

Instalar y configurar el servidor web Apache en Ubuntu 22.04 URL

1) Instalar Apache.

Instalamos con el comando “**apt install apache2**”.

Comprobamos si está activo con el comando “**systemctl status apache2**”.

2) Configurar el cortafuegos.

ATENCIÓN, SI VAMOS A PONERNOS A HACER COSAS CON LOS PUERTOS, NO TOCAR. SI NOS EQUIVOCAMOS, DESACTIVARLO: **ufw disable**

Para acceder a **Apache** desde el exterior, es necesario **abrir puertos específicos** en su sistema. Para ello, en primer lugar, consulta la lista de los perfiles de aplicaciones a los que necesitas acceder:

sudo ufw app list

```
CSA Seleccionar Símbolo del sistema - ssh usuario@10.2.7.108
[CSA jueves 25 septiembre 2025 15:43
[root@server2asir usuario]$ufw app list
Available applications:
  Apache
  Apache Full
  Apache Secure
  OpenSSH
CSA jueves 25 septiembre 2025 15:48
[root@server2asir usuario]$
```

sudo ufw allow 'Apache Full'

```
CSA jueves 25 septiembre 2025 15:48
[root@server2asir usuario]$sudo ufw allow 'Apache Full'
Rules updated
Rules updated (v6)
CSA jueves 25 septiembre 2025 15:50
[root@server2asir usuario]$
```

Después de hacerlo, verifique el estado actual del firewall:

sudo ufw status

Puede ser que no esté activo. Si lo activos y estamos por SSH se puede cortar la conexión.

```
CSA jueves 25 septiembre 2025 15:50
[root@server2asir usuario]$sudo ufw status
Status: inactive
CSA jueves 25 septiembre 2025 15:50
[root@server2asir usuario]$sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
CSA jueves 25 septiembre 2025 15:51
[root@server2asir usuario]$sudo ufw status
Status: active

To                         Action      From
--                         -----      ---
Apache Full                 ALLOW       Anywhere
Apache Full (v6)             ALLOW       Anywhere (v6)

CSA jueves 25 septiembre 2025 15:51
[root@server2asir usuario]$
```

Comandos “ufw”:

Habilitar: ufw enable.

Deshabilitar: ufw disable.

sudo ufw status verbose

Cómo configurar el servidor web Apache en Ubuntu 22.04

Antes de pasar a la configuración principal de Apache, verificaremos si el servicio “apache2” está actualmente activo o no:

systemctl status apache2

```
CSA jueves 25 septiembre 2025 15:51
[root@server2asir usuario]$systemctl status apache2
● apache2.service - The Apache HTTP Server
  Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset:>)
  Active: active (running) since Thu 2025-09-25 14:51:16 UTC; 1h 2min ago
    Docs: https://httpd.apache.org/docs/2.4/
   Process: 662 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCE)
   Process: 1246 ExecReload=/usr/sbin/apachectl graceful (code=exited, status=0/&)
 Main PID: 734 (apache2)
    Tasks: 55 (limit: 3425)
   Memory: 7.9M
      CPU: 518ms
     CGroup: /system.slice/apache2.service
             └─ 734 /usr/sbin/apache2 -k start
                 ├─1250 /usr/sbin/apache2 -k start
                 ├─1251 /usr/sbin/apache2 -k start

sep 25 14:51:16 server2asir systemd[1]: Starting The Apache HTTP Server...
sep 25 14:51:16 server2asir apachectl[681]: AH00558: apache2: Could not reliably >
sep 25 14:51:16 server2asir systemd[1]: Started The Apache HTTP Server.
sep 25 15:41:06 server2asir systemd[1]: Reloading The Apache HTTP Server...
sep 25 15:41:06 server2asir apachectl[1249]: AH00558: apache2: Could not reliably>
sep 25 15:41:06 server2asir systemd[1]: Reloaded The Apache HTTP Server.
lines 1-21/21 (END)
```

Abrimos **etc/apache2/apache2.conf**

En esta ocasión no cambiamos nada: solo se nos explica algunas opciones.

“Timeout”: Función del parámetro

- Timeout especifica **cuánto tiempo en segundos Apache espera:**
 - Recepción de una petición completa desde el cliente.
 - Envío de una respuesta al cliente.
 - Confirmación en ciertas operaciones con **proxies o backends**.

