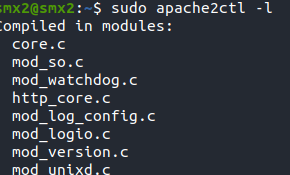
**M8 UF1 Activitat 3**

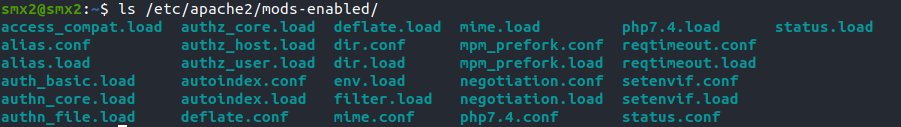
Consulta los módulos estáticos y los módulos dinámicos cargados por defecto en la versión de Apache instalada en la maquina Servidor LinuxXX. Posteriormente, investiga la funcionalidad del modulo userdir, habilítalo y prueba su funcionalidad.

1. **Módulos**
   1. Inicia sesión con un usuario con privilegios de administrador en el Servidor Linux.
   2. Comprueba los módulos estáticos que se han cargado al compilar el servidor ejecutando el siguiente comando: sudo apache2ctl –l

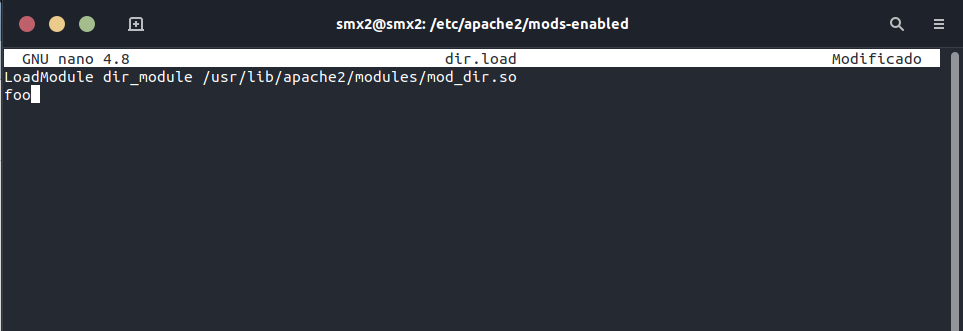


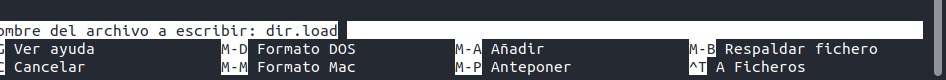
* 1. Comprueba los módulos que se han cargado dinámicamente al arrancar el servidor consultando el directorio /etc/apache2/mods-enable. Observa que los ficheros que aparecen en este directorio son enlaces simbólicos a ficheros de

/etc/apache2/mods-available.



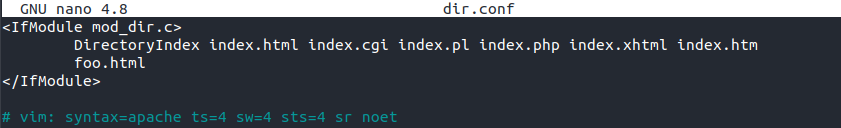
* 1. Edita uno de los ficheros .load (por ejemplo dir.load) y observa cómo se utiliza la directiva LoadModule, para cargar el modulo. Comprueba cual es la ruta donde está el código del modulo (archivo .so).

