**Unidad didáctica nº 3- Actividad de Desarrollo-**

**"Servicio Web: Apache 1ª parte "**

**Servicios en Red e Internet – 2º ASIR**

**I.E.S. Miguel Herrero**

**Curso 2021-22**

**Objetivos:** En esta práctica vamos a instalar y configurar un servidor Web: apache en un SO libre, Linux Ubuntu Server (instalación básica y hosts virtuales):

**Duración:** 4 h

**Recursos:**

Documentacion oficial ubuntu: https://ubuntu.com/server/docs/web-servers-apache

Pledin 3.0-> https://fp.josedomingo.org/serviciosgs/

Servidor DHCP en Mikrotik:

https://alexariza.net/tutorial/montar-servidor-dhcp-en-un-equipo-mikrotik/

https://www.youtube.com/watch?v=IQHLmdprdnQ

**Modo y formato de entrega:** Entregar a través de la plataforma online ***google classroom*** en pdf llamado A2 UD3 Tu nombre y apellidos.pdf generado a partir de este y en el cual deberás añadir las respuestas a las cuestiones planteadas así como las capturas de pantalla, logs y demás evidencias necesarias para demostrar que la práctica se ha realizado cumpliendo los requisitos solicitados en el enunciado.

* **Escenario:**

En esta práctica deberás instalar y configurar el servidor web Apache en Ubuntu Server.



**1ª PARTE:**

**Ejercicio1:** Instalación y configuración de la red.

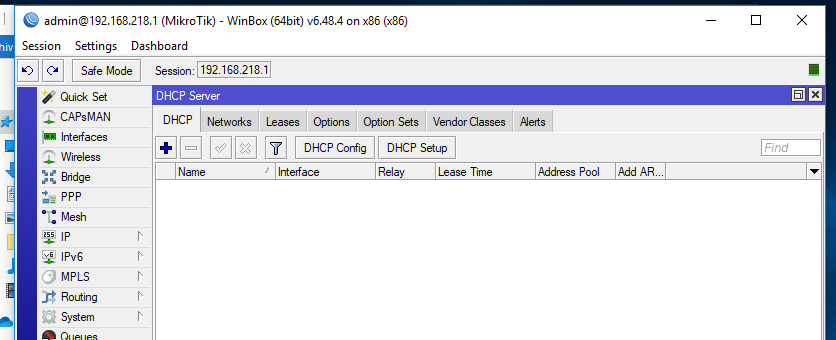
-Completa la siguiente tabla. Sigue el mismo direccionamiento que la Actividad 1 de la unidad 1. (**Nota**: En IPv6 incluye tanto la dirección global como la link-local) y configura de forma adecuada las máquinas virtuales indicadas en la topología

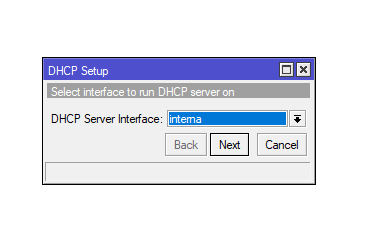
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Dispositivo** | **Interfaz** | **Dirección IPv4** | **Máscara** | | | **Gateway IPv4** |
| **Dirección IPv6 (Unicast Global)** | | | | **Gateway IPv6** |
| **Dirección IPv6 (Link Local)** | | | |
| **Router1 (Mikrotik)** | In | **192.168.218.1** | **/24** | | |  |
|  | | | |  |
|  | | | |
| Out | **10.0.16.218** | **/24** | | |  |
|  | | | |  |
|  | | | |
| **Cliente 1**  **Dual-stack** | NIC | **DINÁMICA (192.168.218.254)** | **DINÁMICA** | | |  |
|  | | | |  |
|  | | | |
| **Cliente 2**  **Dual-stack** | NIC | **DINÁMICA (192.168.218.101)** | **DINÁMICA** | | |  |
|  | | | |  |
|  | | | |
| **Cliente 4**  **Dual-stack** | NIC |  |  | | |  |
|  | | | |  |
|  | | | |
| **Servidor 3**  **Dual-stack** | NIC | 192 | |  | |  |
|  | | | |  |
|  | | | |
| **Servidor 5**  **Dual-stack** | NIC |  | | |  |  |
| NIC |  | | | |  |
|  | | | |

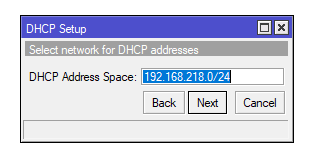
**Ejercicio 2:** Desde uno de los **clientes** (1 o 2) conéctate remotamente al **Servidor 1** (**Ubuntu Server)** e instala y configura adecuadamente el servicio DHCP (configuración automática de red)

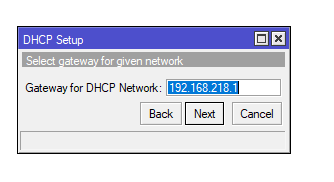
**Nota**: Como alternativa al servidor 1 puedes instalar y configurar el servicio DHCP en en el Mikrotik (ver **recursos**)

