

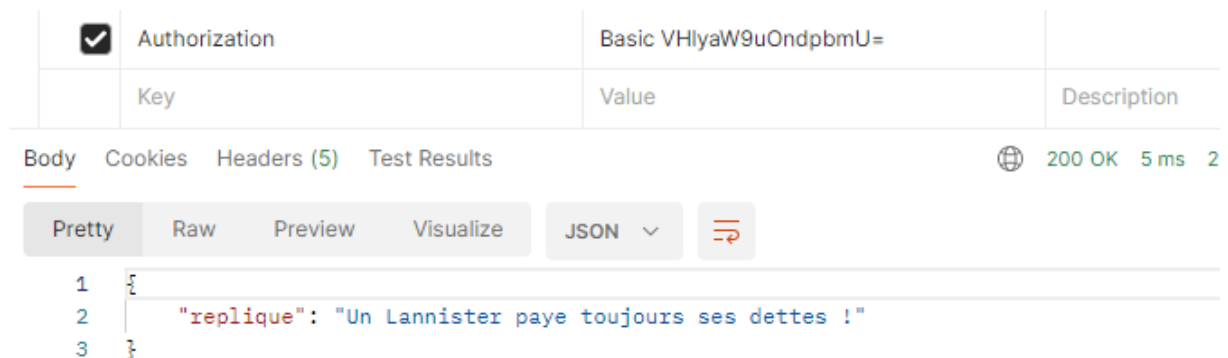
TP4

Etape 1 :

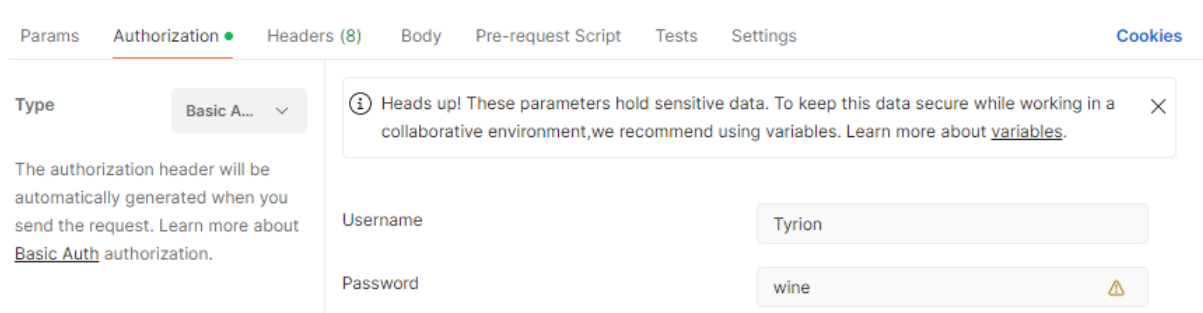
Lorsqu'on essaye d'accéder au endpoint : <http://localhost:3000/secu>

Le message d'erreur s'affiche car on n'est pas authentifié.

Contrairement à <http://localhost:3000/dmz> car il ne demande pas d'authentification.



Nous pouvons maintenant accéder au endpoint « secu » grâce à l'authentification en base64.



Il y a aussi un onglet authorization pour rentrer les valeurs en dur et cela permettra d'avoir automatiquement le header avec les bonnes valeurs.

Lorsqu'on modifie les valeurs en dur, cela modifie directement le header et il y a donc une erreur car ce ne sont pas les bons identifiants.

La fonction `after()` permet d'effectuer le code après que tout soit bien exécuter pour pas avoir problèmes d'opérations asynchrones.

Etape 2 :

Certificat signé.

```
linuxetu@linuxetu:~$ openssl x509 -req -days 365 -in server.req -signkey server.key -out server.crt  
Certificate request self-signature ok
```

Dans Postman, après avoir test le certificat, voici ce que j'ai.

Network

Local Address	192.168.214.1
Remote Address	192.168.214.128

TLS Protocol	TLSv1.3
Cipher Name	TLS_AES_128_GCM_SHA256

Certificate CN	
Issuer CN	
Valid Until	Feb 14 16:29:15 2025 GMT

 Self signed certificate

Tout d'abord c'est un certificat qui n'a pas de signature officielle.

```
▼ peerCertificate: {...}  
  ▼ subject: {...}  
    country: "FR"  
    stateOrProvince: "Some-State"  
    locality: "Paris"  
    organization: "Internet Widgits Pty Ltd"  
  ► issuer: {...}  
  validFrom: "Feb 15 16:29:15 2024 GMT"  
  validTo: "Feb 14 16:29:15 2025 GMT"  
  fingerprint: "29:DD:F4:4C:33:D4:5E:E4:C0:48:0C:88:F8:C7:E4:3D:1E:F1:AE:45"  
  serialNumber: "6b490fa51886eda5226366b8e819808caddb52ef"
```

J'ai bien les valeurs que j'ai remplies dans mon certificat : « FR » et « Paris ».