**ACTIVIDAD 2**

**Enunciado**

Realizar un programa en Java que cree la siguiente estructura de directorios dentro de un servidor FTP:

aplicacionwebapellido

html

imagenes

css

La parte apellido del nombre del directorio aplicacionwebapellido es preciso personalizarla con el primer apellido de la persona que realiza la actividad. Por ejemplo: aplicacionwebochoadealda

El servidor con el que conectarse vía FTP será *proFTPd* y se instalará/configurará en una máquina virtual de Oracle VM VirtualBox que tenga instalada la distribución de GNU/Linux habilitada en el aula.

Justo antes de realizar la conexión con el servidor se visualizará el mensaje “Nos conectamos a xxx.xxx.xxx.xxx”, siendo xxx.xxx.xxx.xxx la dirección IP del servidor FTP.

El acceso al servidor será con un usuario llamado *dinux* y con la contraseña *dinux,* que previamente se habrá dado de alta en la máquina virtual donde esté instalado el servidor proFTPd.

Si la conexión tiene éxito se visualizará el mensaje Login correcto. En caso contrario, el mensaje a visualizar será Login incorrecto…

Si la conexión ha sido exitosa se mostrará un mensaje indicativo del directorio en el que se está después de haber establecido la conexión: Directorio actual: /

Si la creación es satisfactoria se mostrará el mensaje: Directorios creados... Si la creación no es satisfactoria se mostrará el mensaje: NO SE HAN PODIDO CREAR LOS DIRECTORIOS

Si al hacer logout en el servidor se tiene éxito, se mostrará el mensaje: Logout del servidor FTP...En caso contrario, se mostrará el mensaje: Error al hacer logout

Después de que el programa cliente se desconecte del servidor FTP, se mostrará el mensaje Desconectado...

No olvidar que la clase Java FTPClient puede lanzar excepciones y que es preciso tratarlas.

El nombre de la clase principal será Actividad2.

Si no fuera posible crear los diferentes directorios en el servidor FTP, el resultado esperado sería el siguiente (a excepción de la dirección IP del servidor que variará según cada alumno):

Nos conectamos a 172.20.105.140

Login correcto

Directorio actual: /

NO SE HAN PODIDO CREAR LOS DIRECTORIOS

Logout del servidor FTP....

Desconectado...

Si fuera posible crear los diferentes directorios en el servidor FTP, el resultado esperado sería el siguiente (a excepción de la dirección IP del servidor que variará según cada alumno):

Nos conectamos a 172.20.105.140

Login correcto

Directorio actual: /

Directorios creados...

Logout del servidor FTP....

Desconectado...

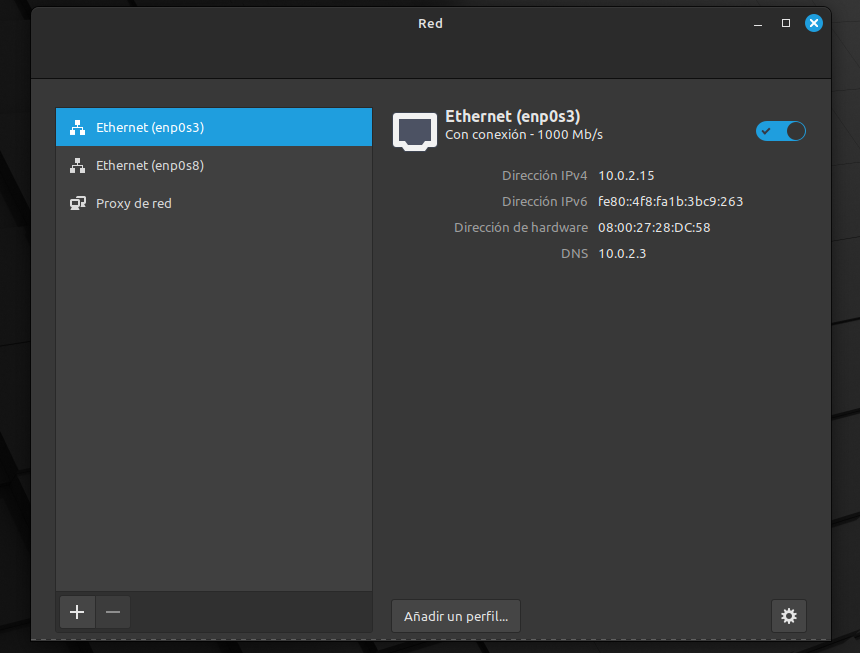
De cara a comprobar la operatividad del servidor proFTPd, puede resultar de utilidad instalar el cliente Filezilla en la máquina virtual y conectarse al mismo mediante la dirección IP del localhost (127.0.0.1) y con los usuarios que se hayan podido habilitar en la misma.