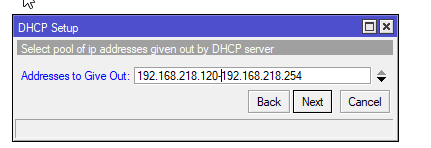
**Hago un servidor DHCP desde mikrotik**

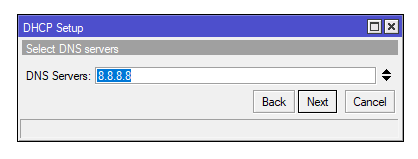
****

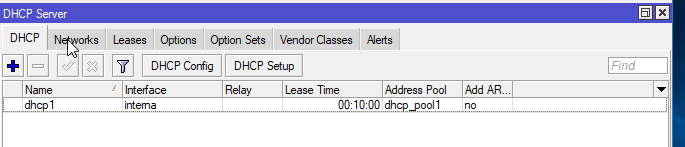
****

****

****

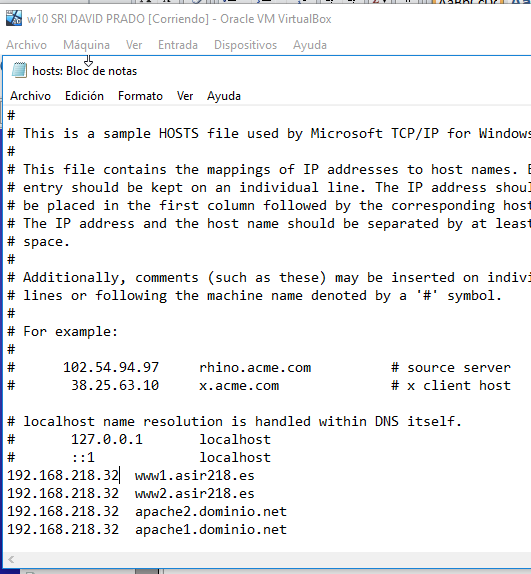
****

****

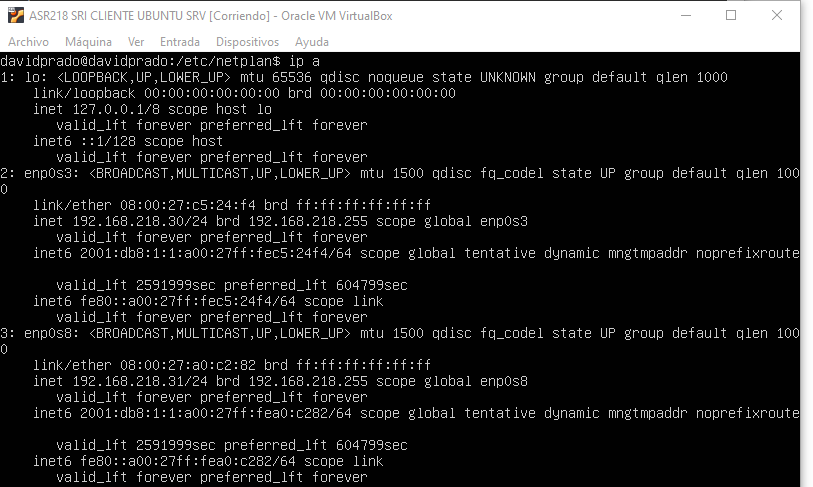
****

**Ejercicio 3**: Configura de manera adecuada algún sistema de resolución de direcciones que sea capaz de traducir los nombres ftp.asirXXX.es, ftp2. asirXXX.es, www1.asirXXX.es, www2. asiXXX.es, web1.asirXXX.es, web2.asirXXX.es, servidor1 y servidor2 a sus correspondientes direcciones IP (ver esquema de red).

