1. Key ExchangeIn this project, our group will use Diffie-Hellman as the key exchange method. According to RFC 3526, “g” is set to the fixed number 2, “p” is set to the fixed large prime showed in Figure1.

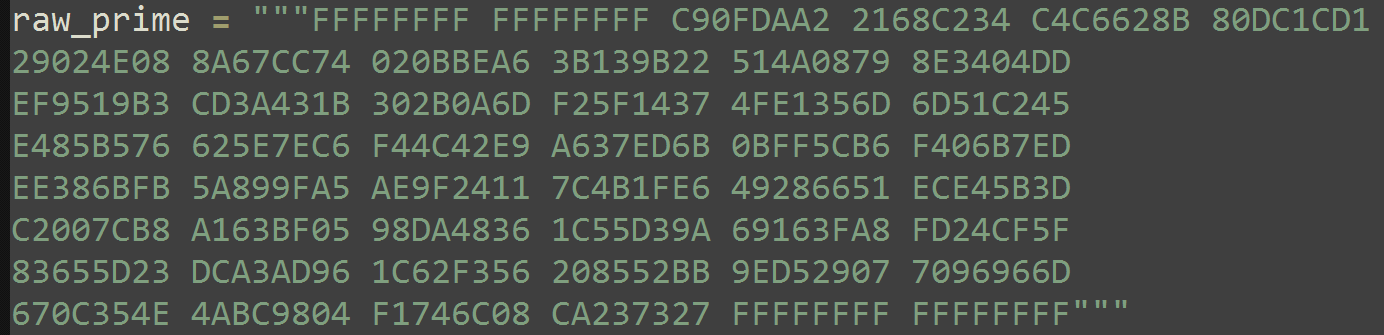


Figure1

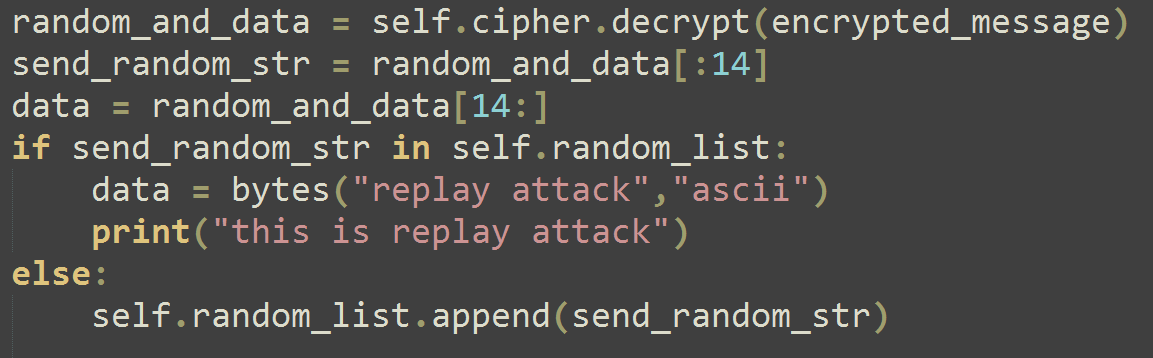
Alice and Bob will pick the random private key “a” and “b” based on this large prime with method “randint(1,prime-1)” from Crypto.Random.random and generate and . Method “create\_dh\_key()” from dh. \_\_init\_\_.py will return and a.

In method “initiate\_session(self)” from lib.comms.py, the instance of StealthConn will call method “create\_dh\_key()” to get this instance’s (denoted as my\_public\_key) and a(my\_private\_key). Then this instance will send to its connecting-peer instance with method “send(self, data)” from lib.comms.py and also receive its connecting-peer instance’s (their\_public\_key) with method “recv (self)” from lib.comms.py. With “my\_private\_key” and “their\_public\_key”, this instance will call calculate\_dh\_secret(their\_public, my\_private) from dh. \_\_init\_\_.py to get the shared hash which will be used in the whole project.

4. Preventing Replay

To prevent replay attack, our group append 14 random bytes to the start of each message. After receiving the message, these 14 random bytes will be take out and stored in “random\_list” in the instance of StealthConn. If the instance received another message with 14 random bytes that have been already stored in the list, this message will be seen as a replay attack and the connection will be cut off.

These random bytes are generated by method “get\_random\_bytes(14)” from Crypto.Random , and are appended during the method “send(self, data)” from lib.comms.py. In method “recv(self)” from lib.comms.py, every time receiving a message, the 14 random bytes will be take out and will be checked with the following code.



5. Special Questions

① Why might we want to allow for peer-to-peer file transfers between bots?

Because even some of the peers are down, the other peers can still have the whole function of this project. Also peer-to-peer file transfer is easy to configured.

② What are the advantages and disadvantages to using a central web server (pastebot.net in our case, similar to pastebin.com) to distribute files when controlling a botnet?

Advantages: Easy to control. The owner only need to upload the files to the central web server.

Disadvantages: Make the whole system easy to be attacked. If the central web server is broken by the attacker, the whole system is down.