Hello,

I am currently stuck on challenge 7. I believe I have a solid understanding of how the packet is sent out from the malware. Below is a dump of the packet.

```
e2e4d0ce15ad:~# xxd 18_02_09_AM.log
00000000: 56c2 e697 33cb 81b1 f0f0 7db4 6d99 42fb  V...3.....}.m.B.
00000010: 7ed7 afc8 91c6 a2c2 4729 8708 ff3a b83d  ~.......G)...:.=
00000020: 1b3f e548 4141 4141 4141 4141 4141 4141  .?.HAAAAAAAAAAAA
00000030: 4141 4141 4141 4141 4141 4141 d580 f64f  AAAAAAAAAAAA...O
00000040: c641 6ede 9bd1 8c1c 0c7e 7a02 4678 2631  .An......~z.Fx&1
00000050: 15bb 06f3 a185 ca69 89af 4846 12fc 2410  .......i..HF..$.
00000060: a815 7d48 196d df3a 9c3b 8956 fef3 ba1a  ..}H.m.:.;.V....
00000070: 6827 cc5c 2cf8 442e a78b e204 fcb1 be01  h'.\,.D.........
00000080: e367 6675 c85d 19c9 1a7f e4cb ccec 1f06  .gfu.]..........
00000090: 605f d02d 6f54 bb89 f82b c144 e8cc 46b9  `_.-oT...+.D..F.
000000a0: 3010 1c68 974f 5fff b494 21dd f522 551e  0..h.O_...!.."U.
000000b0: eee0 f792 900f 061a db43 76e7 2d5f 56ed  .........Cv.-_V.
000000c0: bda1 f51d 923e 275d 1518 f55f 3ca3 889a  .....>']..._<...
000000d0: 2286 1728 872d 16ea 0c37 6fbb ac3d 7d83  "..(.-...7o..=}.
000000e0: 2cdf 0e59 7cfa 7bff 91e6 8ae8 1234 e69c  ,..Y|.{......4..
000000f0: 5c94 deb2 3527 c3d2 61                   \...5'..a
```

I know that the first 0×20 are public key of the malware. Then there are 0×4 which is the header length of the message. Next is the 0×20 Nonce which I have forced to be all A in this case.

In gdb I have the following breakpoints:

```
# This forces the nonce to all A's to make packet readable
break *ozzumyndvjsex+205
  commands
```

```
    p (void *) memcpy(nonce->_M_dataplus->_M_p,
"AAAAAAAAAAAAAAAAAAAAAAAA", 0x18)
    continue
  end
# This catches right after the server pub key is returned and forces
it to be a public key I have the private key for
# private key is all 'K' in this case
break *ozzumyndvjsex+118
  commands
    p (void *) memcpy(pubKey,
"\x95\xf4\xea\xfe\x1f\xe1\xb6\x95\xe4\xbd'\xfc88\xc5'\xb1\x94Z\xfcE^
\xe2\xb0\xe0\xf8\x1b\xdf\xfb\xad\xce$", 34)
    continue
  end
# set up netcat listener on localhost to get the traffic
# I changed the docker image IP to the one the malware calls home to
break *main+1
  commands
    !nc -lp 6666 > "$(date +"%d_%I_%M_%p").log" &
    continue
  end
# Break right before call so I can examine locals
break *ozzumyndvjsex(int, std::__cxx11::basic_string<char,
std::char_traits<char>, std::allocator<char> >)+468
```

Once I get the packet capture from netcat I put it in the following python script

```
import nacl.utils
from nacl.public import PrivateKey, Box, PublicKey, SealedBox



priv_key = b'K'*32
```

```
logFile = '/home/user/Documents/NSA-
codebreaker/task_6/dynamic_anal/shared/18_03_00_AM.log'

capture = open(logFile, "rb")

# get the bits before the nonce
Ox24 = capture.read(0x24)
# extract the malwares public key
client_pub = Ox24[:0x20]

# read out the rest of the message 0x100 is an arbitrally high
number
message = capture.read(0x100)

# create the public and private keys to decrypt
serverPrivate = PrivateKey(priv_key)
cleintPublic = PublicKey(client_pub)
# create the box in python to decrypt
bob_box = Box(serverPrivate, cleintPublic) #SealedBox(priv_key)

# print out the decrypted message, this will have to be changed to
submitt
print(bob_box.decrypt(message))
```

When running the code it fails to properly decrypt the cipher text and returns this error.

```
Traceback (most recent call last):
  File "/home/user/PycharmProjects/pythonProject/main.py", line 23,
in <module>
    print(bob_box.decrypt(message))
  File
"/home/user/PycharmProjects/pythonProject/venv/lib/python3.9/site-
```

```
packages/nacl/public.py", line 260, in decrypt
    plaintext = nacl.bindings.crypto_box_open_afternm(
  File
"/home/user/PycharmProjects/pythonProject/venv/lib/python3.9/site-
packages/nacl/bindings/crypto_box.py", line 228, in
crypto_box_open_afternm
    ensure(res == 0, "An error occurred trying to decrypt the
message",
  File
"/home/user/PycharmProjects/pythonProject/venv/lib/python3.9/site-
packages/nacl/exceptions.py", line 81, in ensure
    raise raising(*args)
nacl.exceptions.CryptoError: An error occurred trying to decrypt the
message

Process finished with exit code 1
```

This seems to imply that I do not have the proper keys, however I know I have the correct public key from examine it in GDB.

I do have a good idea of what is in the cypher text. For example, from gdb, I know that the first chunk of the message sent is output of the fingerprint function. On my system it is

`dXNlcm5hbWU9dW5rbm93bg==,dmVyc2lvbj0yLjAuMi4wLVVRVA==,b3M9TGludXg=,dGltZXN0YW1wPTE2MzY5OTEzNjQ=`

which decodes to:

username=unknownversion=2.0.2.0-UQTos=Linuxtimestamp=1636991364

I am very confused why this is not working properly and beginning to suspect that I may be going about this very wrong because I do not see anywhere that there is a uuid for the victim. I am hoping that once this decrypts it becomes obvious, but I don't want to spend a massive amount (more) time on this if I am going off on a wrong trail.

Thank you a ton for the help and these challenges. They have been a blast so far and I have learned a ton.

- Nicholas (hourglass on discord)