Faculté des Sciences & Techniques — Université de Limoges

Master 1ère année

Protocoles & Prog. Réseau

TD n°5

Protocole HTTP, cookie et format JSON

Récupération et traitement de données au format JSON

1 – Sous Linux, vous testerez la récupération de la météo avec les commandes suivantes :

xterm
```
$ curl 'http://wttr.in/Limoges'
$ http://wttr.in/Limoges?format=j1
```

Soit le programme suivant :

```
#!/usr/bin/python3

import requests

r = requests.get('http://wttr.in/Limoges?format=j1')
contenu = r.json()
```

À quoi correspond :

- ▷ `contenu['weather'][0]` ?
- ▷ `contenu['weather'][1]` ?
- ▷ `len(contenu['weather'][1]['hourly'])` ?

Indication : pour afficher les données JSON de manière « *agréable* », vous pouvez utiliser :

```
from pprint import pprint
pprint(r.json())
```

Récupérez les **informations pertinentes** de la météo courante et affichez les à l'utilisateur.

LemonLDAP::NG

Ce protocole permet de réaliser du « *Single Sign On* », c-à-d une authentification unique pour de multiples services au travers du Web, à travers l'interface d'un navigateur, en utilisant les techniques et protocoles suivant : `SSL`, `HTML`, `HTTP` et `CGI`.
Une documentation ainsi qu'une liste d'organismes utilisant ce protocole est accessible à `http://lemonldap-ng.org/references`.

Déroulement du protocole d'authentification :

L'**authentification initiale** se fait par l'intermédiaire du serveur Web hébergeant le programme CAS :

- la connexion directe à ce serveur Web, d'adresse `cas.unilim.fr`, sur le port associé au protocole « https », c-à-d en établissant une connexion SSL, « *Secure Socket Layer* », sur le port `443`, puis en échangeant suivant le protocole « http » ;
- envoi des données contenant les « login » et « mot de passe » de l'utilisateur dans une requête HTTP de type « POST » (un « *token* » de sécurité est ajouté pour éviter les CSRF, « *Cross Site Request Forgery* ») ;
- réception d'une réponse de la part du serveur, avec deux cas possibles :
  - **l'authentification a fonctionné**, c-a-d le login et mdp correspondent à un utilisateur autorisé, alors l'entête HTTP de la réponse contient la définition du cookie :

    ```
    Set-Cookie: lemonldap=ac32089a8ba7d62785155200ff03da0d; domain=.unilim.fr;
    path=/; secure; HttpOnly
    ```

    explications des champs paramètrant les cookies :
    - *« secure » : ces cookies ne peuvent être échangés qu'en mode connexion sécurisée, « https » ;*
    - *« HttpOnly » : ces cookies ne peuvent être échangés qu'avec des requêtes de type « http », en interdisant leur accès par des scripts Javascripts ;*
    - *« domain=.unilim.fr » : ces cookies ne concernent que le domaine « unilim.fr »*
  - **l'authentification n'a pas fonctionné**, c-a-d login et/ou mot de passe erronés, l'entête HTTP ne contient pas de définition de cookie.

L'utilisation de connexion chiffrée à l'aide de SSL/TLS permet de rendre confidentielle les données échangées entre la navigateur et le serveur Web (empêcher la récupération de l'identifiant et du mot de passe associé permettant l'authentification de l'utilisateur).

Les étapes de la **procédure d'authentification** sont les suivantes :

1. établissement d'une **première** connexion sécurisée SSL vers le serveur Web réalisant l'authentification, de TSAP :(`cas.unilim.fr`, `443`) ;
2. envoi d'une requête simple `GET / HTTP/1.0\nHost: cas.unilim.fr\n\n`
3. récupération d'une page HTML contenant la ligne :

   ```
   xterm
   <input id="token" type="hidden" name="token" value="160125632_12345" />
   ```

4. récupération de la valeur associée au champs `token` *ici la valeur* `160125632_12345`.
5. établissement d'une **seconde** connexion sécurisée SSL vers le serveur Web réalisant l'authentification, de TSAP :(`cas.unilim.fr`, `443`) ;
6. envoi de la requête POST avec les données suivantes ;
   - le champ : `user` qui contient votre identifiant d'accès unilim.fr ;
   - le champ : `password` qui contient votre mot de passe unilim.fr ;
   - le champ : `token` avec la valeur récupérée lors de la **première** requête au serveur.

   L'envoi des données du formulaire suivant la méthode `POST`, sur la ressource `/`, se fait en transmettant les données suivantes :

   ```
   POST / HTTP/1.0\r\n
   Host: cas.unilim.fr\r\n
   Content-Length: 12345\r\n
   Content-Type: application/x-www-form-urlencoded\r\n
   \r\n
   ```

   immédiatement suivie des données du formulaires formatées comme pour la méthode `GET` (d'où l'indication MIME *x-www-form-urlencoded*) :

   ```
   user=toto&password=XXXX&token=YYYYYYYYYY
   ```

   Il faudra bien sûr remplacer la valeur `12345` *du* `Content-Length` *par la taille exacte des données transmises.*
7. récupération d'une réponse avec l'entête HTTP contenant ou non la définition de cookie :

```
xterm
HTTP/1.1 302 Moved Temporarily
Server: nginx/1.14.2
Date: Mon, 07 Dec 2020 09:44:38 GMT
Content-Type: text/html
Content-Length: 161
Connection: keep-alive
Location: https://cas.unilim.fr/
Set-Cookie: lemonldap=aed2ea062cf11e...aff; domain=.unilim.fr; path=/; secure; HttpOnly
Set-Cookie: lemonldappdata=; path=/; HttpOnly=1; SameSite=Lax; secure
```

Pour savoir si l'authentification s'est bien passée, il suffit de trouver la chaîne « `Set-Cookie: lemonldap` » dans l'entête de la réponse.

Connexion SSL/TLS

Une connexion « SSL/TLS », correspond :

- à l'établissement d'une connexion TCP sur le port par défaut `443` ;
- à la négociation :
  - d'algorithmes de chiffrement disponibles chez le client et le serveur ;
  - des paramètres de ces algorithmes ;
- à l'authentification du serveur par l'intermédiaire d'un certificat électronique ;
- éventuellement, à l'authentification du client par un certificat (cette authentification est rarement faite).

Pour réaliser les deux connexions, nous utiliserons des bibliothèques de Python 3 :

```
import urllib.request
request = urllib.request.Request('https://cas.unilim.fr')
rep = urllib.request.urlopen(request)
contenu_page = rep.read()     # contient le token
```

Et pour récupérer le cookie :

```
#!/usr/bin/python3

import urllib.request
import urllib.parse

cookieProcessor = urllib.request.HTTPCookieProcessor()
opener = urllib.request.build_opener(cookieProcessor)
data = urllib.parse.urlencode({'user':'toto','password':'XXXXXX','token':'YYYYYYY'})

request = urllib.request.Request('https://cas.unilim.fr',bytes(data,encoding='as
cii'))
reponse = opener.open(request)
cookies = [c for c in cookieProcessor.cookiejar if c.name=='lemonldap']
print(cookies)
```

Écrivez un programme Python vérifiant les login/mdp d'un utilisateur auprès du serveur Lemon:LDAP de l'Université.