“KeepAlive”: Sirve para **permitir o desactivar conexiones persistentes** (que permiten que el mismo cliente realice varias peticiones en una única conexión TCP).

“MaxKeepAliveRequests”: controla **cuántas peticiones puede servir una misma conexión persistente** (Keep-Alive) antes de que se cierre.

“Include ports.conf”: Sirve para **incluir el archivo ports.conf** en la configuración principal de Apache.

Ese archivo (/etc/apache2/ports.conf) define en qué **puertos** escuchará Apache.

Es una forma modular de mantener ordenada la configuración:

- apache2.conf → configuración general.
- ports.conf → solo configuración de puertos.

Configuración de archivo de Puertos.

Ahora vamos a configurar el archivo de puertos: **/etc/apache2/ports.conf**

Normalmente, se usa solo el 80, pero para esta práctica hemos usado también el 90.

Cada vez que modifiques el archivo, debes usar:

systemctl restart apache2

systemctl reload apache2

```
GNU nano 6.2                               /etc/apache2/ports.conf
# If you just change the port or add more ports here, you will likely also
# have to change the VirtualHost statement in
# /etc/apache2/sites-enabled/000-default.conf

Listen 80
Listen 90

<IfModule ssl_module>
    Listen 443
</IfModule>

<IfModule mod_gnutls.c>
    Listen 443
</IfModule>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

En “Listen 80” quiere decir los puertos por los que puede/debe escuchar.

Configurar el host virtual en el servidor web Apache

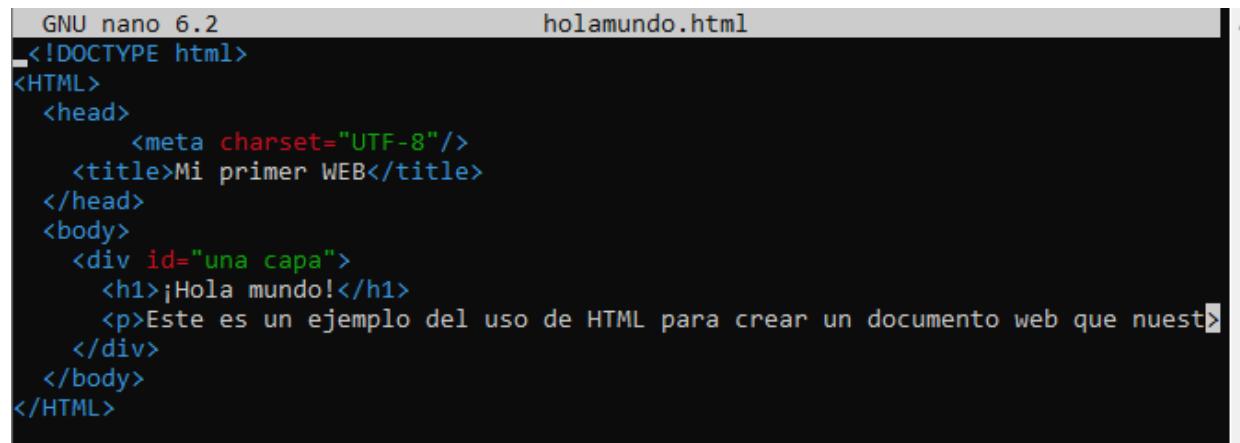
Para **establecer** un **host virtual** en **Apache**, cree un directorio que se pueda utilizar para almacenar datos en el sitio web. Para ello, nos desplazaremos al directorio “**/var/www**” utilizando el siguiente comando “**cd**”: **cd /var/www**

Ahora vamos al directorio “**html**” que debe de estar vacío o con un documento llamado “**index.html**” (este no se toca). Vamos a crear uno nuevo.

“**nano holamundo.html**”

E introducimos un código cualquiera de internet de ejemplo de “Hola Mundo”:

```
<!DOCTYPE html>
<HTML>
<head>
    <meta charset="UTF-8"/>
    <title>Mi primer WEB</title>
</head>
<body>
    <div id="una capa">
        <h1>¡Hola mundo!</h1>
        <p>Este es un ejemplo del uso de HTML para crear un documento web que nuest</p>
    </div>
</body>
</HTML>
```



The screenshot shows a terminal window with the title "GNU nano 6.2" and the file name "holamundo.html". The content of the file is the same HTML code provided in the previous text block, showing the creation of a document with a header and a paragraph.

