

A1_T3. Inspeccionar scripts amb exemples bàsics

Primer contacte amb scripts CGI.

Creació de scripts CGI:

Guia d'estil per Python. Utilitzarem Python per estudiar el protocol CGI, i és necessari que segueixin les normes habituals en escriure codi. [PEP8](#) per aplicar l'estil recomanat en escriure codi de Python.

Sempre has de treballar sota el teu compte d'usuari, i situar els scripts i pàgines a les carpetes apropiades sota el directori /var/www.

A banda de estudiar la documentació, pensa que l'script dump.py et mostra on es troben les dades els scripts CGI.

Enllaços recomanats:

Els presentats en l'activitat dedicada a l'organització de la documentació.

Pràctiques:

Crea un script per cada un dels següents exemples, amb identificadors i títols apropiats, i documenta amb comentaris l'script de forma apropiada.

Avís: pots provar els scripts en un terminal usant aquesta tècnica del Shell: `$ REQUEST_METHOD=GET QUERY_STRING='...' ./script.py`.

Avís: tots els scripts comencen amb el mateix codi, que no es mostra en els exemples. Aquest és (sense comentaris):

```
#!/usr/bin/python

import os, sys
import cgi; cgi.enable()

write = sys.stdout.write

write('Content-Type: text/plain; charset=UTF-8\r\n')
write('\r\n')
```

Exemples:

1. Mostra variables d'entorn

```
for (k, v) in os.environ.items():  
    write("%s = %s\n" % (k, v))
```

Iteritems ahora pasa a ser **items**

2. Mostra query string

```
import urllib  
  
camps = os.environ['QUERY_STRING'].split("&")  
  
for i in range(len(camps)):  
    camps[i] = camps[i].split("=", 1)  
  
for i in range(len(camps)):  
    camps[i] = (camps[i][0], urllib.unquote_plus(camps[i]  
[1]))  
  
for (k, v) in camps:  
    write("%s = %s\n" % (k, v))
```

urllib.unquote_plus ahora es **urllib.parse.unquote_plus**

Para ejecutarlo:

**REQUEST_METHOD=GET QUERY_STRING='Nom=Diego&Cognoms=Sanchez
%20Piedra' python3 2.py**

3. Consulta entorn

```
user_agent = os.environ.get("HTTP_USER_AGENT", "A bad  
client!")  
write("User-Agent:" + user_agent + '\n')  
  
if "HTTP_REFERER" in os.environ:  
    http_referer = os.environ["HTTP_REFERER" ]  
    write("Referer: " + http_referer + '\n')  
  
if os.environ.has_key("HTTP_ACCEPT_LANGUAGE"):  
    write("Accept-Language: " +  
os.environ["HTTP_ACCEPT_LANGUAGE"] + '\n')
```

```
has_key ya no existe, la alternativa es get:  
if os.environ.get("HTTP_ACCEPT_LANGUAGE"):
```

Lo que hace este script es mostrar en el navegador el user_agent que se está usando, en caso de que haya la variable de entorno HTTP_REFERER la muestra y en caso de que haya la variable ACCEPT_LANGUAGE también la muestra.

4. Consulta la documentació de la funció `urllib.unquote_plus`.
Ahora la libreria `urllib` usa el módulo `parse` para las funciones de URL quoting, entonces seria `urllib.parse.unquote_plus`, lo que hace `unquote_plus` es lo que hace `unquote` pero además reemplaza los signos mas en espacios, lo que hace la función `unquote` es reemplazar `%XX` con caracter equivlante segun su URL encoding.
<https://docs.python.org/3/library/urllib.parse.html#module-urllib.parse>
https://www.w3schools.com/tags/ref_urlencode.ASP
5. Estudia amb tota atenció l'script `dump.py`
Lo que hace es coger todas las variables de entorno del servidor y mostarlas en un formato especificado en el script junto a un banner que está compuesto por `'-'`.
6. Implementa partint de zero una funció de nom `decode` equivalent a `urllib.unquote_plus`.