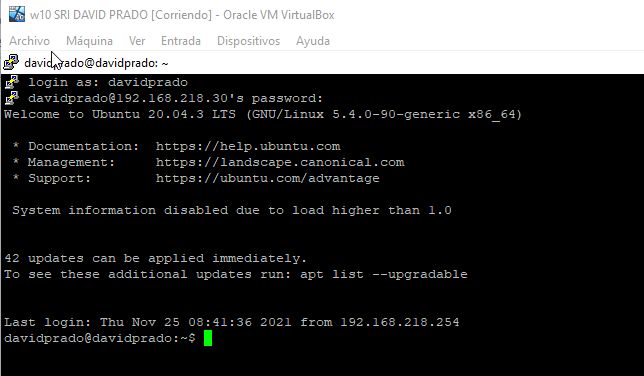
**Pista**: Archivo hosts



**He añadido las dos IP al Ubuntu apache**

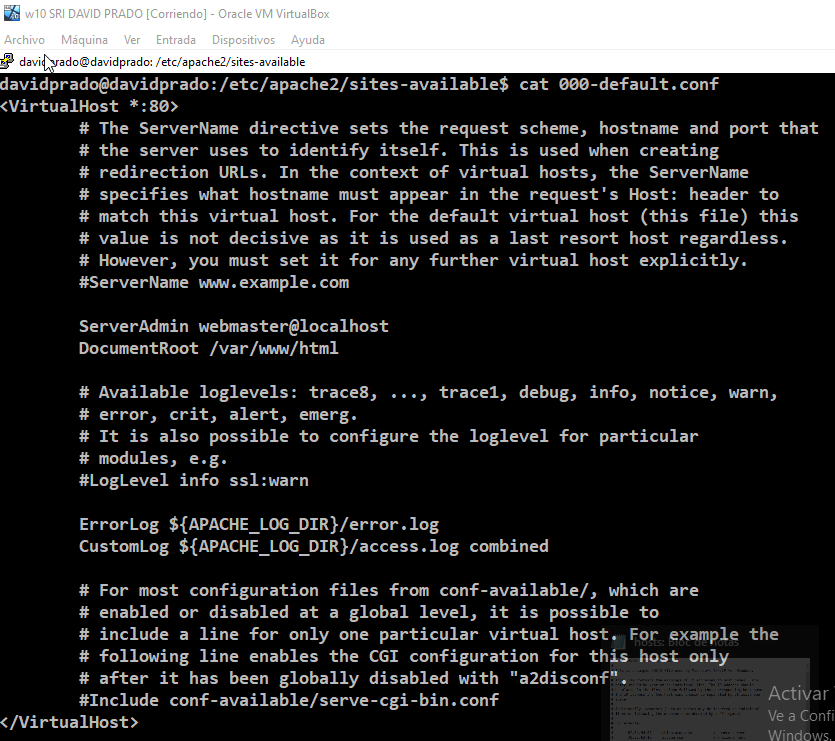
****

**Ejercicio 4:** Creación de distintos sitios web (o VirtualHost como los llama Apache) en el servidor web Apache. Desde uno de los clientes (1, 2 o 3) conéctate remotamente al **servidor1.** Vamos a añadir (**si no lo está)** y configurar, **si no lo están**, varios sitios webs/VirtualHosts (o **servidores virtuales**):

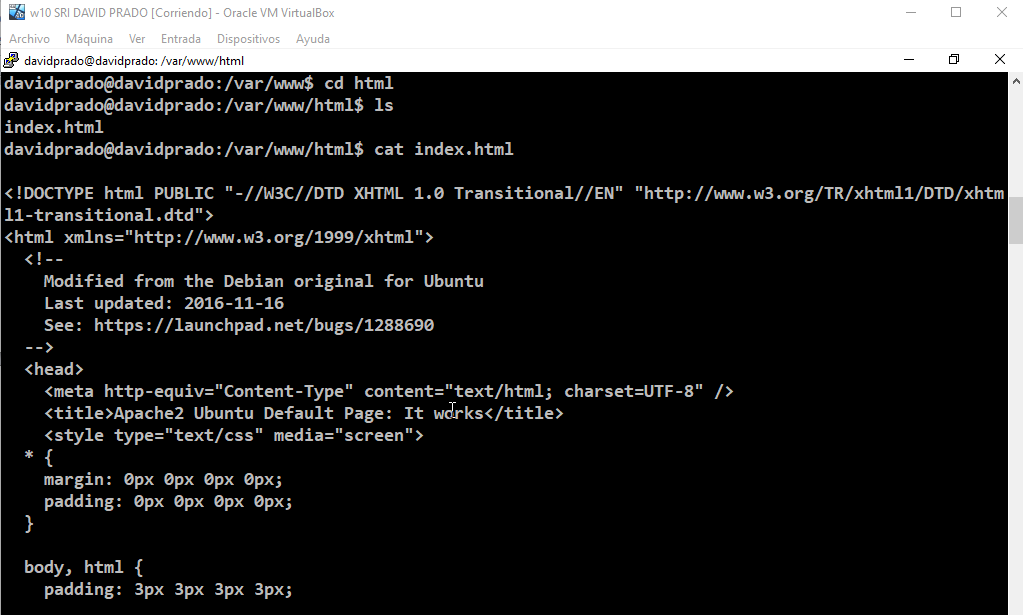


**Sitio 1:** Sitio web por defecto

* 1. El nombre archivo .conf: **000-default.conf**



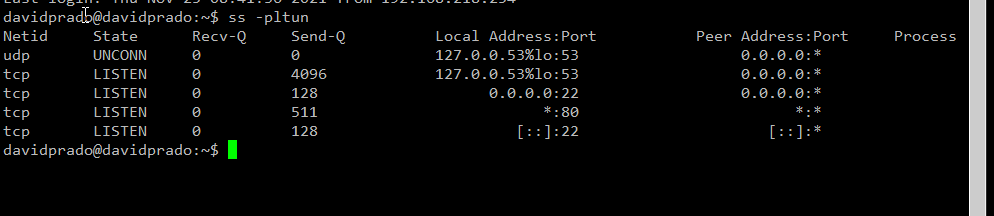
* 1. La ruta de acceso será **/var/www/html**



* 1. URL: **http://192.168.XXX.30/**
  2. Nombre del host: **No tiene**

Realiza las siguientes comprobaciones en el **servidor 1**:

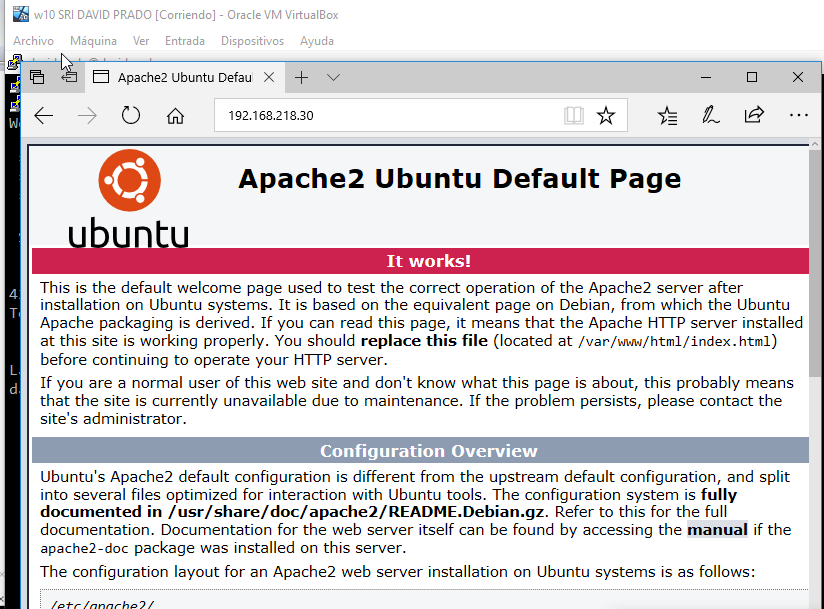
-Comprueba en el **servidor 1** que el puerto 80 está abierto y en escucha



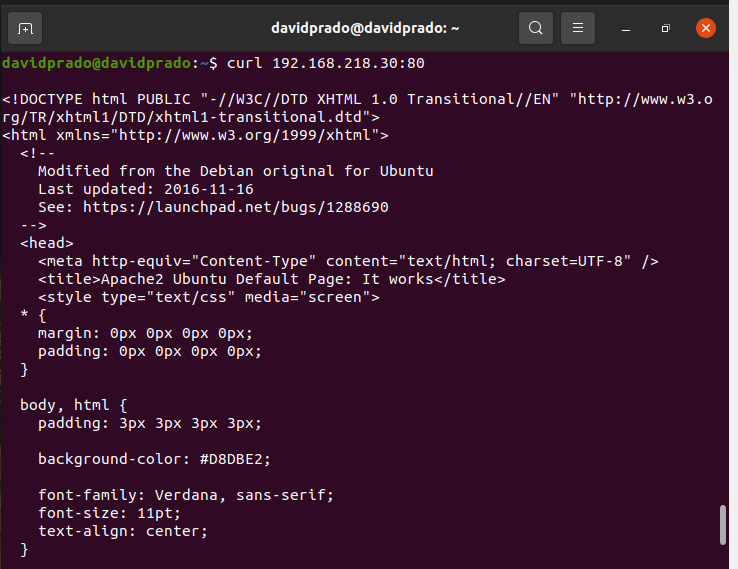
-Desde un cliente Linux escanea los puertos de **servidor 1** con el comando nc(netcat) y/o rustcat(rc) comprobando, entre otros, que el puerto 80 está abierto.



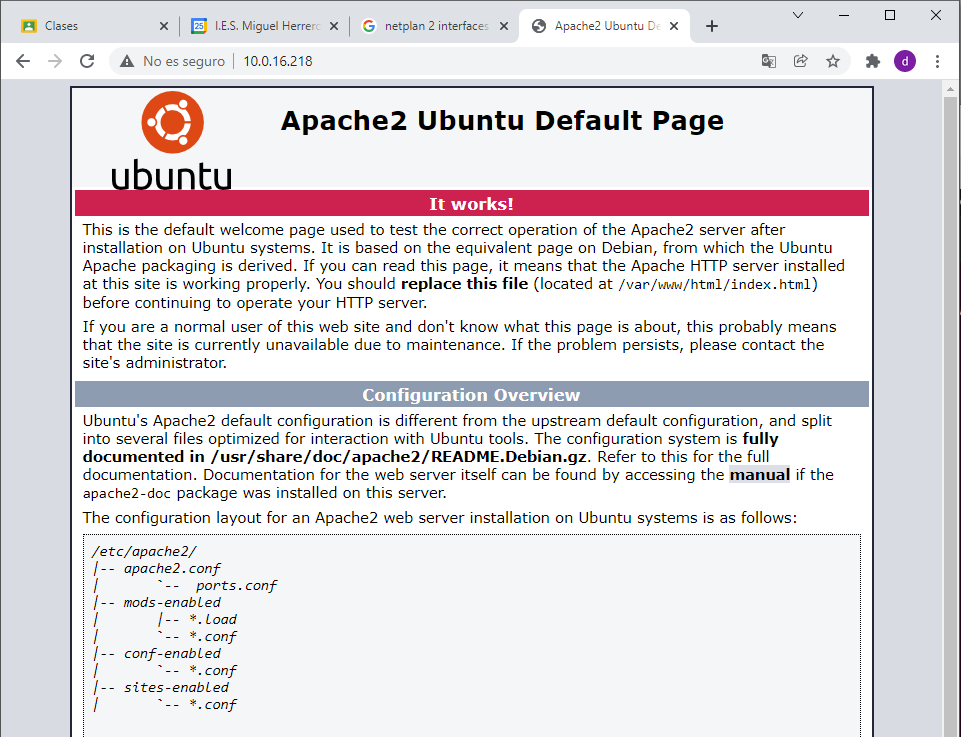
-Accede al sitio web (sitio 1) indicando su URL desde uno de los **clientes 1 o 2 (**desde la LAN Windows o Linux gráfico**)**,