```
GNU nano 6.2          holamundo.html
<!DOCTYPE html>
<HTML>
<head>
    <meta charset="UTF-8"/>
    <title>Mi primer WEB</title>
</head>
<body>
    <div id="una capa">
        <h1>¡Hola mundo!</h1>
        <p>Este es un ejemplo del uso de HTML para crear un documento web que nuest</p>
    </div>
</body>
</HTML>
```

Si en nuestro navegador web (en el host u otro equipo con conexión a la misma red) ponemos la ip del host y el nombre del archivo, nos saldrá:

<http://10.2.7.108/holamundo.html>



¡Hola mundo!

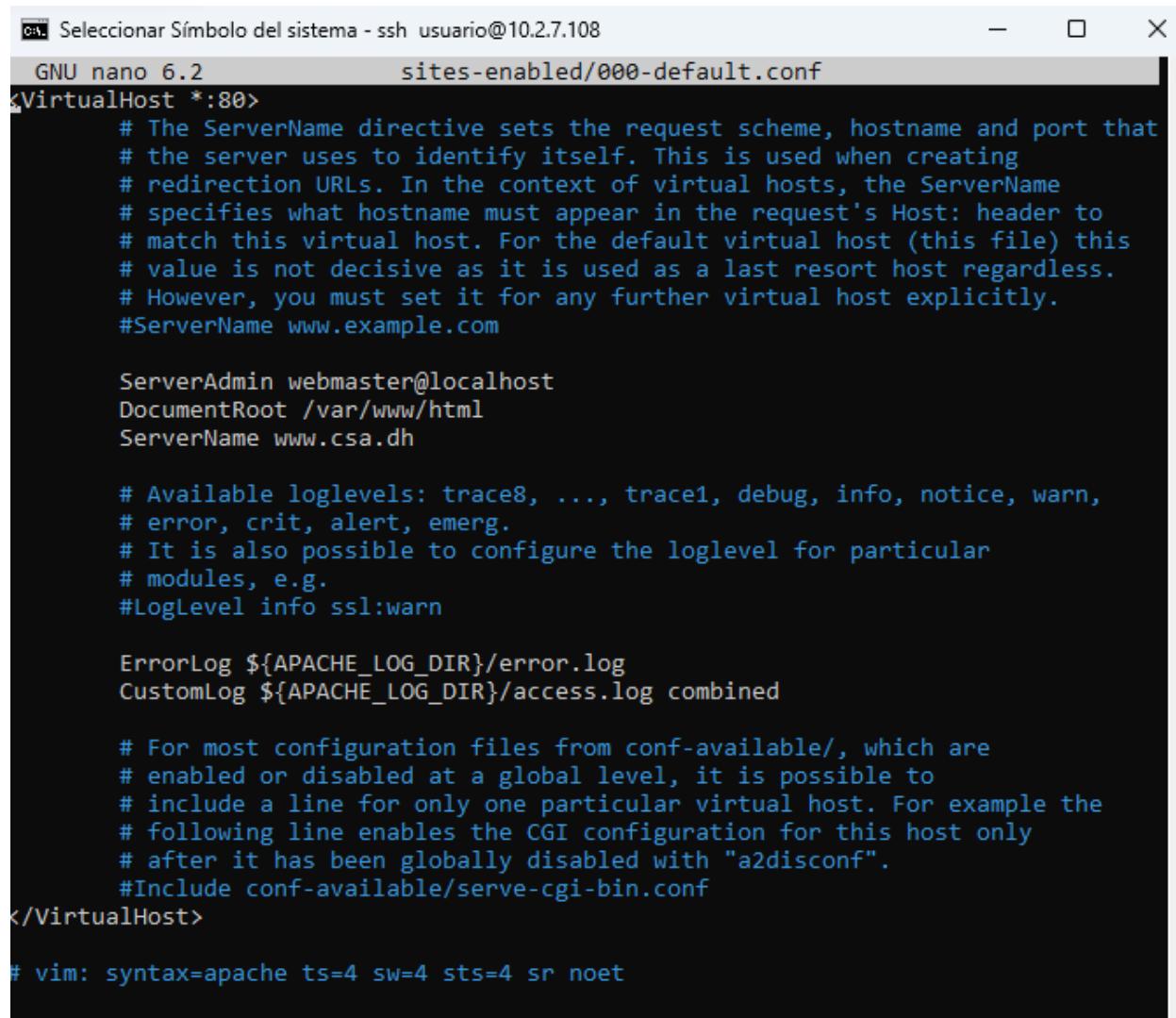
Este es un ejemplo del uso de HTML para crear un documento web que nuestro navegador es capaz de interpretar.

Ahora vamos a modificar los sitios.

/etc/apache2/sites-enabled: En esta carpeta se crearán enlaces simbólicos a los archivos de la carpeta sites-available para los hosts que queramos activar en cada momento.

/etc/apache/sites-available: En esta carpeta se encuentran todos los archivos de configuración de los hosts virtuales, cada host se corresponde con un archivo.

nano sites-enabled/000-default.conf