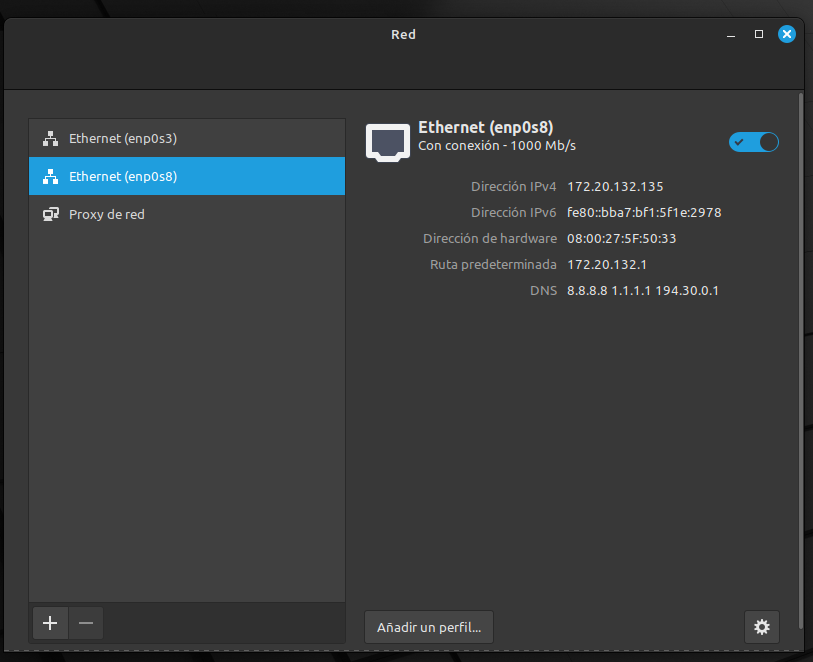
**Solución**

Visitar [www.redeszone.net/gnu-linux/proftpd/](http://www.redeszone.net/gnu-linux/proftpd/) para ver los detalles sobre cómo configurar el servidor proFTPd.

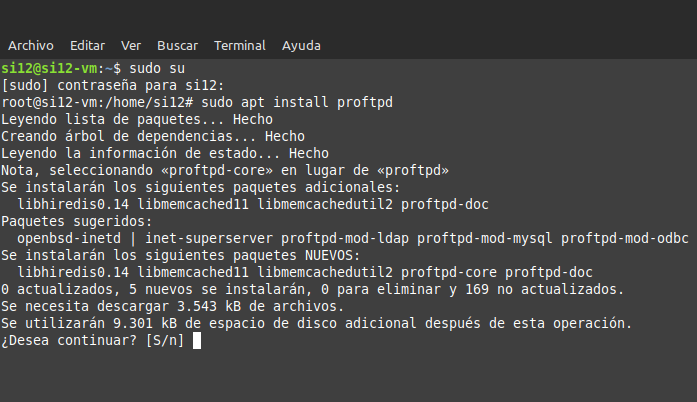
Incluir aquí evidencias de las siguientes cuestiones relacionadas con la máquina virtual en la que estará alojado el servidor proFTPd:

1. **Cómo se han configurado dos interfaces de red en la máquina virtual: una en modo NAT (para poder descargar el servidor proFTPd) y otra en modo puente (para poder acceder a la misma desde el cliente Java del anfitrión).**

