

# Tarea 1.

Redes de Computadores.

June 3, 2017

Profesor:

- Gabriel Torres.

Ayudantes:

- Fabián Saso      fabian.saso@alumnos.inf.utfsm.cl
- Catherin Vargas      catherin.vargas.13@sansano.usm.cl

## 1 Parte 1: Sockets y Server TCP.

La primera parte de la tarea consiste en programar un servidor que sea capaz de:

- Enlazarse a un socket y escuchar por conexiones entrantes en los puertos a especifica. Dicho servidor deberá retornar un HTML dependiendo del URI que se le otorgue al browser.
- Cuando este servidor inicie se debe crear un Thread Pool.
- Por cada conexión entrante asignarle un thread para su ejecución.  
EJ: localhost:8080/about.html deberá retornar una página about.html
- Para esto deberá programar un HTTP Handler para cada URI.

Se deberán programar los siguiente HTTP Handlers:

1. **GET http://localhost:8080:** Es la página de bienvenida y deberá retornar una página home.html, con un response header 200 (OK).
2. **GET http://localhost:8080/home\_old:** Corresponde a una página que ya no existe, deberá enviar un response header 301 (Moved Permanently) y redirigir a la página de bienvenida.
3. **GET http://localhost:8080/secret:** Corresponde a una página secreta, ingresos a esta página deberán retornar un response code 403 (Forbidden) si es que no esta autenticado y una página HTML que le advierta al usuario que no puede ingresar. No es necesario el uso de cookies ni sesiones para autenticar al usuario.

## 2 Parte 2: Métodos HTTP.

Se deberá generar un formulario para que los usuarios puedan autenticarse y así acceder a la página que se esconde tras **http://localhost:8080/secret**. Se usará un browser a su elección para ver las páginas HTML que retornará su servidor.

En esta parte deberá programar los siguientes HTTP Handler:

1. **GET http://localhost:8080/login:** Retorna un formulario con usuario y password como campos, luego al hacer click en "Log in", debe hacer un request hacia:  
**POST http://localhost:8080/secret** y si el usuario y password coinciden con los aprobados por el servidor, retorna una página 200(OK), en vez de 403 (Forbidden).

2. **POST http://localhost:8080/secret:** Tiene como propósito la validación para el usuario y password proveniente del formulario **login**, los parámetros de autenticación son los siguientes:

- Usr: root
- pwd: rdc2017

Si las credenciales coinciden se dejará un archivo que le permitirá a la aplicación saber que ya hubo una autenticación aceptada. Puede usar cualquier método para esta operación. Al ingresar a **GET http://localhost:8080/secret**, luego de un login exitoso, deberá mostrar una página HTML con el contenido “Autenticación Completada”

Es decir, si bien no pueden acceder a **GET http://localhost:8080/secret** ya que siempre retornará 403(Forbidden) si es que no están autenticados, la forma de autenticarse es mandando el formulario a **POST http://localhost:8080/secret**, lo que generará el archivo dejando constancia de una previa autenticación exitosa.

### 3 Bonus.

Agregue una funcionalidad extra a su servidor. Registre en un archivo de texto log.txt la IP que hizo el Request por cada conexión. El formato del archivo debe ser <IP> <URL> <Hora de la consulta>.

### 4 Restricciones.

- No puede usar ninguna implementación en Java que genere servidores.
- No puede usar ninguna implementación en Java para la generación del ThreadPool. (EJ: Executor, ExecutorService).

### 5 Entrega.

- La tarea debe desarrollarse en lenguaje de Programación Java 1.7 o 1.8.
- Se debe trabajar en grupos de a los más 3 personas.
- Las dudas o aclaraciones deben ser canalizadas a través de la plataforma Moodle.
- La entrega debe realizarse a través de Moodle, en un archivo comprimido RAR O ZIP y debe llevar el nombre T1-Rol1-Rol2-Rol3.
- Dentro de la entrega debe ir un archivo README.txt que debe contener nombre y rol de los integrantes del grupo y cualquier supuesto que utilizó durante el desarrollo de la tarea, junto con un instructivo de cómo compilar el código adjunto (bastan los comandos y librerías usadas, no quiero un paper sobre como correr cada tarea).
- El plazo máximo de entrega será el 18 de Junio a las 23:55 hrs.
- Por cada día de atraso se descontarán 20 puntos, considerando el fin del día las 23:59 hrs. Todas las entregas desde las 00:00 serán consideradas atrasadas por un día.
- Copias serán evaluadas con nota 0.