:6c:e4:c3:3d:c0:3f:46:d0:
    8a:18:7f:4c:3c:a1:5d:e8:b6:32:8f:03:ca:03:72:
    7e:45:0d:00:21:f2:77:55:81:3c:ae:eb:ef:38:cf:
    7d:0f:cf:9a:48:6c:4a:25:13:34:cf:4f:5e:13:1d:
    13:79:a4:d6:ff:d1:0f:d1:a1:70:74:24:01:72:76:
    33:c2:67:82:9d:7e:e7:a9:5e:83:7b:2e:47:5d:a5:
    f0:5f:a3:e8:eb:d3:af:4b:af:d4:a6:d3:68:cc:98:
    b6:64:a3:a5:fc:58:a2:3f:02:e2:ac:a5:81:0e:14:
    98:25:c3:a8:dd:72:ab:d0:79:7c:31:42:f6:55:7e:
```

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ ls
OWNED by metens
BobPrivDHKey.pem

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ cd ../Alice

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ openssl genpkey -paramfile ../PubDHParams.pem \
-out AlicePrivDHKey.pem;

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ openssl pkey -in AlicePrivDHKey.pem -pubout \
-out ../AlicePubDHKey.pem

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ ls
OWNED by metens
AlicePrivDHKey.pem

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$
```

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ ls
OWNED by metens
Alice  AlicePubDHKey.pem  Bob  BobPubDHKey.pem  PubDHParams.pem
```

Task 4: Secret key generation

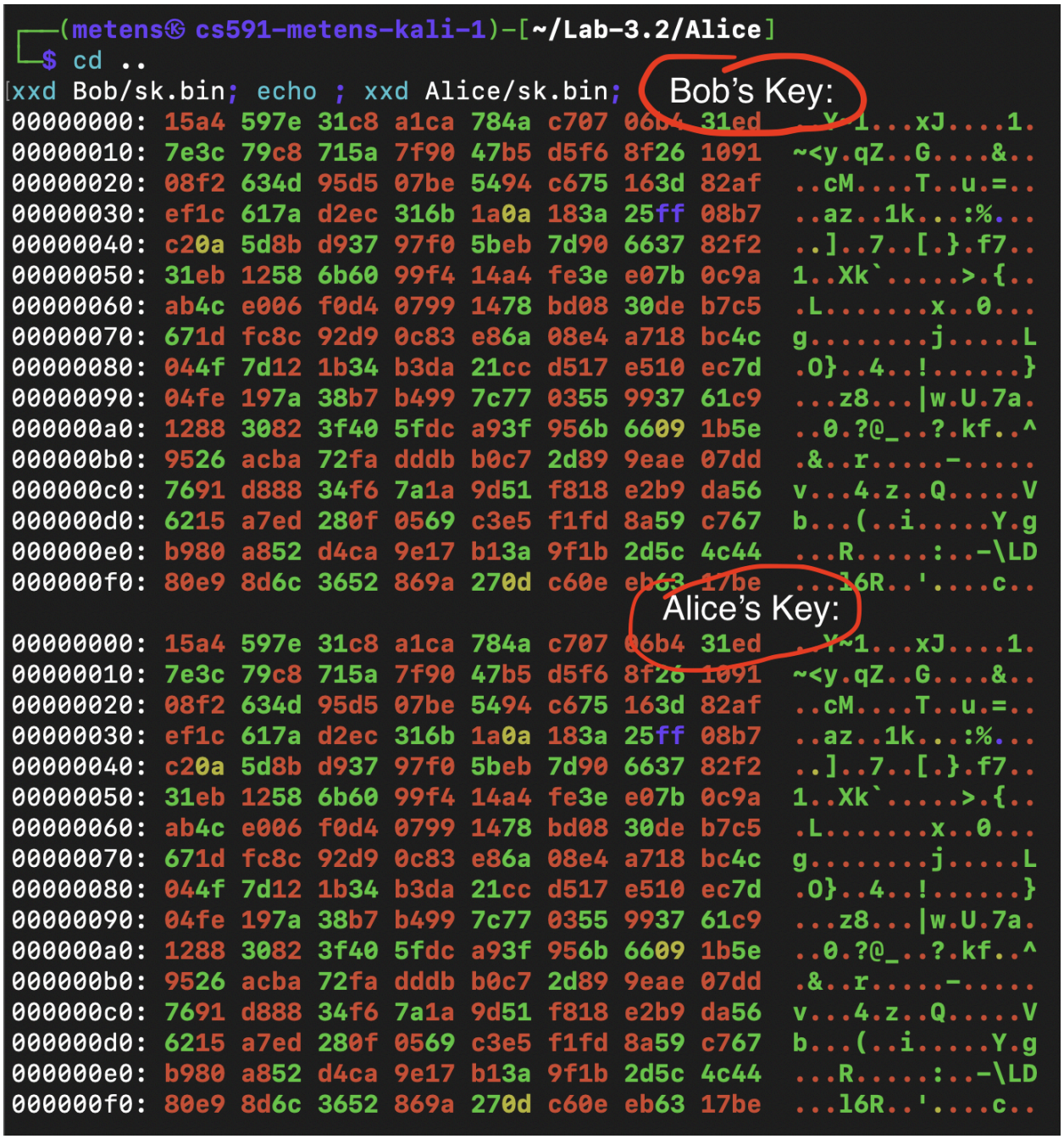
The Diffie-Hellman algorithm allows Bob to take Alice's public key (~/Lab-3.2/AlicePubDHKey.pem) and his private key (BobPrivDHKey.pem) to generate another key (Bob_sk.bin) that combines the two. Here is the demonstration:

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ cd Bob

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ openssl pkeyutl -derive -inkey BobPrivDHKey.pem -peerkey \
../AlicePubDHKey.pem -out sk.bin

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ ls
OWNED by metens
BobPrivDHKey.pem  sk.bin
```