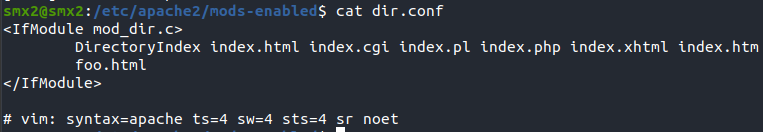




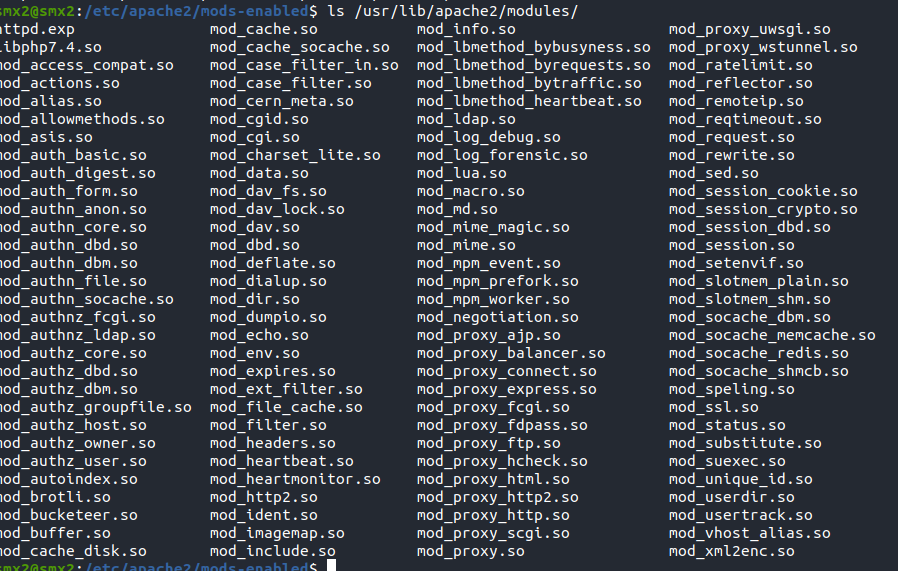


* 1. Edita uno de los ficheros .conf (por ejemplo dir.conf) y observa cómo se añaden directivas dentro de una declaración <IfModule nombremodulo>... </IfModule>, que se ejecuta si se carga el modulo.

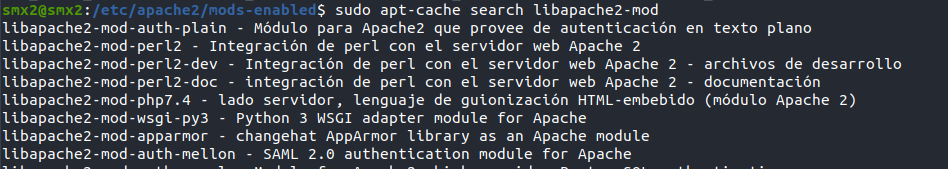




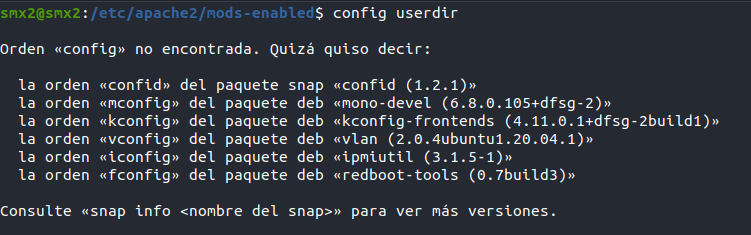
* 1. Consulta el directorio /usr/lib/apache2/modules/ y observa los módulos disponibles para cargar.



* 1. Ejecuta el siguiente comando para mostrar los paquetes disponibles en los repositorios de Ubuntu que permiten instalar módulos adicionales en Apache. sudo apt-cache search libapache2-mod

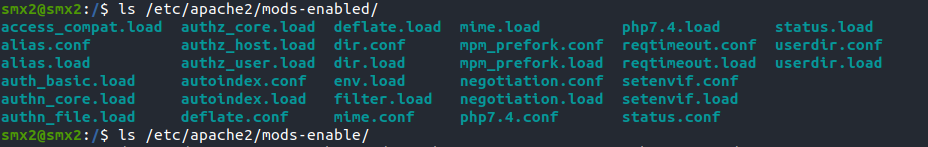


1. **Modulo userdir**
   1. Busca información de la utilidad del modulo userdir de apache.