```
Selección Símbolo del sistema - ssh usuario@10.2.7.108
GNU nano 6.2           sites-enabled/000-default.conf
<VirtualHost *:80>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html
    ServerName www.csa.dh

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

- **<VirtualHost *:80>**

Indica que este VirtualHost escucha en todas las interfaces (*) en el puerto 80 (HTTP sin cifrar).

Si pusieras 192.168.1.100:80, solo respondería a esa IP concreta.

- **ServerAdmin webmaster@localhost**

Dirección de correo del administrador del sitio.

Aparece en páginas de error predeterminadas.

Normalmente se sustituye por un correo real, ejemplo: admin@miweb.com.

- **DocumentRoot /var/www/html**

Carpeta desde la cual Apache servirá los archivos web (la raíz del sitio).

En este caso: **/var/www/html**

Si alguien pide `http://www.csa.dh/index.html`, Apache buscará `/var/www/html/index.html`.

- **ServerName www.csa.dh**

Nombre de dominio que debe coincidir con el Host: enviado en la petición HTTP.

Sirve para diferenciar sitios si tienes varios VirtualHosts en el mismo servidor (ejemplo: `www.miweb.com` y `www.otraweb.com`).

Si no se configura bien, el sitio puede no responder como esperas.

Versión 2:

```
ca: Símbolo del sistema - ssh usuario@10.2.7.108
GNU nano 6.2           sites-enabled/sitio2.conf
<VirtualHost *:90>
    # The ServerName directive sets the request scheme, hostname and port that
    # the server uses to identify itself. This is used when creating
    # redirection URLs. In the context of virtual hosts, the ServerName
    # specifies what hostname must appear in the request's Host: header to
    # match this virtual host. For the default virtual host (this file) this
    # value is not decisive as it is used as a last resort host regardless.
    # However, you must set it for any further virtual host explicitly.
    #ServerName www.example.com

    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/sitio2
    ServerName www.sitio2.dh
    DirectoryIndex holamundo2.html

    # Available loglevels: trace8, ..., trace1, debug, info, notice, warn,
    # error, crit, alert, emerg.
    # It is also possible to configure the loglevel for particular
    # modules, e.g.
    #LogLevel info ssl:warn

    ErrorLog ${APACHE_LOG_DIR}/error.log
    CustomLog ${APACHE_LOG_DIR}/access.log combined

    # For most configuration files from conf-available/, which are
    # enabled or disabled at a global level, it is possible to
    # include a line for only one particular virtual host. For example the
    # following line enables the CGI configuration for this host only
    # after it has been globally disabled with "a2disconf".
    #Include conf-available/serve-cgi-bin.conf
</VirtualHost>

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet
```

- **<VirtualHost *:90>**

El VirtualHost escucha en todas las interfaces (*) pero en el puerto 90.

Es poco común, ya que HTTP estándar usa el puerto 80.

Para acceder habría que ponerlo en la URL: <http://www.sitio2.dh:90/>

- **ServerAdmin webmaster@localhost**

Dirección del administrador del sitio.