**servidor 1** (desde la LAN Linux de terminal)



**cliente 4 (**desde la WAN Windows o Linux**)**



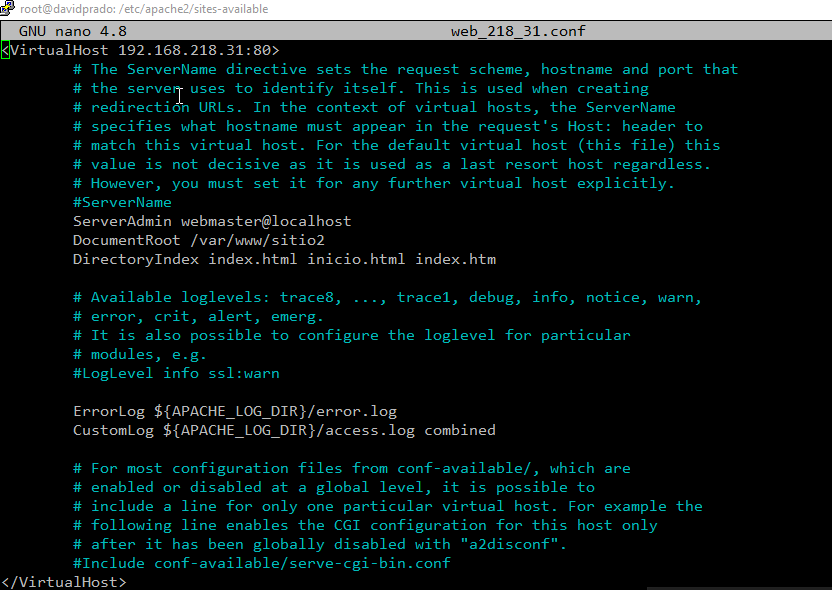
**Nota**: Al ser el **servidor 1** ubuntu Server (sistemas operativos de terminal) puedes utilizar las siguientes utilidades dese la línea de comandos para abrir las webs del servidor:

**curl** http://192.168.XXX.30/

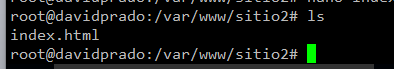
**wget** <http://192.168.XXX.30/>

**Sitio 2:** Creamos un segundo servicio web (sitio web) en el mismo servidor físico 2 pero accesible con la dirección IP: **192.168.XXX.31**

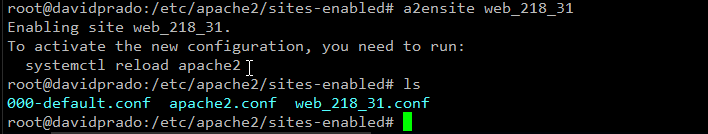
1. El nombre archivo .conf: **web\_XXX\_31.conf**

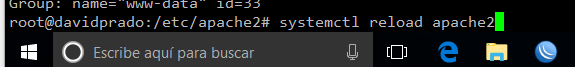


1. La ruta de acceso (directorio de contenido): **/var/www/sitio2**









1. URL: **http://192.168.XXX.31/**
2. Nombre del host: **No tiene**
3. Crea en su directorio de contenido (documentroot) un archivo llamado **index.html** con el siguiente código html:

<!DOCTYPE html>

<html>

<head>

<title> Tu\_nombre\_y\_apellidos </title>

</head>

<body>

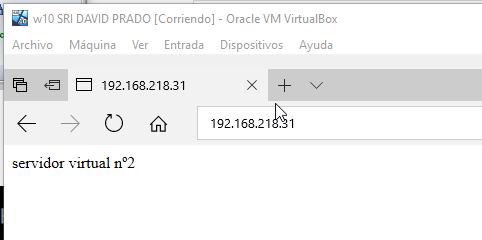
<h1> Servidor virtual nº 2</h1>

</body>

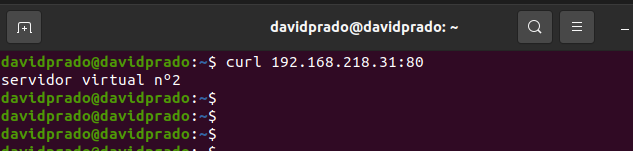
1. **E**stablece **index.html** como documento predeterminado.