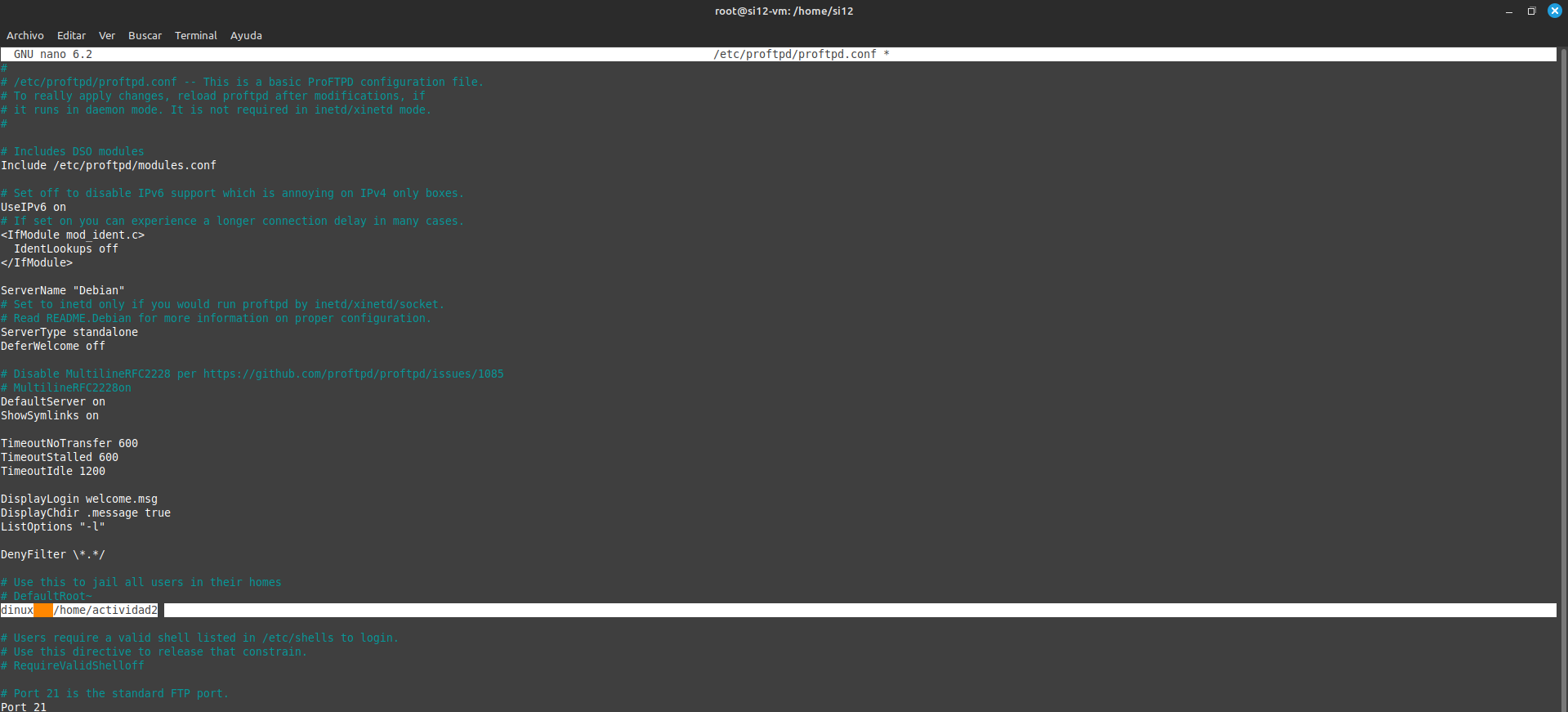




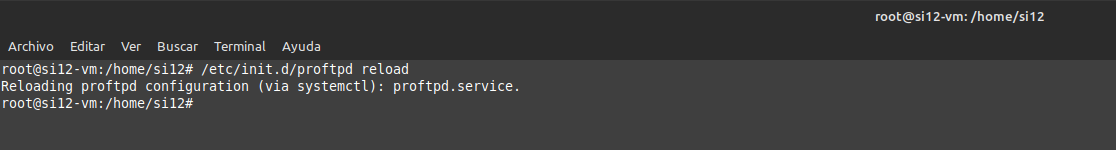
1. **Cómo se ha hecho la instalación del servidor en la máquina virtual**



