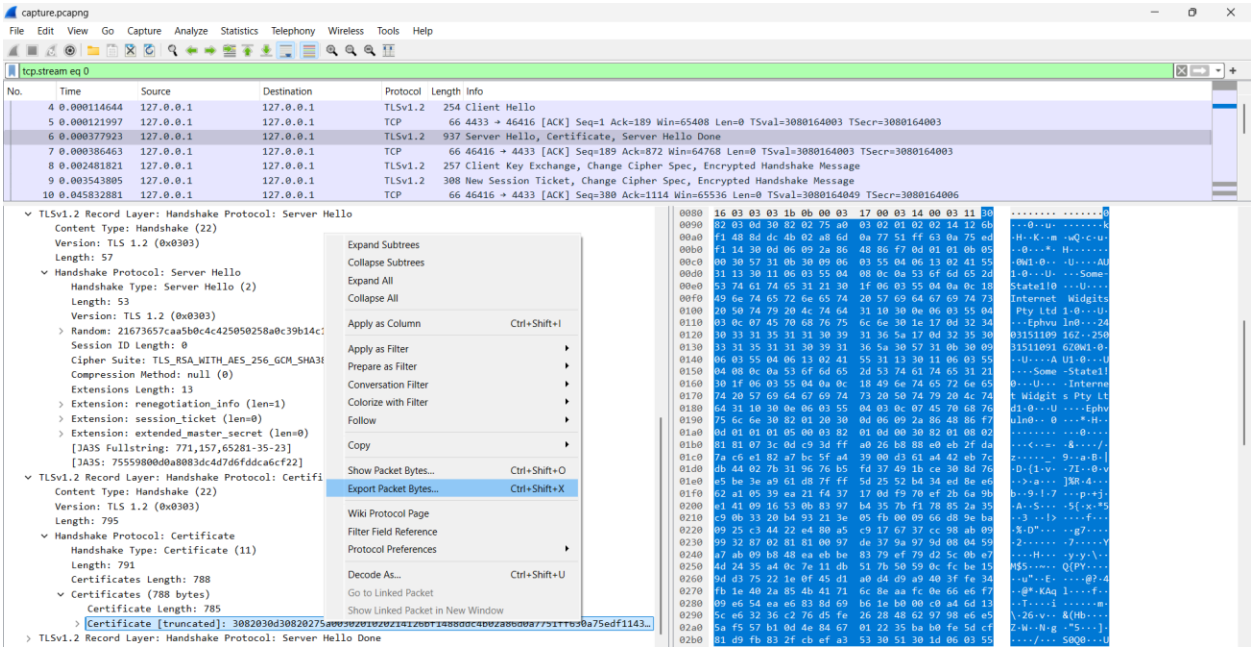
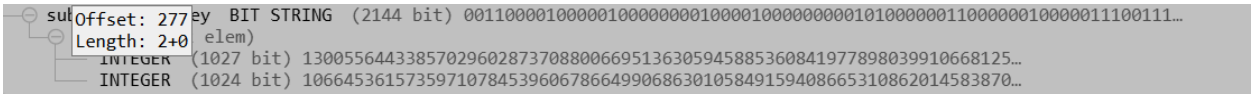
So we got a Network-Cryptography chall that gives us a pcap with TLS 1.2…. what can we do…. Wait did you say TLS 1.2??

Unlike TLS 1.3, TLS 1.2 can be broken given the cipher suite is weak enough, which in this case it is. First, we need to extract the certificate:



Next up, we need to decode the der certificate. For that I used https://lapo.it/asn1js/, however openssl and other websites can be used as well. What we need are the public key values n and e (since this is RSA)

For me, said values can be found here:



Next up, we use dcode.fr to factorize n and figure out the other values here: https://www.dcode.fr/rsa-cipher

Next, we need to build the private key. Using python and the values we got from dcode we can do this easily.

```
from cryptography.hazmat.primitives.asymmetric import rsa
from cryptography.hazmat.primitives import serialization

n =
130055644338570296028737088006695136305945885360841977898039910668125879751574486
764906816923410482739205155657114240527768168203053997895679015196247676419703889
538999903490602678285554393578343399759692191214661836733928884019109486359016675
675039757184019867766890822754507772195761374052546965228229290867 9
```

```
e =
106645361573597107845396067866499068630105849159408665310862014583870062061704662
230754284832387896920427209753236862548800746662398609212688373613186979102970308
417884832531601035544107102590028211579550508699494971288803583755640940424098301
425895738898909222425910339731329121362635050810847489912118168559
d = 183375761153070104774226049013299425645851327487604585722804806129075187793 99
p =
256203676801321105558790751640928140809926277076644026861746675449387778232279417
467429462291409906746865989861534939632608465093981738544910139327703796 99
q =
507625987114247645783098351610285365050170627795357285981937390214229913840750976
193648379204737394771927419934677465309372731785688309674653044959155650 21
dp = d % (p - 1)
dq = d % (q - 1)
qi = pow(q, -1, p)
private_numbers = rsa.RSAPrivateNumbers(
    p=p, q=q, d=d, dmp1=dp, dmq1=dq, iqmp=qi,
    public_numbers=rsa.RSAPublicNumbers(e, n)
)
private_key = private_numbers.private_key()
with open("private_key.pem", "wb") as f:
    f.write(
        private_key.private_bytes(
            encoding=serialization.Encoding.PEM,
            format=serialization.PrivateFormat.PKCS8,
            encryption_algorithm=serialization.NoEncryption(),
        )
    )

print("Private key saved as private_key.pem")
```

After doing this we got our private key. All that's left is to import it (Edit -> Preferences -> Protocols -> TLS -> RSA keys, select the file we got and save everything). Now the pcap should have refreshed. To get the flag we simply need to right click a TLS packet and Follow -> TLS stream.

Made with love by: AndreiCat