Actividad 2b

1. Realiza los siguientes pasos y anótalos:

* Explora el fichero de configuración apache2.conf e identifica los valores de las directivas de serverroot, user, group y timeout.







* Cambia el valor del puerto por defecto para que sea 90 y reinicia el servidor para comprobar que funciona. En el manual, tienes explicado cómo hacerlo. Déjalo de nuevo como estaba.





* Observa la configuración del servidor virtual por defecto y determina los valores de Documentroot y ErrorLog y para qué sirven.

Documentroot: Esta directiva dice la carpeta donde se tienen que alojar los archivos para que los lea el apache.

ErrorLog: Esta directiva Dice la ruto donde se guarda el archivo con todos los errores que recibe apache.

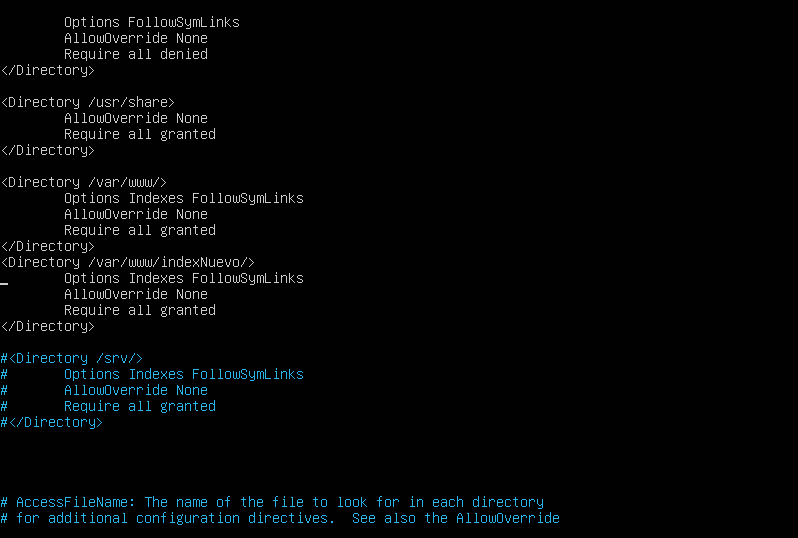
* Accede desde tu máquina host al servidor apache simplemente poniendo la IP y el puerto. ¿Qué página se abre por defecto? La directiva DirectoryIndex es la que determina este comportamiento. Averigua dónde se define (puedes hacerlo rápidamente con grep). Esta directiva también la puedes configurar en el propio servidor por defecto. Define otra página y comprueba que funciona.



* Si ahora nos cargamos esta página, ¿qué se muestra?

Se mostraría el index.html

* La directiva Directory del fichero apache2.conf define cómo se sirven los contenidos. Por defecto, lo que esté dentro de /var/www hereda su configuración. Prueba a crear un subdirectorio y pon dentro un fichero html con el mismo nombre que asociaste a la página por defecto hace dos pasos. Pon también una página index.html y determina qué página se abre desde el navegador (accediendo a ip:puerto/tu subdirectorio). Quita ambos ficheros y determina ahora qué se muestra.

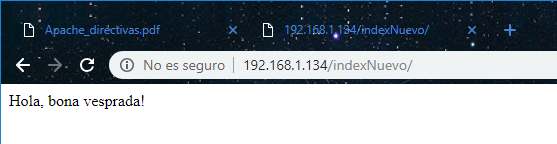


Falta indexNuevo/ficher que vols mostrar.

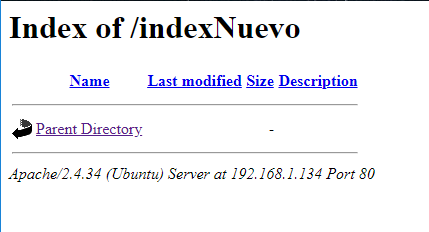
Posar el indexNuevo en html/

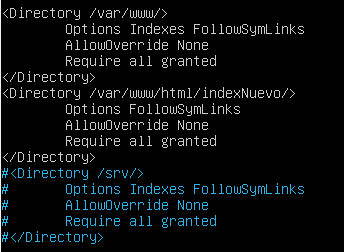
Mirar apache2.conf y mirar el servername.

