

# Laboratorio 1: Aplicación Cliente

Cátedra de Redes y Sistemas Distribuidos

## Objetivos

---

- Aplicar la comunicación cliente/servidor por medio de la programación de sockets, desde la perspectiva del cliente.
- Familiarizarse con un protocolo de aplicación y su uso por medio de su RFC.
- Leer y comprender código Python no-trivial.

## Hget

---

hget es un cliente HTTP, desarrollado en Python 3, que obtiene una página desde un servidor y la almacena localmente en un archivo. El protocolo [HTTP](#) está definido en el [RFC 2616](#).

## Tarea

---

Descargar el kickstart del laboratorio desde el aula virtual. Descomprimir con: `tar -xvzf kickstart_lab1.tar.gz`.

Deberán completar el cliente para que funcione<sup>1</sup>. El código de hget está esencialmente ya escrito y se lo damos con solamente una de las funciones incompletas: `connect_to_server()`.

A continuación daremos un ejemplo de la ejecución:

```
$ ls
hget.py
$ python3 hget.py http://www.trashloop.com/
Contactando servidor 'www.trashloop.com'...
Contactando al servidor en 138.68.199.111...
Enviando pedido...
Esperando respuesta...
$ ls
download.html  hget.py
```

---

<sup>1</sup> Prestar atención que muchas páginas trabajan con HTTPS. La S al final de HTTPS indica Secure, ya que HTTPS utiliza encriptación e intercambio de llaves. Nuestra implementación trabaja en texto plano, por lo que usaremos HTTP puro, y deberemos acceder a páginas HTTP y no HTTPS.

Luego de solicitar una página por su URL, obtenemos un archivo llamado download.html. Puede elegirse un nombre de archivo distinto con la opción -o:

```
$ python3 hget.py http://www.fallingfalling.com/ -o caida_eterna.html
Contactando servidor 'www.fallingfalling.com'...
Contactando al servidor en www.fallingfalling.com...
Enviando pedido...
Esperando respuesta...
$ ls
download.html  caida_eterna.html  hget.py
```

Nota: En el código pueden ver que las funciones tienen un *docstring* que describe su comportamiento. Además de funcionar como documentación, pueden ser utilizados como tests automáticos. Para ejecutar estos tests pueden hacer:

```
$ python3 -m doctest hget.py
```

Además el archivo hget-test.py tiene tests adicionales que deben ser ejecutados con:

```
$ python3 hget-test.py -v
```

## Tarea Estrella

---

Opcionalmente, y con la posibilidad de que se otorguen puntos extras en la evaluación parcial, se pide investigar qué mecanismos permiten funcionar a nombres de dominio como:

- `http://中文.tw/`
- `https://🐼.la2`

Ayuda: investigue sobre el término “encoding”.

## Requisitos del código a entregar

---

- No se solicita un informe para este laboratorio, salvo para el punto estrella, cuya respuesta deberá estar dada en texto plano o Markdown. Pero el código que escriban debe contener comentarios con detalles de lo que hicieron.
- Las entregas serán a través del repositorio Git provisto por la Facultad para la Cátedra, con **fecha límite indicada en el cronograma del aula virtual**.
- El código debe satisfacer PEP8.
- La biblioteca de sockets de Python permite saltarse algunos pasos y omitir algunos argumentos si estamos usando los valores por defecto. No usen eso (para este Lab), así se ven los valores explícitos.

---

<sup>2</sup> No es necesario que se pueda descargar con nuestro cliente por ser HTTPS.