Se usa en páginas de error y mensajes de contacto.

- **DocumentRoot /var/www/sitio2**

Carpeta donde están los archivos web para este sitio.

En este caso: /var/www/sitio2.

- **ServerName www.sitio2.dh**

Nombre de dominio que identifica este VirtualHost.

Solo responderá a peticiones hechas con ese Host: en la cabecera.

- **DirectoryIndex holamundo2.html**

Define la página de inicio por defecto.

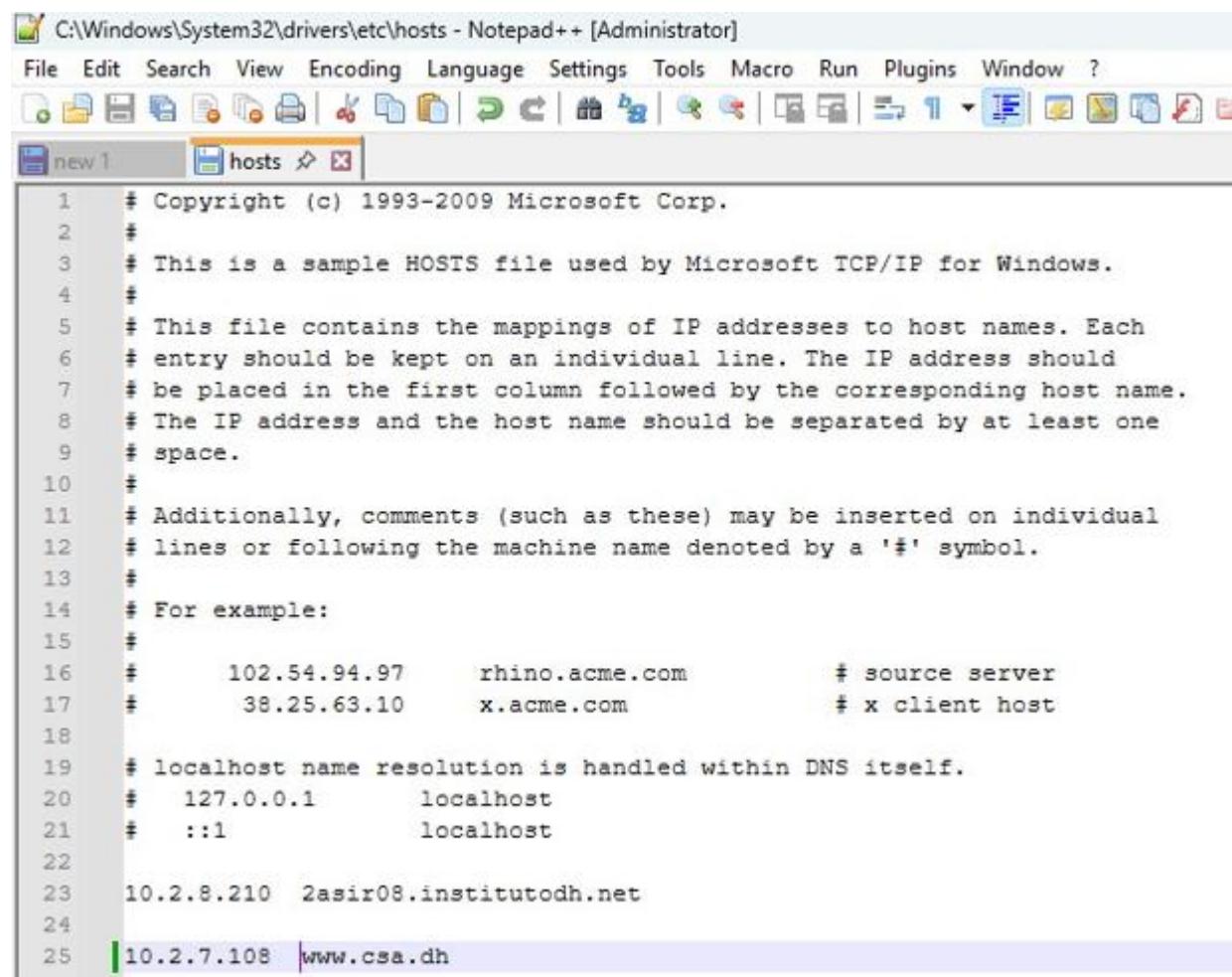
En lugar de buscar index.html o index.php, Apache servirá holamundo2.html.