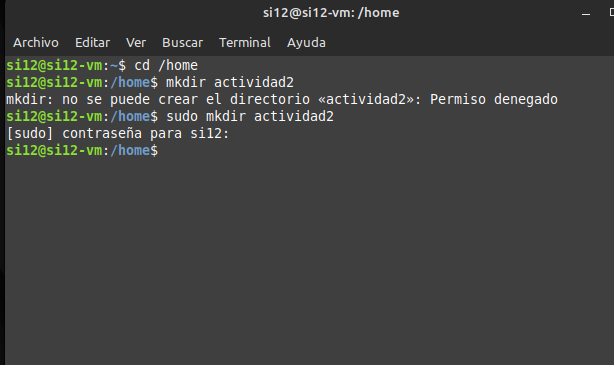
1. **Cómo se ha configurado el servidor para que cuando se entre al mismo con el usuario dinux, éste se conecte por defecto a la carpeta /home/actividad2. Esta configuración aparece en el archivo de configuración del servidor en un parámetro llamado DefaultRoot.**



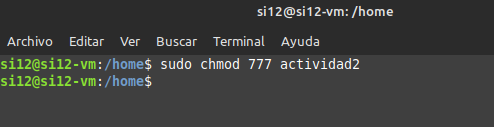
1. **Cómo se ha reiniciado el servidor para que el cambio realizado en el punto anterior tenga efecto:**



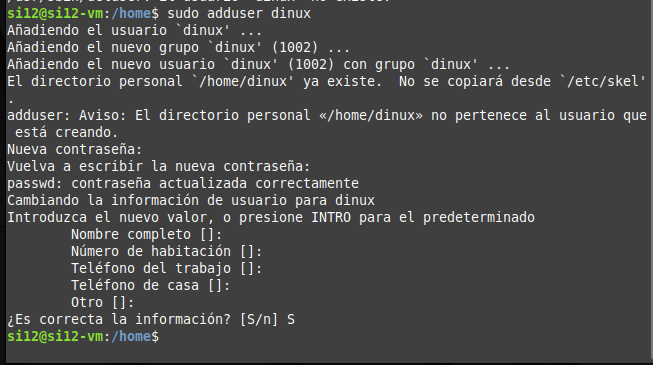
1. **Cómo se ha creado la carpeta actividad2 dentro del directorio /home:**

****

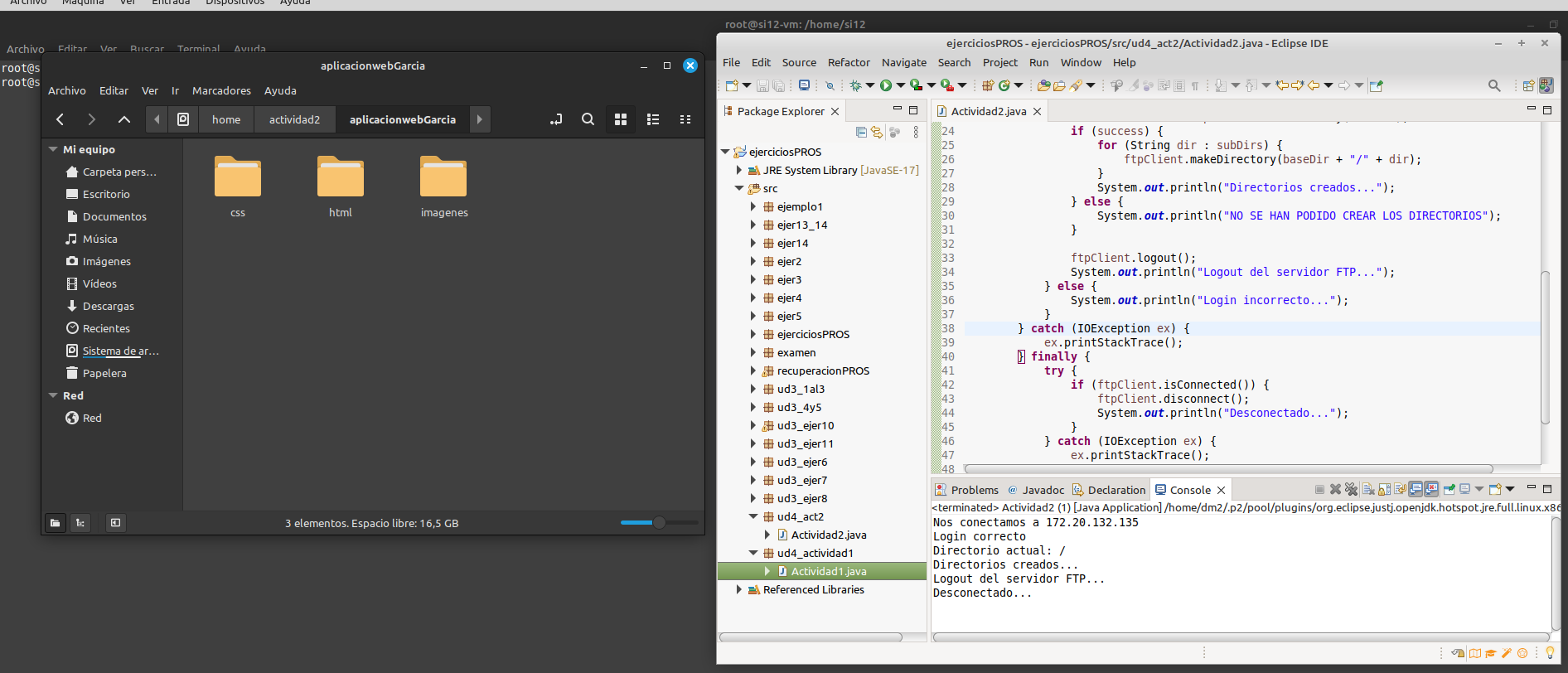
1. **Cómo se han dado permisos de escritura a cualquier usuario a la carpeta actividad2 dentro del directorio /home:**