* 1. Comprueba consultando el directorio /etc/apache2/mods-enabled que el modulo userdir no está habilitado.
  2. Habilita el modulo ejecutando el siguiente comando: sudo a2enmod userdir
  3. Verifica que dentro del directorio /etc/apache2/mods-enabled se han creado enlaces simbólicos del modulo userdir (ficheros .conf y .load) hacia

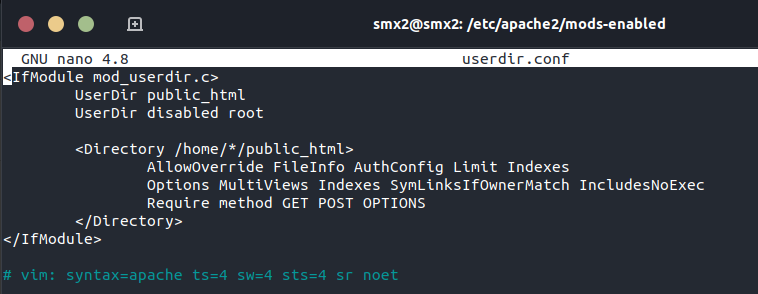
/etc/apache2/mods-availables.



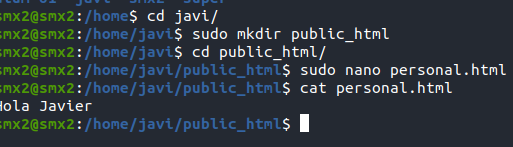
* 1. Reinicia el servidor para que los cambios tengan efecto.



* 1. Consulta el fichero /etc/apache2/mods-enabled/userdir.conf. Observa que está habilitado el uso de directorios personales para todos los usuarios excepto para root y que public\_html es el nombre del subdirectorio que pueden crear los usuarios en su directorio home para poner sus páginas personales.



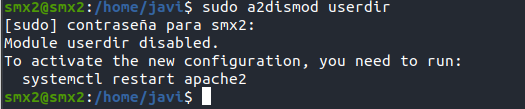
* 1. Crea un usuario y crea el directorio /home/<nombre\_usuario>/public\_html. Dentro del directorio crea un fichero denominado personal.html con el contenido que quieras.



* 1. Accede desde la máquina real a la direción <http://localhost/>~<nom\_user> i luego http://IP-Máquina/~<nom\_user>



* 1. Una vez comprobado que funciona, deshabilita el modulo ejecutando el siguiente comando, y después reinicia el servidor: sudo a2dismod userdir



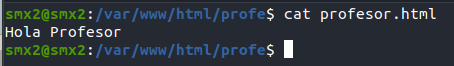
1. **Control de acceso por IP y nombre de dominio**

En la maquina Servidor Linux crea el directorio */var/www/html/profe* y configura Apache para que solo se pueda acceder desde la maquina virtual.

* 1. Inicia sesión con un usuario con privilegios de administrador en el Servidor Linux.



* 1. Dentro del directorio */var/www/html/profe* crea un documento denominado profesor.html con el contenido que quieras.