And similarly, Alice can generate the same exact key that Bob has using her private key and Bob's public key:

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ cd ../Alice
openssl pkeyutl -derive -inkey AlicePrivDHKey.pem -peerkey \
../BobPubDHKey.pem -out sk.bin

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ ls
OWNED by metens
AlicePrivDHKey.pem  sk.bin
```

Now, both Alice and Bob have their private key, public key, and the same sk.bin key.



As we can see, both of the sk.bin keys are the same. Alice and Bob have the same private decrypt key that only they can see and use to decrypt messages.

The comparison resulted in no output. No news is good news; the keys are a match.

Task 5: Asymmetric encryption

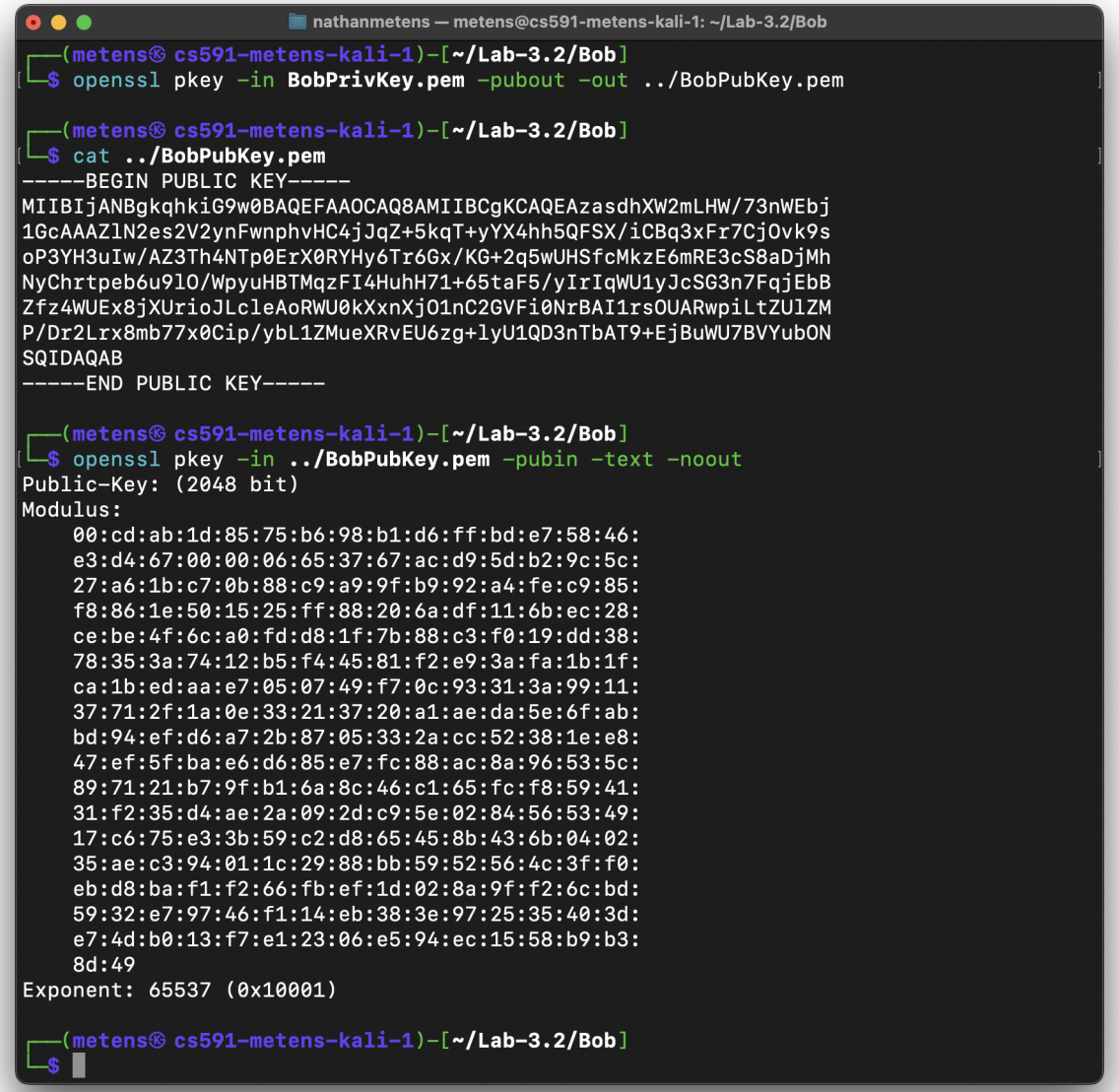
```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ ls
OWNED by metens
Alice  Bob

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ cd Bob

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ ls
OWNED by metens
BobPrivDHKey.pem

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ openssl genpkey -algorithm rsa -out BobPrivKey.pem

..+.....+.+.......+.+.....+.....+.+..+...........+.....+.......+.....+.....+..
.+.....+.....+....+.....+.....++++++++++++++++++++++++++++++++++++*.............
+.....................+..+.......+......+.....+......+..........+.....+.+...+++++++++
++++++++++++++++++++++++++++++*....+...+...........+.......+..+....+.....+....+.....
......+.+...+.....+.+.....+.........+.........+.....+...+....+.....................+
.....+.........+......+...+.+...+.....+.+..........+....+..............+.........
.+..+......+.......+...........+..+.+.........+.....+....+..+....+...+............
.+...........+.......+...........+..+.+..+.......+...+........+.......+........+....
...+........+.........+.............+.+...+...........+.....+.............+.....
.....+...+..+.......+..............+.....+.........+.....+............+...+.....+.
+.....+....+......+...+........+.......+..+.........+................+........+......
............+....+.....+.+........+.+........+.......+...+...+.....+.........+.......
.........+...+..+....+...+.....+..............+...........+........+......+.+..+..
....+....+.....+..............+...........+...+.+............+.+.........+......+
..+...+....+.....+............+........+.+.....+......+...+...........+...+.......+
..+...+...........+.......+.....................................+...+....+.........
......+......+...+.........+.......+........+++++++
...+......+++++++++++++++++++++++++++++++++++*...........+.......+.....+.+..+...
..........+.........+...+........+.........................+.........++++++++++
++++++++++++++++++++++++++*...+..++++++