Realiza las siguientes comprobaciones en el **servidor 1**:

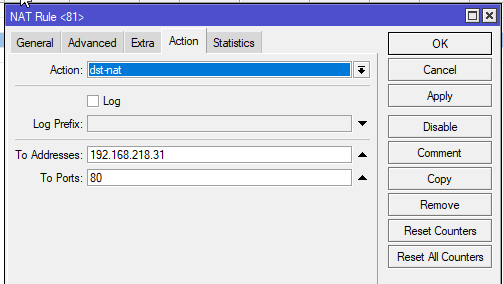
-Accede al sitio web (sitio 2) indicando su URL desde uno de los **clientes 1 o 2 (**desde la LAN Windows o Linux gráfico**)**,

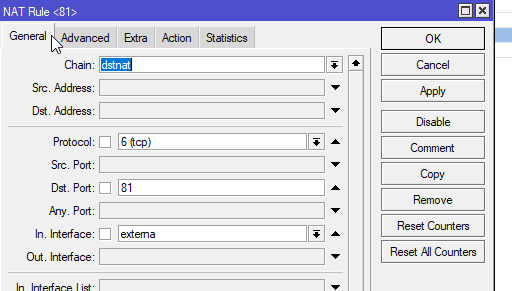


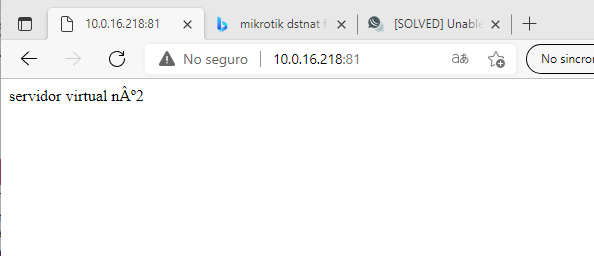
**servidor 1** (desde la LAN Linux de terminal)



**cliente 4 (**desde la WAN Windows o Linux**)**







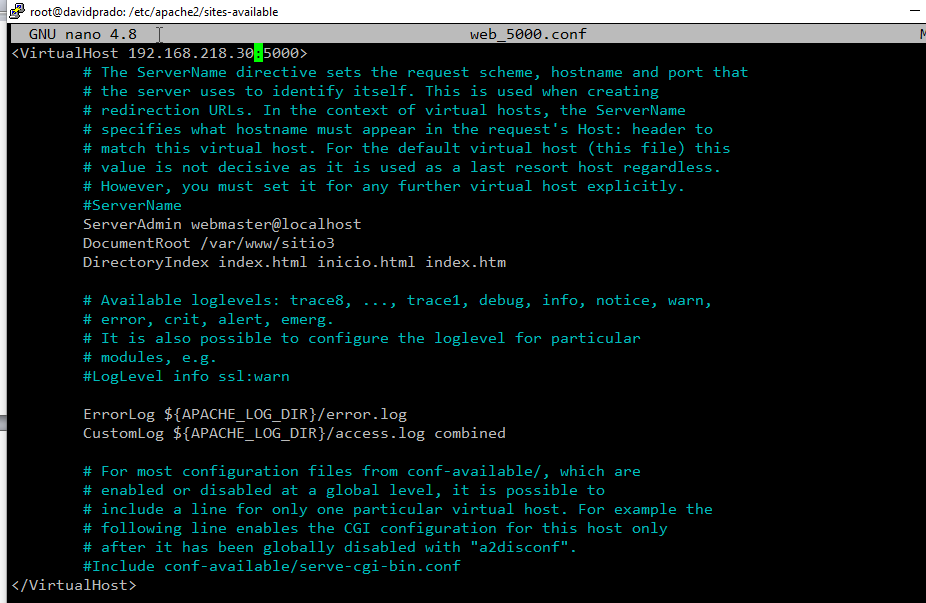
**Nota**: Al ser el **servidor 1** ubuntu Server (sistemas operativos de terminal) puedes utilizar las siguientes utilidades dese la línea de comandos para abrir las webs del servidor:

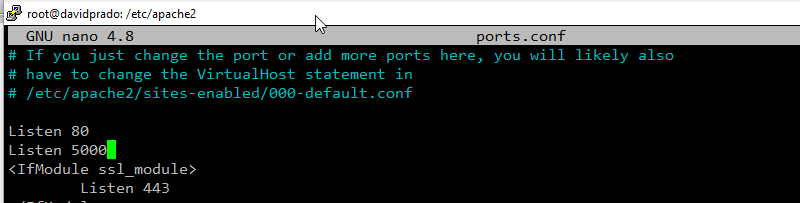
**curl** http://192.168.XXX.31/

**wget** http://192.168.XXX.31/

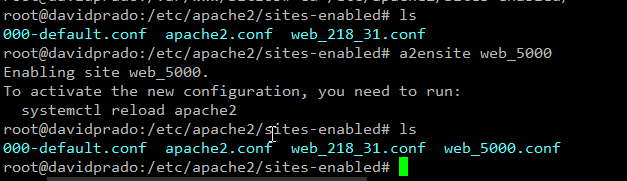
**Sitio 3:** Creamos un tercer servicio web (sitio web) en el mismo servidor físico 1 pero accesible a través del puerto **5000** y con la dirección **192.168.XXX.30**

1. El nombre archivo .conf: **web\_5000 .conf**





1. La ruta de acceso será **/var/www sitio3/**
2. URL**:** [**http://192.168.XXX.30:5000**](http://192.168.XXX.30:5000)



1. Nombre del host: **No tiene**.
2. Crea en su directorio de contenido (documentroot) un archivo llamado **index.html** con el siguiente código html:





<!DOCTYPE html>

<html>

<head>

<title> Tu\_nombre\_y\_apellidos </title>

</head>

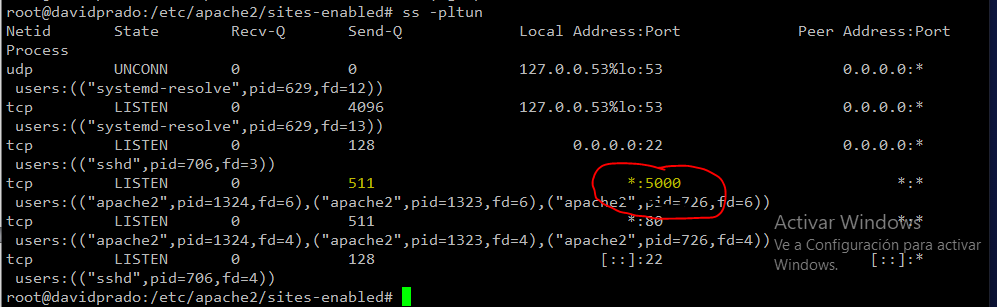
<body>

<h1> Servidor virtual nº 3</h1>

</body>

Realiza las siguientes comprobaciones en el **servidor 1**:

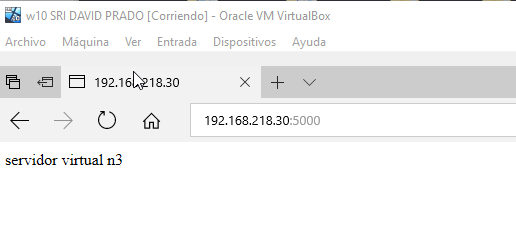
-Comprueba en el **servidor 1** que el puerto 5000 está abierto y en escucha



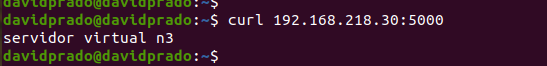
-Desde un cliente Linux escanea los puertos de **servidor 1** con el comando nc(netcat) y/o rustcat(rc) comprobando, entre otros, que el puerto 5000 está abierto.



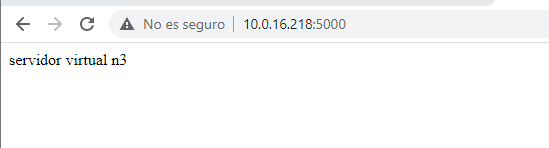
-Accede al sitio web (sitio 3) indicando su URL desde uno de los **clientes 1 o 2 (**desde la LAN Windows o Linux gráfico**)**,



**servidor 1** (desde la LAN Linux de terminal)



**cliente 4 (**desde la WAN Windows o Linux**)**



**Nota**: Al ser el **servidor 1** ubuntu Server (sistemas operativos de terminal) puedes utilizar las siguientes utilidades dese la línea de comandos para abrir las webs del servidor:

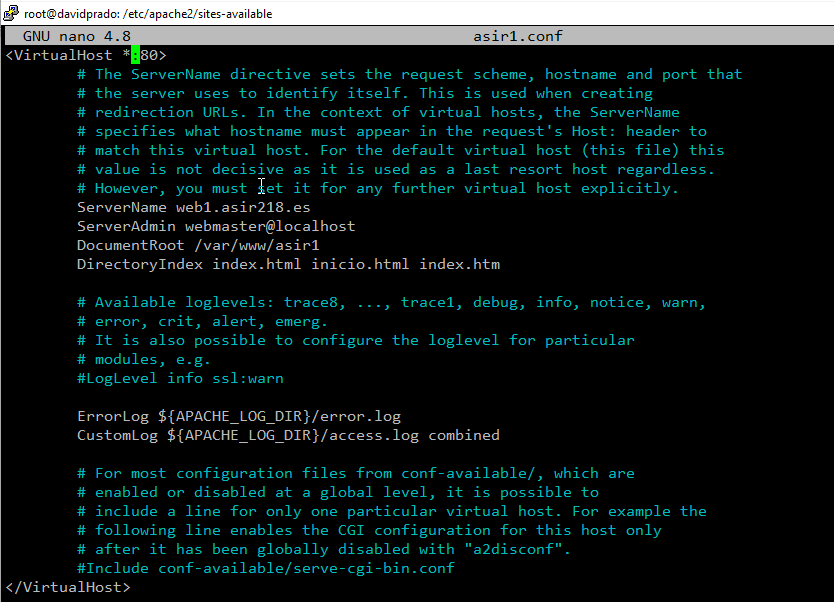
**curl** http://192.168.XXX.30:5000/

**wget** http://192.168.XXX.30:5000/

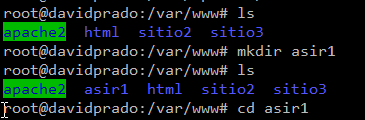
**Sitio 4: :** Creamos un cuarto servicio web (sitio web) en el mismo servidor físico pero accesible según la cadena de entrada **web1.asirXXX.es**:

1. El nombre archivo .conf: **asir1.conf**.

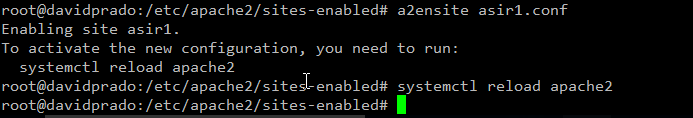




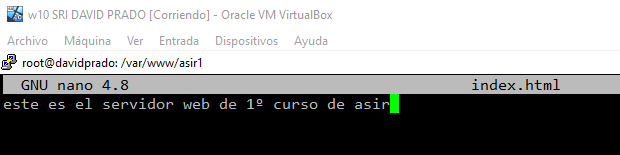
1. La ruta de acceso será **/var/www/asir1**



1. URL: [**http://web1.asirXXX.es**](http://web1.asirXXX.es)



1. Nombre del host: **web1.asirXXX.es**
2. Crea en el interior de **/var/www/asir1**el archivo **index.html** con el siguiente contenido:



<!DOCTYPE html>

<html>

<head>

<title> Tu\_nombre\_y\_apellidos </title>

</head>

<body>

<h1> ESTE ES EL SERVIDOR WEB DE 1º CURSO DE ASIR </h1>

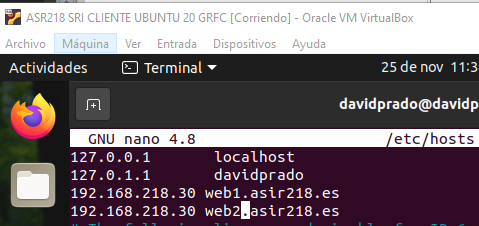
</body>

</html>

Realiza las siguientes comprobaciones en el **servidor 1**:

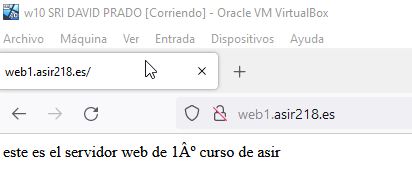
-Accede al sitio web (sitio 4) indicando su URL desde uno de los **clientes 1 o 2 (**desde la LAN Windows o Linux gráfico**)**

**servidor 1** (desde la LAN Linux de terminal)





**cliente 4 (**desde la WAN Windows o Linux**)**



**Nota**: Al ser el **servidor 1** ubuntu Server (sistemas operativos de terminal) puedes utilizar las siguientes utilidades dese la línea de comandos para abrir las webs del servidor:

**curl** http://web1.asirXXX.es/

**wget** http:// web1.asirXXX.es

/

**Sitio 5: :** Creamos un quinto servicio web (sitio web) en el mismo servidor físico pero accesible según la cadena de entrada **web2.asirXXX.es**:

1. Configuración básica:
2. El nombre archivo .conf: **asir2.conf**.
3. La ruta de acceso será **/var/www/asir2**
4. URL: **http://web2.asirXXX.es**
5. Nombre del host: **web2.asirXXX.es**
6. Crea en el interior de **/var/www/asir2/** el archivo **index.html** con el siguiente contenido:

<!DOCTYPE html>

<html>

<head>

<title> Tu\_nombre\_y\_apellidos </title>

</head>

<body>

<h1> ESTE ES EL SERVIDOR WEB DE 2º CURSO DE ASIR R</h1>

</body>

</html>

Realiza las siguientes comprobaciones en el **servidor 1**:

-Accede al sitio web (sitio 5) indicando su URL desde uno de los **clientes 1 o 2 (**desde la LAN Windows o Linux gráfico**)**,  **servidor 1** (desde la LAN Linux de terminal), **cliente 4 (**desde la WAN Windows o Linux**)**

**Nota**: Al ser el **servidor 1** ubuntu Server (sistemas operativos de terminal) puedes utilizar las siguientes utilidades dese la línea de comandos para abrir las webs del servidor:

**curl** http://web2.asirXXX.es/

**wget** http://web2.asirXXX.es /

1. **Para ver el funcionamiento del protocolo vamos a iniciar el analizador de protocolo *wireshark* en un cliente para capturar el tráfico intercambiado. Utiliza los filtros adecuados. Realiza el diagrama en el que se muestren los mensajes intercambiados entre cliente y servidor. Nota:** Si no recuerdas los filtros a utilizar visita la siguiente web:

https://seguridadyredes.wordpress.com/2008/03/24/analisis-de-red-con-wireshark-filtros-de-captura-y-visualizacian/

**Ejercicio 5. Instalar certificados digitales y asegurar las comunicaciones entre cliente y servidor web.**

* Instala en el servidor web un certificado digital autofirmado.
* Utiliza el certificado anterior para asegurar la comunicación entre clientes y tu servidor web.
  + https://web1.asirXXX.es
  + https://web2.asirXXX.es
* Captura el tráfico intercambiado y observa la diferencia respecto a la comunicación cuando te conectas con HTTPs.
* Realiza el diagrama en el que se muestren los mensajes intercambiados entre cliente y servidor.
* Haz una redirección de forma que cada vez que accedamos con HTTP se solicite el recurso utilizando HTTPS.
* Comprueba que el puerto web seguro (443) está abierto y en escucha y comprueba desde uno de los clientes **1 o 2 y 4** y **servidor1 (**servidor web**)** su funcionamiento.

**AMPLIACIÓN**

**2ª PARTE:** Implementa los sitios 4 y 5 en cloud (Windows 2016/19 y Ubuntu Server 18/20)