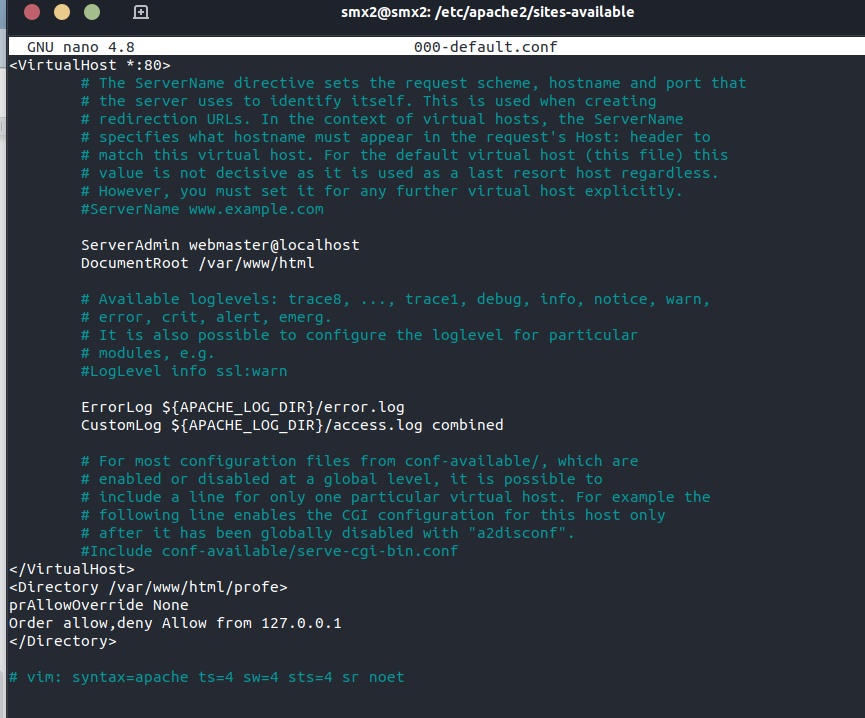
* 1. Edita el fichero de configuración /etc/apache2/sites-available/default y utiliza la directiva <Directory> junto con las directivas Order, Allow y Deny para denegar el acceso al directorio a todos los equipos excepto al local.

*<Directory /var/www/html/profe>*

*prAllowOverride None*

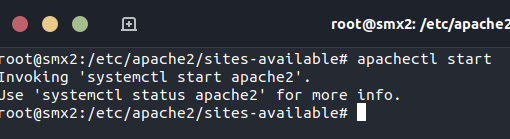
*Order allow,deny Allow from 127.0.0.1*

*</Directory>*

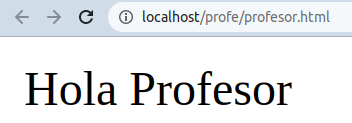
**

* 1. Reinicia el servidor para que los cambios tengan efecto.

Nota: Si aparece un error edita apache2.conf y añade “ServerName localhost”.



* 1. [Comprueba que se puede acceder a http://localhost/profe/profesor.html](http://localhost/profe/profesor.html) desde la maquina virtual, pero no así desde la maquina real (comprueba que desde la maquina real sí que podemos acceder a index.html, para descartar otros errores, y que solo está prohibido el acceso a la pagina que hemos configurado).

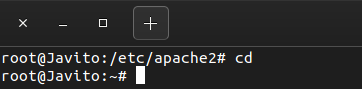


1. **Autenticación y autorización Basic y Digest**

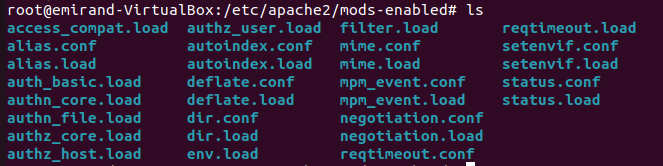
En la máquina Servidor Linux configura la autentificación HTTP Basic sobre el directorio /var/www/profe para que solo puedan acceder los usuarios profesor1 y profesor2. Configura la autentificación HTTP Digest sobre el directorio

/var/www/departamento para que solo puedan acceder los usuarios admin1 y admin2

* 1. Inicia sesión con un usuario con privilegios de administrador en el Servidor Linux

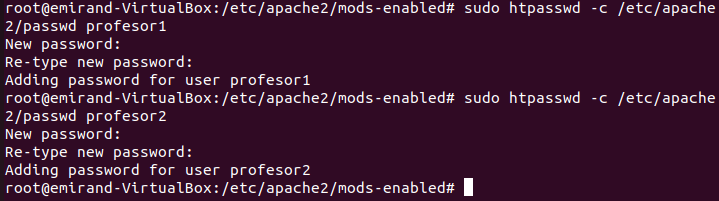


* 1. Comprueba consultando el directorio /etc/apache2/mods-enabled, que el modulo auth\_basic esta habilitado.