1. **Cómo se ha creado el usuario dinux con la contraseña dinux con el que entrar en el servidor desde el cliente Java (o desde otro cliente, como Filezilla):**



1. **Cómo el programa cliente en Java ha llegado a crear la estructura de directorios requerida en el enunciado:**

****

**codigo Actividad2.java:**

package ud4\_act2;

import org.apache.commons.net.ftp.FTPClient;

import java.io.IOException;

public class Actividad2 {

public static void main(String[] args) {

String server = "172.20.132.135"; // IP del servidor FTP

String user = "dinux";

String pass = "dinux";

String apellido = "Garcia";

String baseDir = "aplicacionweb" + apellido;

String[] subDirs = {"html", "imagenes", "css"};

FTPClient ftpClient = new FTPClient();

try {

System.out.println("Nos conectamos a " + server);

ftpClient.connect(server);

if (ftpClient.login(user, pass)) {

System.out.println("Login correcto");

System.out.println("Directorio actual: " + ftpClient.printWorkingDirectory());

boolean success = ftpClient.makeDirectory(baseDir);

if (success) {

for (String dir : subDirs) {

ftpClient.makeDirectory(baseDir + "/" + dir);

}

System.out.println("Directorios creados...");

} else {

System.out.println("NO SE HAN PODIDO CREAR LOS DIRECTORIOS");

}

ftpClient.logout();

System.out.println("Logout del servidor FTP...");

} else {

System.out.println("Login incorrecto...");

}

} catch (IOException ex) {

ex.printStackTrace();

} finally {

try {

if (ftpClient.isConnected()) {

ftpClient.disconnect();

System.out.println("Desconectado...");

}

} catch (IOException ex) {

ex.printStackTrace();

}

}

}

}

**Resolución de problemas**

No he tenido problemas a la hora de resolver el ejercicio.

**REQUISITOS**

1. Entregar la solución de la actividad en un único archivo comprimido cuyo nombre es: el nombre original de la actividad, seguido de un guión bajo y del primer apellido del alumno. Donde tendremos:
   1. Un archivo de extensión .odt o .docx que contendrá
      1. Una copia del código del programa
      2. Una captura de pantalla de todas las evidencias como del resultado de la ejecución del programa
   2. El archivo con el código fuente de extensión .java

**CRITERIOS DE CORRECCIÓN**

1. Cumplimiento de los requisitos (15%).
2. Hacer lo que se indica en el enunciado (30%).
3. Aportación de evidencias en la solución (25%)
4. Claridad del código Java (15%).
5. Documentación del código Java (15%).
6. Explicación detallada de cómo se han resuelto problemas que hayan podido surgir en la realización de la actividad (hasta un 15% adicional sobre la nota obtenida). Dicha explicación se incluirá al principio de los archivos de código fuente Java mediante líneas de comentarios. La primera línea de comentarios sólo incluirá lo siguiente: // Resolución de problemas.
7. Variables no usadas (-15%).
8. Librerías no usadas (-15%).