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$
```

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ ls
OWNED by metens
BobPrivDHKey.pem  BobPrivKey.pem

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ cat BobPrivKey.pem
-----BEGIN PRIVATE KEY-----
MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQDNqx2FdbaYsdb/
vedYRuPUZwAABmU3Z6zZXbKcXCemG8cLiMmpn7mSpP7JhfiGHlAVJf+IIGrfEWvs
KM6+T2yg/dgfe4jD8BndOHg1OnQStfRFgfLpOvobH8ob7arnBQdJ9wyTMTqZETdx
LxoOMyE3IKGu2l5vq72U79anK4cFMyrMUjge6EfvX7rm1oXn/IisipZTXIlxIbef
sWqMRsFl/PhZQTHyNdSuKgktyV4ChFZTSRfGdeM7WcLYZUWLQ2sEAjWuw5QBHCmI
u1lSVkw/8OvYuvHyZvvvHQKKn/JsvVky55dG8RTrOD6XJTVAPedNsBP34SMG5ZTs
FVi5s41JAgMBAAECggEAB3uo3NC+Hosl7/MoZjFaG8HNobKZ3yyyAMykcYT22lfb
Y6OL9tR9a4T+UjXCJ5sVvdc+l2TajpgzLYoEj68THD5wLSERBlQHlOF3mjQwwfeF
1juWq36+V9tNvrtkgvhPXhUTi414fNZABqm4kHiyxi81QsfmNLjycmdRj46JYJp9
yoNFfdlhoCFiwRMeKiQP3PD9Nl1iTf7Chpm78GQwh6bW6Z8kf5I81j230TbN1kOM
BqXELsuEu0mSvGzwAjJyHFU8HT49yx08J6Qi3V4YenHPs38t1Co5X/KNbyIwvC5z
6ZH6yjQsvBRpTrfNyZh1CqFzuDmX72k3cZlMYcuv6QKBgQDpRL/k9khv/bz2KrFT
UEaV0Oqc/TEHDCK0LulqaElOhaEIVcJKxTO/11kEpFsiwhMUo/t6i1i7iu5GWzhq
GGpNwmpasVCfY2ohS0vNpXT6I1oIR1VsS8TYe4MIAYMnK2gEbgk6SMHgnjtvzFZQ
l9c4eQ0MngXbGlcXHeCDldeFdQKBgQDhtdZIhGCOuRwLRlTW/JrW/2RD6fDRo6JA
ixw4Yj67akbxF2ezTEc7Kz/OmIBsalNFot8or1z6S/Org0avM6vvtKVjcm7s9Ivx
tSoi/7ZybDguhp6oBdskc/wsD/F9AOWqp5InFBbyka6pasM+eaxYSftm76TVDF9g
P2Q77V3qBQKBgQC98NEgKrpJFp9rAtBpOpB/JSIkurXVOzh250wA7PHkEkNReiM+
ptzAdSVy9jGPV+AZin0XB6g07aefel+xZDgLISdguckgP98HRTxwMy3REEAbRZp1
GC0s6UWEm1aj6mBEWyUAZIYZ1D4phoUyErGYtli90V6LTiNkG1vZSYW4kQKBgDVy
mLZ9WlwRq3lkAy0Rjh0BRx3p5NPPutkr19fmM9KvfPMURsMneeqh+C8U37xMhUGF
KvD4bbz7hU0E/hT42GsfzNY6v9eupQDslHjLMsnvpHIoAKPrFxSPvpQ0hkvCjaAW
pD5KvJBePQzzMh62M0CI7FZYwyCMGH2zJSonsEh9AoGBAMZOTXeEhYDmHK9sKRb9
pMPdUKW2EXScaOepjYkLVEp7AwoqdvWoG8AMXz5MK+oW5zP4ZDKhvcYLDjAUi1P
TR+BABU0YmMYoG+W4/K9m/xXvPeDN9J3dqtGpmYcappU0ZOA0e7YafAfqnevOz50
HiL42fRbmYypIICXcmgsjE4X
-----END PRIVATE KEY-----
```