Si visitas <http://www.sitio2.dh:90/>, Apache entregará /var/www/sitio2/holamundo2.html (si existe).

AHORA TENDRÍAMOS QUE HACER UN:

systemctl reload apache2

Después, si estamos en un cliente “Windows”, debemos ir a “C:\Windows\System32\drivers\etc\hosts” y al final poner la ip del host y el nombre del “ServerName”.



```
C:\Windows\System32\drivers\etc\hosts - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
new 1 hosts x

1 # Copyright (c) 1993-2009 Microsoft Corp.
2 #
3 # This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
4 #
5 # This file contains the mappings of IP addresses to host names. Each
6 # entry should be kept on an individual line. The IP address should
7 # be placed in the first column followed by the corresponding host name.
8 # The IP address and the host name should be separated by at least one
9 # space.
10 #
11 # Additionally, comments (such as these) may be inserted on individual
12 # lines or following the machine name denoted by a '#' symbol.
13 #
14 # For example:
15 #
16 #      102.54.94.97      rhino.acme.com          # source server
17 #      38.25.63.10      x.acme.com              # x client host
18 #
19 # localhost name resolution is handled within DNS itself.
20 #      127.0.0.1      localhost
21 #      ::1            localhost
22 #
23 10.2.8.210 2asir08.institutodh.net
24
25 10.2.7.108 www.csa.dh
```



Nota como el DNS puede resolver el nombre.

```
Microsoft Windows [Versión 10.0.26100.6584]
(c) Microsoft Corporation. Todos los derechos reservados.

C:\Users\UsuarioASIR>ping www.csa.dh

Haciendo ping a www.csa.dh [10.2.7.108] con 32 bytes de datos:
Respuesta desde 10.2.7.108: bytes=32 tiempo<1ms TTL=62

Estadísticas de ping para 10.2.7.108:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
              (0% perdidos),
    Tiempos aproximados de ida y vuelta en milisegundos:
        Mínimo = 0ms, Máximo = 0ms, Media = 0ms

C:\Users\UsuarioASIR>
```

Como hemos visto antes en la configuración de “sitio2”, sus archivos están **/var/www/sitio2/**

Los archivos de configuración que se encuentran en “etc/apache2/sites-available” o en “enabled”, debes usar el comando “a2ensite” para habilitarlo o “a2dissite” para deshabilitarlo. Y después un “systemctl reload apache2”.

```
CSA jueves 25 septiembre 2025 09:01
[root@server2asir www]$cd /etc/apache2/sites-available/
CSA jueves 25 septiembre 2025 09:03
[root@server2asir sites-available]$a2ensite sitio2.conf
Enabling site sitio2.
To activate the new configuration, you need to run:
    systemctl reload apache2
CSA jueves 25 septiembre 2025 09:03
```

El segundo sitio también hay que ponerlo en el archivo “hosts”:

```
10.2.7.108 www.csa.dh
10.2.7.108 www.sitio2.dh
```



En **/etc/apache2/mods-available/dir.conf** se encuentra la configuración por defecto de “**DirectoryIndex**”.

Pertenece al módulo mod_dir de Apache y define el **orden de búsqueda de páginas de inicio predeterminadas** (cuando alguien entra a un directorio sin especificar un archivo).

Si visitas `http://localhost/` y tu DocumentRoot es `/var/www/html`, Apache intentará servir en este orden:

1. `/var/www/html/index.html`
2. `/var/www/html/index.cgi`
3. `/var/www/html/index.pl`
4. `/var/www/html/index.php`
5. `/var/www/html/index.xhtml`
6. `/var/www/html/index.htm`

El primero que encuentre será el que muestre.

Modificación útil

Si quisieras que tu página de inicio fuese otra (por ejemplo `holamundo2.html` como en tu VirtualHost anterior), puedes editar `dir.conf` o sobreescribirlo dentro del bloque `<VirtualHost>` con:

`DirectoryIndex holamundo2.html`