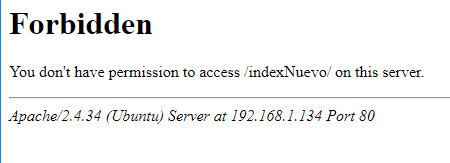
Sudo systemctl restart apache2



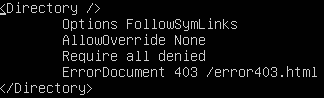
Al borrar los dos archivos que hay en indexNuevo/



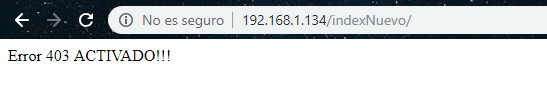
* Crea una nueva directiva Directory para /var/www/html/tu subdirectorio similar a la que tienes definida como /var/www pero sin la opción Indexes. Determina qué se muestra. 



* Crea un fichero html en /var/www/html para mostrar un error en caso de no encontrar la página. Utiliza la directiva ErrorDocument dentro del servidor por defecto para que acceda a esta página. Comprueba que funciona.

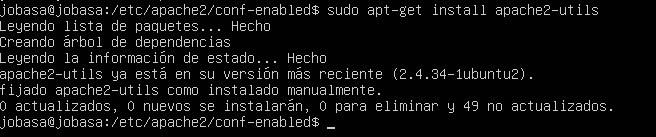




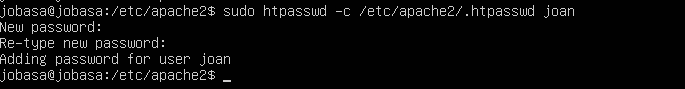


1. . En ocasiones necesitamos restringir el acceso a determinadas secciones de nuestro sitio web. Las propias aplicaciones suelen tener sus propios mecanismos de autenticación y autorización, pero el servidor web también puede usarse para especificar restricciones de acceso. Sigue los siguientes pasos y anótalos:

* Para restringir el acceso, crearemos un fichero de contraseñas mediante la herramienta htpasswd. Si no la tienes instalada, lo puedes hacer instalando el paquete apache2-utils



* Utiliza esta utilidad para crear un usuario con su contraseña. Ubica este fichero en un directorio con permisos para que cuando entremos desde fuera, podamos acceder a este fichero, por ejemplo en la siguiente ruta: sudo htpasswd -c /etc/apache2/.htpasswd usuario\_autenticado



* Una vez tenemos el fichero de contraseñas, tenemos que configurar Apache para que autentique a los usuarios antes de ofrecer el contenido restringido. Una opción para hacer esto es mediante un fichero .htaccess que nos permite definir items de configuración dentro de un directorio de contenidos y que Apache leerá cada vez que haya una petición a este directorio. Para ello, crea un fichero .htaccess en el directorio que quieras restringir (el directorio raíz de tu sitio web

u otro subdirectorio) con este contenido:

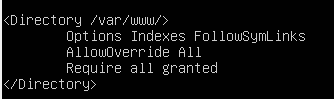
AuthType Basic

AuthName "Restricted Content"

AuthUserFile -directorio donde esté .htpasswdRequire valid-user



* Finalmente, modifica la directiva AllowOverride del fichero de configuración apache2.conf que hace referencia al directorio raíz /var/www para que tenga el valor All en vez de None.



* Reinicia el servidor y comprueba que el acceso a el directorio especificado está ahora restringido.
* A parte de usarlo para cuestiones de seguridad, el fichero .htaccess también sirve para otras cosas. Una de ellas es poder es redireccionar a los usuarios hace páginas personalizadas cuando se produce un error. Prueba a crear una página para el error 404 de páginas no encontradas, y redirige a esta página mediante el siguiente código de .htaccess: ErrorDocument 404 /error/error\_personalizado.html