Here are the two prime numbers that were used to generate the modulus:

```
prime1:          (metens㉿ cs591-metens-kali-1)
    00:e9:44:bf:e4:f6:48:6f:fd:bc:f6:2a:b1:53:50:
    46:95:d0:ea:9c:fd:31:07:0c:22:b4:2e:e9:6a:68:
    49:4e:85:a1:08:55:c2:4a:c5:33:bf:d7:59:04:a4:
    5b:22:c2:13:14:a3:fb:7a:8b:58:bb:8a:ee:46:5b:
    38:6a:18:6a:4d:c2:6a:5a:b1:50:9f:63:6a:21:4b:
    4b:cd:a5:74:fa:23:5a:08:47:55:6c:4b:c4:d8:7b:
    83:08:01:83:27:2b:68:04:6e:09:3a:48:c1:e0:9e:
    3b:6f:cc:56:50:97:d7:38:79:0d:0c:9e:05:db:1a:
    57:17:1d:e0:83:95:d7:85:75
prime2:
    00:e1:b5:d6:48:84:60:8e:b9:1c:0b:46:54:d6:fc:
    9a:d6:ff:64:43:e9:f0:d1:a3:a2:40:8b:1c:38:62:
    3e:bb:6a:46:f1:17:67:b3:4c:47:3b:2b:3f:ce:98:
    80:6c:6a:53:45:a2:df:28:af:5c:fa:4b:f3:ab:83:
    46:af:33:ab:ef:b4:a5:63:72:6e:ec:f4:8b:f1:b5:
    2a:22:ff:b6:72:6c:38:2e:86:9e:a8:05:db:24:73:
    fc:2c:0f:f1:7d:00:e5:aa:a7:92:27:14:16:f2:91:
    ae:a9:6a:c3:3e:79:ac:58:49:fb:66:ef:a4:d5:0c:
    5f:60:3f:64:3b:ed:5d:ea:05
```

```
modulus:         (metens㉿ cs591-metens-kali-1)
    00:cd:ab:1d:85:75:b6:98:b1:d6:ff:bd:e7:58:46:
    e3:d4:67:00:00:06:65:37:67:ac:d9:5d:b2:9c:5c:
    27:a6:1b:c7:0b:88:c9:a9:9f:b9:92:a4:fe:c9:85:
    f8:86:1e:50:15:25:ff:88:20:6a:df:11:6b:ec:28:
    ce:be:4f:6c:a0:fd:d8:1f:7b:88:c3:f0:19:dd:38:
    78:35:3a:74:12:b5:f4:45:81:f2:e9:3a:fa:1b:1f:
    ca:1b:ed:aa:e7:05:07:49:f7:0c:93:31:3a:99:11:
    37:71:2f:1a:0e:33:21:37:20:a1:ae:da:5e:6f:ab:
    bd:94:ef:d6:a7:2b:87:05:33:2a:cc:52:38:1e:e8:
    47:ef:5f:ba:e6:d6:85:e7:fc:88:ac:8a:96:53:5c:
    89:71:21:b7:9f:b1:6a:8c:46:c1:65:fc:f8:59:41:
    31:f2:35:d4:ae:2a:09:2d:c9:5e:02:84:56:53:49:
    17:c6:75:e3:3b:59:c2:d8:65:45:8b:43:6b:04:02:
    35:ae:c3:94:01:1c:29:88:bb:59:52:56:4c:3f:f0:
    eb:d8:ba:f1:f2:66:fb:ef:1d:02:8a:9f:f2:6c:bd:
    59:32:e7:97:46:f1:14:eb:38:3e:97:25:35:40:3d:
    e7:4d:b0:13:f7:e1:23:06:e5:94:ec:15:58:b9:b3:
    8d:49
publicExponent: 65537 (0x10001)
```



Bob’s public key (the modulus) is 2048 bits. Now we do the same process for Alice…

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ cd ../Alice;

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ openssl genpkey -algorithm rsa -out AlicePrivKey.pem;
...+..+...+++++++++++++++++++++++++++++++++++++++*.+.......+......+.+........+......
+.+...+..+...+.......+...+..................+..+.......+++++++++++++++++++++++++++++
+++++++++*..........+.....+.....................+.+......+..+.+.............+..+..
.......+.+.....++++++
..+......++++++++++++++++++++++++++++++++++++++*.....+...+.......+++++++++++++++++++
++++++++++++++++++*........+............+.....+.+........+......+.+..............
...+..+..............+...+.+......+........+.+..+....+............+.+...+........
.........+...+..+...+....+...+.....+.........+....+.........+..+.+...+......+........
+...+....+.........+......+...+...+...+.....+.........+.+..+...+.+...+..+......+....+
.....+.........+....+...+..+.+..+.......+..........+....+...........+...+..+...+.+.
..........+............+..+...+...+....+.....+.+...........+..+.+..+..........+....
.+.+...........+..+...........+.+..+......+.......+...+......+..........+.......+.
.....+.....+......+....+..+.......+...........+....+.....+.......................
+...+.+..........+.........+....+...+........+...........+....+........+.+.....+...
.......+..+....++++++

