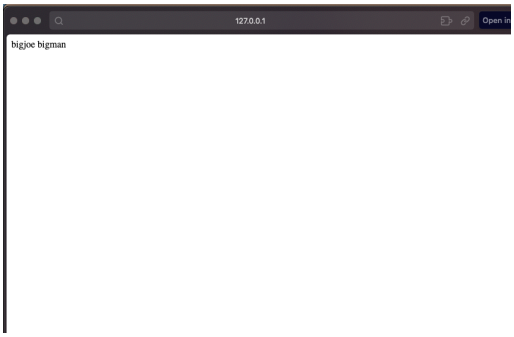
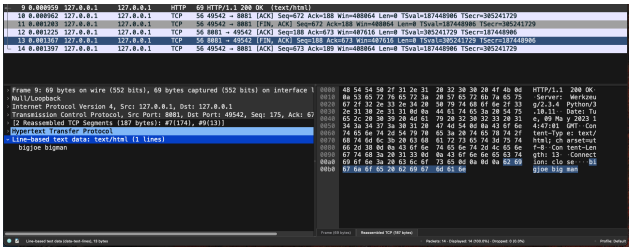
RS40 TP2 / yann derré

PART I





On peut voir dans wireshark les échanges fait avec le server sur localhost. La requête GET vers le server pour demander la page, et le retour non chiffré de la page, contenant notre mot de passe ici changé par “bigjoe bigman”. On inspecte le packet retourné par le serveur nous donnant un status 200 OK. Le reste des packets échangés sont des packets d’acknowledgement (handshake) de fin de transaction entre le client et le serveur.

PART II

On configure nos paramètres dans les fonctions correspondantes

Voici la configuration de mon CA:

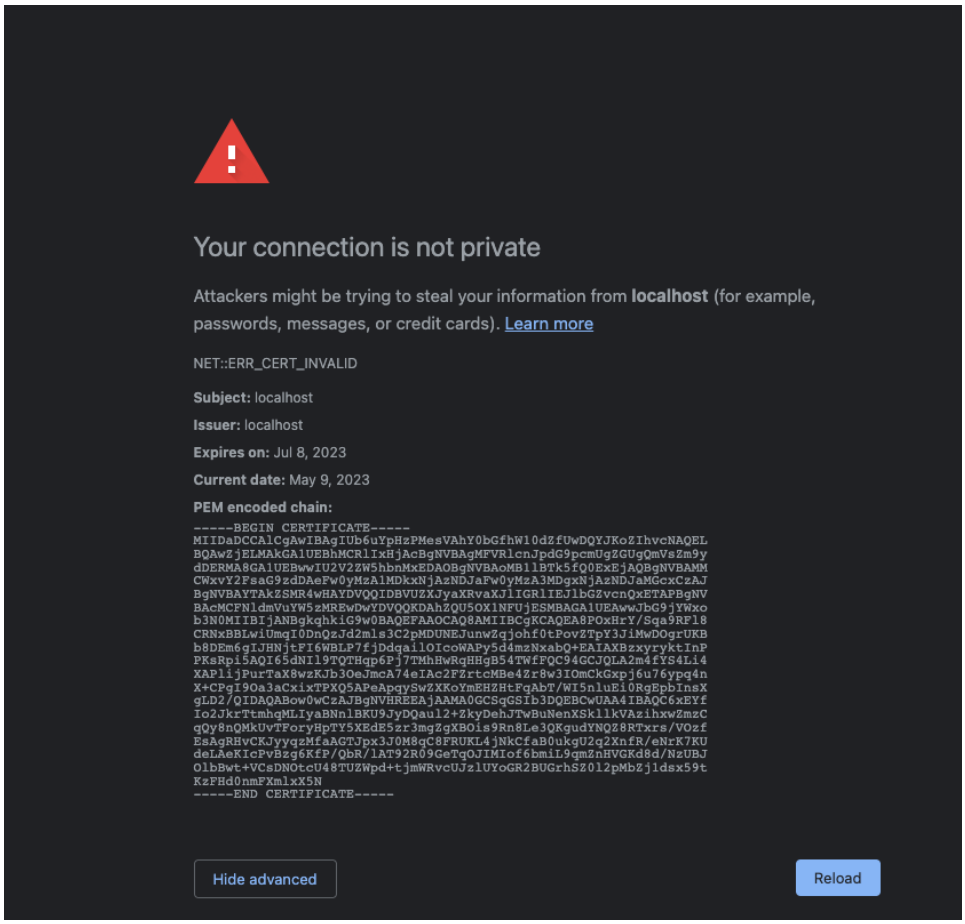
```
2-rs40/build.py
CA PUBLIC KEY:

<Name(C=FR,ST=Territoire de Belfort,L=Sevenans,O=YANN_CA,CN=localhost
-----BEGIN CERTIFICATE-----
MIIDbTCCAlWgAwIBAgIUTyxEaYIywBC111wA0qysbZbCq8wwDQYJKoZIhvcNAQEL
BQAwZjELMAkGA1UEBhMCRlIxHjAcBgNVBAgMFVRlcnJpdG9pcmUgZGUgQmVsZm9y
dDERMA8GA1UEBwwIU2V2ZW5hbnMxEDAOBgNVBAoMB1lBTk5fQ0ExEjAQBgNVBAMM
CWxvY2FsaG9zdDAeFw0yMzA1MDkxNjAzNDJaFw0yMzA3MDgxNjAzNDJaMGYxCzAJ
BgNVBAYTAkZSMR4wHAYDVQQIDBVUZXJyaXRvaXJlIGRlIEJlbGZvcnQxETAPBgNV
BAcMCFNldmVuYW5zMRAwDgYDVQQKDAdZQU5OX0NBMRIwEAYDVQQDDAlsb2NhbGhv
c3QwggEiMA0GCSqGSIb3DQEBAQUAA4IBDwAwggEKAoIBAQC69McVBvVoW7hAjT4W
rodOU0DYKbkxcyuOziC6DALbVund8h+UAyvd2hBDjnHn7fKKxk9PnDz4Mi2RKcAE
/itafB20Il7CxS/sYQjr4zP+3AMV4fTE3tyC9k8oex+DrYR0xd4JIjnd4HrfpPxW
Xrb5IlEswwzlXvdMNtlw/cCeBkdOQNPlaQ5Sx9jKV4tl5ym0UvX3E45ur/hoQ9O4
K9pUVTDBAT/syVaTaDCgE8DqktvDSi+v6h9joffnul1hD0kyz8zurcMLsJbYNiSn
CSp1oG3SfKj/1VcKQVnVzi7Lb4vWdIoKvlhhu8Ju8vyVqcJytvgh7F+bS42OaKSv
WbmNAgMBAAGjEzARMA8GA1UdEwEB/wQFMAMBAf8wDQYJKoZIhvcNAQELBQADggEB
ABjNMmDbtIjs9wnicgX5s5KXGbtfSYyNLK1BiXLssohXyX9sPf6gOpcOuhOZFNvC
0FcBTWiZ9xgTpZa107uS/2LgN4r8FBWH3LsDNPSNCLnRFtkIVkmFAbb4mB8b3TLE
inSnsvxFNmntAuaip3KTa6kGQg3vKSuRKen34cAy0w/P6QJsje02IWMkHzTaezjC
e8zOcu9Yja5k0TwD18RcLu17hctwMd7COPqz6txZDELp/Z8mklk5ZPUeI+1pP7Oo
NSyjTFw5IURrQXylD18KwNJ6xgMiEQVzOr8VA1gt15CmxudzpzMW7BUxkPEgfhQ7
Ck2Y2d3w81qPncMbwffQkx4=
-----END CERTIFICATE-----

finished ...
```

```
CA_PASSWORD = "boss"
SERVER_PASSWORD = "boss"

CA_CONFIGURATION = Configuration("FR", "Territoire de Belfort", "Sevenans", "YANN_CA", "localhost")
SERVER_CONFIGURATION = Configuration("FR", "Territoire de Belfort", "Sevenans", "YANN_SER", "localhost")
```

J’ai modifié la fonction print_pem pour déchiffrer le CA

```python
def print_perms(filename: str):
    pem_data = pem.parse_file(filename)

    cert = x509.load_pem_x509_certificate(pem_data[0].as_bytes(), default_backend())
    #Print the subject of the certificate
    print(cert.subject)
    #print the content of the certificate
    print(pem_data[0].as_text())
```

Le navigateur nous affiche un message d’erreur de certificat invalide. Notre autorité crée n’est pas connue par le navigateur et ne peut donc pas vérifier le certificat.



Une manière de contourner cette limitation est d’obtenir un nom de domaine et d’un certificat auprès d’une vraie CA reconnue. Il n’est pas possible de générer un certificat pour localhost.

PART III

```
def login():
    if request.method == 'POST':
        username = request.form['username']
        password = request.form['password']

        if username == USER1_LOGIN and password == USER1_PASSWORD:
            return ("Logged in successfully! \n" + SECRET_MESSAGE)
        else:
            return 'Wrong username or password!'

    return '<form action="" method="post">\n <p><input type=text name=username
```

```
app = Flask(__name__)
USER1_LOGIN = "bigjoe"
USER1_PASSWORD = "boss"
```

J’envoie un form à l’utilisateur. Le serveur compare avec ses valeurs. On peut ensuite valider et envoyer le message secret en retour. !!! NE FONCTIONNE PAS SANS HTTPS, la requête POST avec les identifiants utilisateurs seront en clair.