* 1. Para usar la autentificación basic hay que crear un fichero accesible por Apache en el que se guardaran los usuarios y sus contraseñas. Para crear este fichero se utilizara el comando htpasswd (si no se reconoce el comando deberás instalar apache2-utils)

1. Crea el fichero y añade el usuario profesor1 (la opción –c es para crear el fichero) sudo htpasswd –c /etc/apache2/passwd profesor1
2. Añade el usuario profesor2 (no se usa el comand –c porque el fichero ya existe) sudo htpasswd /etc/apache2/passwd profesor2



* 1. Edita el fichero de configuración /etc/apache2/sites-available/default y permite el acceso a directorio /var/www/profesor a los usuarios profesor1 y profesor2.

*<Directory /var/www/html/profe>*

*Options Indexes FollowSymLinks MultiViews AllowOverride None*

*Order allow,deny Allow from 127.0.0.1 AuthType Basic*

*AuthName "Acces restringit" AuthUserFile /etc/apache2/passwd*

*<RequireALL>*

*Require user profesor2 profesor1*

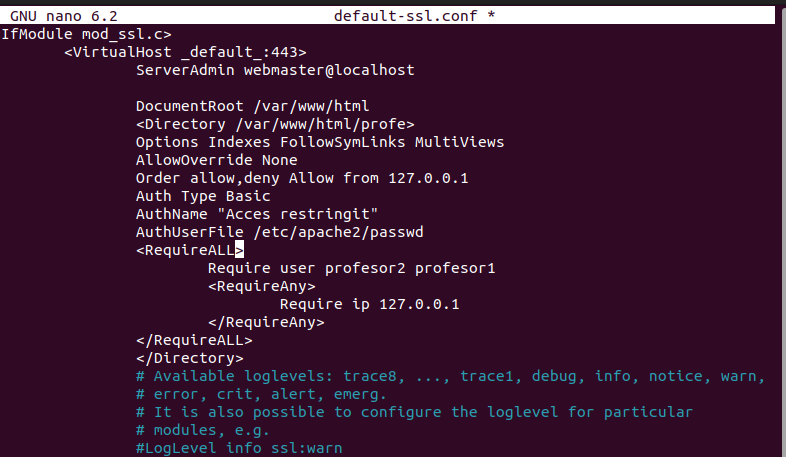
*<RequireAny>*

*Require ip 127.0.0.1*

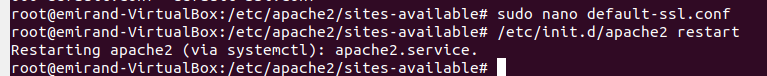
*</RequireAny>*

*</RequireALL>*

*</Directory>*

**

* 1. Reinicia el servidor para que los cambios tengan efecto.

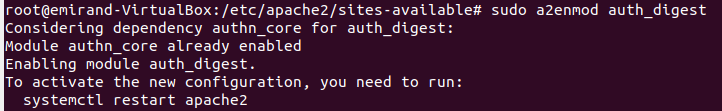


* 1. Accede con el navegador desde la maquina virtual (recuerda que tiene el acceso restringido por IP) a la pagina localhost/profe/profesor.html con el usuario profesor1. Intenta el acceso con otro usuario

**Autentificación Digest**

* 1. Crea el directorio /var/www/html/departamento. Dentro de este un departamento.html con el contenido que creas.
  2. Habilita auth\_digest.

sudo a2enmod auth\_digest



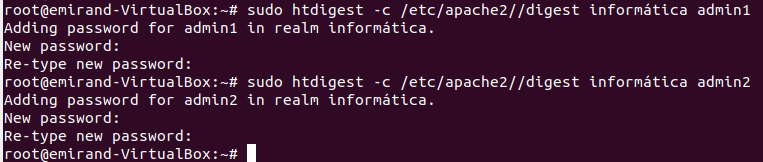
* 1. Reinicia el servidor.



* 1. Para usar autenticación digest hay que crear un fichero accesible por Apache en el que se guardarán las contraseñas asociadas al dominio (realm).