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ openssl pkey -in AlicePrivKey.pem -pubout -out ../AlicePubKey.pem;

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ cat ../AlicePubKey.pem;
-----BEGIN PUBLIC KEY-----
MIIBIjANBgkqhkiG9w0BAQEFAAOCAQ8AMIIBCgKCAQEA29CUZuGz/H48Iy4cr95X
Lc2XxnaYAPqrkkht+rwJcVVc7LZE3E7dgvzrj3+YBOqp+t9QFuBDILgoOrJ8KrJe
4ERThMZu+dhBm8nNo9n0ITeMtgSaUWUkZefD6zz1pm8pRqBvJijvwbrk59ZJCKHh
mZgaIFmOHFFb63gXz2efrIRDysiuzF/w1UWa5ESV/OAH0YUU7ZXZ3A2f2fU0UIbH
kjlxgbUJlHowrF2F0f5kVuHOlYUdlR9zJbHDRzcA+twaYMZdKOmJCViQHiUhiwtl
QSqFDIRGjuhUgD7D0M8iFN+yIYJ0KXgWV5rtgZwunMZPnHjX5wuc6ukZc4ZaiCwh
1wIDAQAB
-----END PUBLIC KEY-----
```

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ openssl pkey -in ../AlicePubKey.pem -pubin -text -noout
Public-Key: (2048 bit)
Modulus:
    00:db:d0:94:66:e1:b3:fc:7e:3c:23:2e:1c:af:de:
    57:2d:cd:97:c6:76:98:00:fa:ab:92:48:6d:fa:bc:
    09:71:55:5c:ec:b6:44:dc:4e:dd:82:fc:eb:8f:7f:
    98:04:ea:a9:fa:df:50:16:e0:43:20:b8:28:3a:b2:
    7c:2a:b2:5e:e0:44:53:84:c6:6e:f9:d8:41:9b:c9:
    cd:a3:d9:f4:21:37:8c:b6:04:9a:51:65:24:65:e7:
    c3:eb:3c:f5:a6:6f:29:46:a0:6f:26:28:ef:c1:ba:
    e4:e7:d6:49:08:a1:e1:99:98:1a:20:59:8e:1c:51:
    5b:eb:78:17:cf:67:9f:ac:84:43:ca:c8:ae:cc:5f:
    f0:d5:45:9a:e4:44:95:fc:e0:07:d1:85:14:ed:95:
    d9:dc:0d:9f:d9:f5:34:50:86:c7:92:39:71:81:b5:
    09:94:7a:30:ac:5d:85:d1:fe:64:56:e1:ce:95:85:
    1d:95:1f:73:25:b1:c3:47:37:00:fa:dc:1a:60:c6:
    5d:28:e9:89:09:58:90:1e:25:21:8b:0b:65:41:2a:
    85:0c:84:46:8e:e8:54:80:3e:c3:d0:cf:22:14:df:
    b2:21:82:74:29:78:16:57:9a:ed:81:9c:2e:9c:c6:
    4f:9c:78:d7:e7:0b:9c:ea:e9:19:73:86:5a:88:2c:
    21:d7
Exponent: 65537 (0x10001)
```

Alice’s modulus (public key) is also 2048 bits.

Task 6: Encryption

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ ls
OWNED by metens
Alice  AlicePubKey.pem  Bob  BobPubKey.pem


┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ cd Alice

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ ls
OWNED by metens
AlicePrivDHKey.pem  AlicePrivKey.pem

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ python3 -c 'print(512*"A")' > file.txt;

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ openssl pkeyutl -encrypt -in file.txt -pubin -inkey \
../BobPubKey.pem -out ../file_BobPubKey.bin
Public Key operation error
A09CD69BFFFF0000:error:0200006E:rsa routines:ossl_rsa_padding_add
_PKCS1_type_2_ex:data too large for key size:../crypto/rsa/rsa_pk
1.c:133:
```

Due to the large size of 512 bit As in the file.txt (512 / 8 = 64 bytes), the encryption failed. The error says: “data too large for key size”.

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ cd ../Bob;

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ openssl pkeyutl -decrypt -in ../file_BobPubKey.bin \
-inkey BobPrivKey.pem -out decrypt.txt;
diff decrypt.txt ../Alice/file.txt

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ ls
OWNED by metens
BobPrivDHKey.pem  BobPrivKey.pem  decrypt.txt

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ cat decrypt.txt
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$
```

The decrypted message is the same as the original:

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ cat decrypt.txt
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ cat ../Alice/file.txt
AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ diff decrypt.txt ../Alice/file.txt
```

Since the total size of the encrypted message is limited and it is slow, hard to accelerate, and good for only small amounts of data, we need to find a way to send large amounts of data back and forth. We can use asymmetric + symmetric together. Combining both asymmetric and symmetric encryption allows us to use asymmetric encryption to establish a secret symmetric encryption key and then use the symmetric encryption key for large-scale data transfer.

Task 7: Combining symmetric and asymmetric encryption

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ cd Alice;

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ python3 -c 'print(512*"A")' > file.txt;
openssl rand 32 > sk.bin;
openssl enc -aes-256-cbc -in file.txt -out ../file_sk.bin \
-pass file:sk.bin

*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ ls
OWNED by metens
AlicePrivDHKey.pem  AlicePrivKey.pem  file.txt  sk.bin

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ openssl pkeyutl -encrypt -in sk.bin -pubin -inkey \
../BobPubKey.pem -out ../sk_BobPubKey.bin

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ ls
OWNED by metens
AlicePrivDHKey.pem  AlicePrivKey.pem  file.txt  sk.bin

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ ls ../
OWNED by metens
Alice            Bob            file_sk.bin
AlicePubKey.pem  BobPubKey.pem  sk_BobPubKey.bin

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$
```

Task 8: Bob

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Alice]
└─$ cd ../Bob ;
openssl pkeyutl -decrypt -in ../sk_BobPubKey.bin \
-inkey BobPrivKey.pem -out sk.bin

┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2/Bob]
└─$ openssl enc -d -aes-256-cbc -in ../file_sk.bin -out decrypt.txt
 \
-pass file:sk.bin ;
diff decrypt.txt ../Alice/file.txt
*** WARNING : deprecated key derivation used.
Using -iter or -pbkdf2 would be better.
```

Alice has her private key, which is different from Bob's private key:

```
┌──(metens㉿cs591-metens-kali-1)-[~/Lab-3.2]
└─$ diff Bob/BobPrivKey.pem Alice/AlicePrivKey.pem
2,27c2,27
< MIIEvgIBADANBgkqhkiG9w0BAQEFAASCBKgwggSkAgEAAoIBAQDNqx2FdbaYsdb/
< vedYRuPUZwAABmU3Z6zZXbKcXCemG8cLiMmpn7mSpP7JhfiGHlAVJf+IIGrfEWvs
< KM6+T2yg/dgfe4jD8BndOHg1OnQStfRFgfLpOvobH8ob7arnBQdJ9wyTMTqZETdx
< LxoOMyE3IKGu2l5vq72U79anK4cFMyrMUjge6EfvX7rm1oXn/IisipZTXIlxIbef
< sWqMRsFl/PhZQTHyNdSuKgktyV4ChFZTSRfGdeM7WcLYZUWLQ2sEAjWuw5QBHCmI
< u1lSVkw/8OvYuvHyZvvvHQKKn/JsvVky55dG8RTrOD6XJTVAPedNsBP34SMG5ZTs
< FVi5s41JAgMBAAECggEAB3uo3NC+Hosl7/MoZjFaG8HNobKZ3yyyAMykcYT22lfb
< Y6OL9tR9a4T+UjXCJ5sVvdc+l2TajpgzLYoEj68THD5wLSERBlQHlOF3mjQwwfeF
< 1juWq36+V9tNvrtkgvhPXhUTi414fNZABqm4kHiyxi81QsfmNLjycmdRj46JYJp9
< yoNFfdlhoCFiwRMeKiQP3PD9Nl1iTf7Chpm78GQwh6bW6Z8kf5I81j230TbN1kOM
< BqXELsuEu0mSvGzwAjJyHFU8HT49yx08J6Qi3V4YenHPs38t1Co5X/KNbyIwvC5z
< 6ZH6yjQsvBRpTrfNyZh1CqFzuDmX72k3cZlMYcuv6QKBgQDpRL/k9khv/bz2KrFT
< UEaV0Oqc/TEHDCK0LulqaElOhaEIVcJKxTO/11kEpFsiwhMUo/t6i1i7iu5GWzhq
< GGpNwmpasVCfY2ohS0vNpXT6I1oIR1VsS8TYe4MIAYMnK2gEbgk6SMHgnjtvzFZQ
< l9c4eQ0MngXbGlcXHeCDldeFdQKBgQDhtdZIhGCOuRwLRlTW/JrW/2RD6fDRo6JA
< ixw4Yj67akbxF2ezTEc7Kz/OmIBsalNFot8or1z6S/Org0avM6vvtKVjcm7s9Ivx
< tSoi/7ZybDguhp6oBdskc/wsD/F9AOWqp5InFBbyka6pasM+eaxYSftm76TVDF9g
< P2Q77V3qBQKBgQC98NEgKrpJFp9rAtBpOpB/JSIkurXVOzh250wA7PHkEkNReiM+
< ptzAdSVy9jGPV+AZin0XB6g07aefel+xZDgLISdguckgP98HRTxwMy3REEAbRZp1
< GC0s6UWEm1aj6mBEWyUAZIYZ1D4phoUyErGYtli90V6LTiNkG1vZSYW4kQKBgDVy
< mLZ9WlwRq3lkAy0Rjh0BRx3p5NPPutkr19fmM9KvfPMURsMneeqh+C8U37xMhUGF
< KvD4bbz7hU0E/hT42GsfzNY6v9eupQDslHjLMsnvpHIoAKPrFxSPvpQ0hkvCjaAW
< pD5KvJBePQzzMh62M0CI7FZYwyCMGH2zJSonsEh9AoGBAMZOTXeEhYDmHK9sKRb9
< pMPdUKW2EXScaOepjYkLVEp7Awoqdv/WoG8AMXz5MK+oW5zP4ZDKhvcYLDjAUi1P
< TR+BABU0YmMYoG+W4/K9m/xXvPeDN9J3dqtGpmYcappU0ZOA0e7YafAfqnevOz50
< HiL42fRbmYypIICXcmgsjE4X
---
> MIIEvAIBADANBgkqhkiG9w0BAQEFAASCBKYwggSiAgEAAoIBAQDb0JRm4bP8fjwj
> Lhyv3lctzZfGdpgA+quSSG36vAlxVVzstkTcTt2C/OuPf5gE6qn631AW4EMguCg6
> snwqsl7gRFOExm752EGbyc2j2fQhN4y2BJpRZSRl58PrPPWmbylGoG8mKO/BuuTn
> 1kkIoeGZmBogWY4cUVvreBfPZ5+shEPKyK7MX/DVRZrkRJX84AfRhRTtldncDZ/Z
> 9TRQhseSOXGBtQmUejCsXYXR/mRW4c6VhR2VH3MlscNHNwD63Bpgxl0o6YkJWJAe
```

Someone else could also make a private key. Then, the file.txt could be created with any message. George could create a message, claiming to be Alice, create his own symmetric key, encrypt the message with the symmetric key, take Bob's public key, and generate the encrypted

symmetric key which is publicly available. Bob would take the encrypted symmetric key, use his private key and decrypt the message sent from George claiming to be Alice. Bob then decrypts the file and sees a message from “Alice” (George) and will send a message back to “Alice”. So in conclusion, Bob does NOT know if Alice sent the message, or if someone else impersonating Alice sent the message.