Creamos admin1 y admin2 con la misma idea anterior.

sudo htdigest -c /etc/apache2/digest informática admin1 sudo htdigest -c /etc/apache2/digest informática admin2



* 1. Edita /etc/apache2/sites-avilable/000-default.conf y permite acceso a los usuarios admin1 y admin2 a /var/www/html/departamento.

<Directory /var/www/html/departamento>

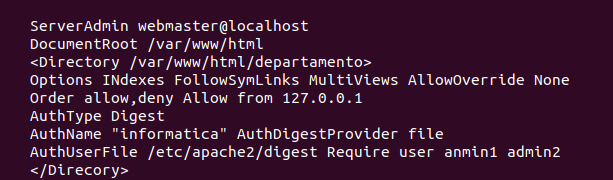
Options Indexes FollowSymLinks MultiViews AllowOverride None

Order allow,deny Allow from 127.0.0.1 AuthType Digest

AuthName "informatica" AuthDigestProvider file

AuthUserFile /etc/apache2/digest Require user admin1 admin2

</Directory>



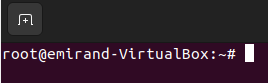
1. **Ficheros .htaccess**

En la maquina ServidorLinux:

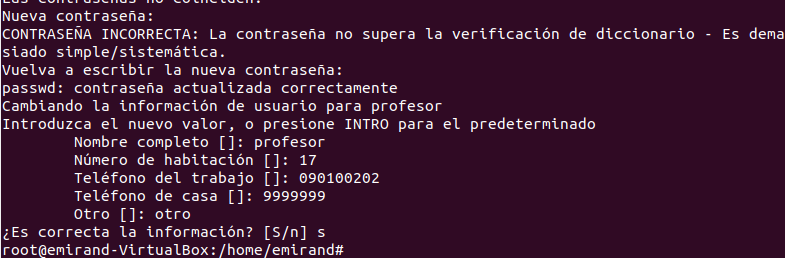
* Habilita en apache el uso de ficheros de configuración personalizada de directorios (.htaccess) en el directorio **/home/profesor/blog** para que sea el propio usuario profesor el que pueda controlar como sirve Apache los contenidos de ese directorio.
* Configura el alias /blog que permita el acceder al directorio /home/profesor/blog - Como usuario profesor haz uso del fichero .htaccess y configura sobre el directorio

/home/profesor/blog.

* Que solo se pueda acceder desde la maquina virtual - Autentificacion HTTP Digest para que solo pueda acceder el usuario blog.
  1. Inicia sesión con un usuario con privilegios de administrador en el Servidor Linux.



* 1. Crea el usuario profesor. sudo adduser profesor



* 1. Edita el fichero de configuración /etc/sites-available/default. Crea el alias y habilita el uso de ficheros .htaccess permitiendo sobreescribir todas las directivas en el directorio /home/profesor/blog. Elimina las directivas anteriores y añade la directiva AllowOverride All



* 1. Reinicia el servidor para que los cambios tengan efecto



* 1. Inicia sesión en ServidorLinux como usuario profesor.

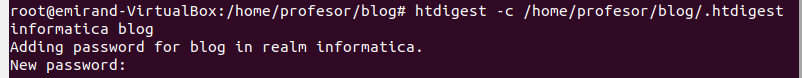


* 1. Crea el directorio /home/profesor/blog. Crea dentro el fichero blog.html con el contenido que quieras.

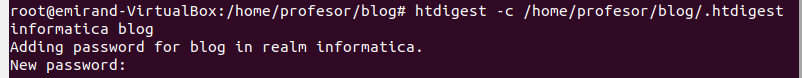


* 1. Crea el fichero /home/profesor/blog/.htdigest y añade al usuario blog.

htdigest –c /home/profesor/blog/.htdigest informática blog



* 1. Crea el fichero /home/profesor/blog/.htaccess y añade las directivas para realizar la configuración pedida (no es necesario añadir la directiva <Directory> porque el fichero ya está en el directorio en el que se aplicara su configuración.



* 1. Comprueba si se puede acceder desde la maquina real desde la maquina virtual, y con varios usuarios, incluido el usuario blog.