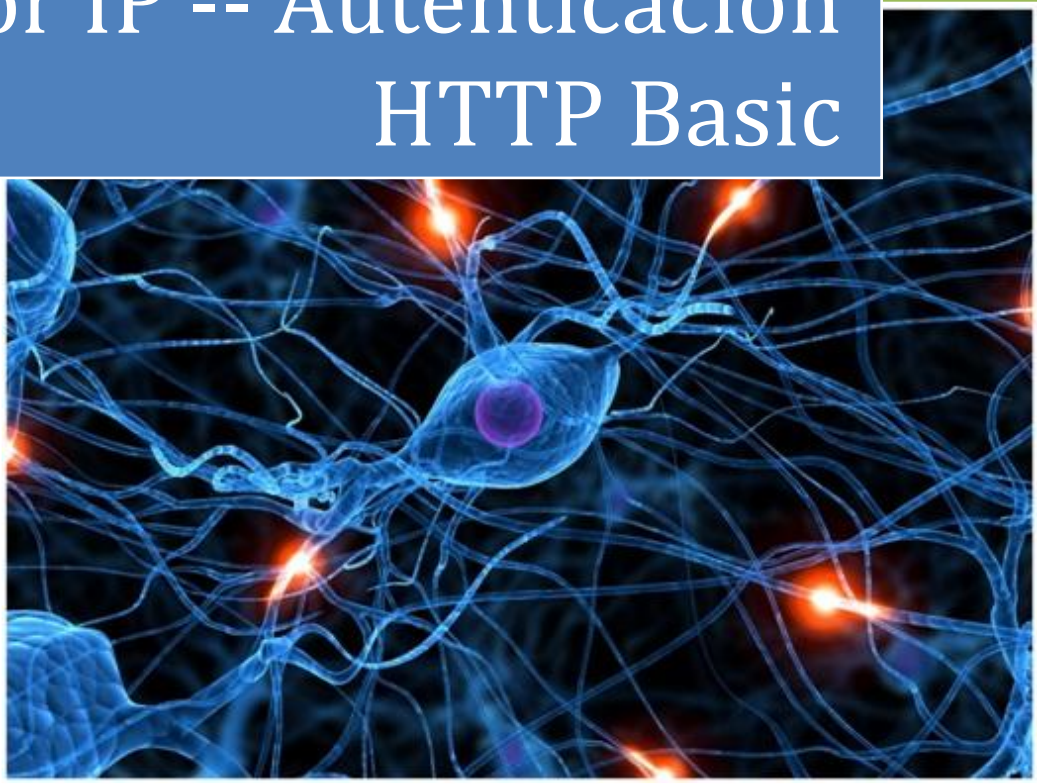
2011-2012

Actividad 8 Administración servidor Web HTTP (Apache2) en Ubuntu Server --Control de acceso por IP -- Autenticación HTTP Basic



José Jiménez Arias

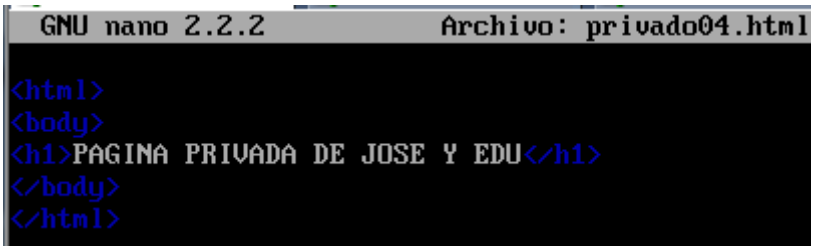
IES Gregorio Prieto

2011-2012

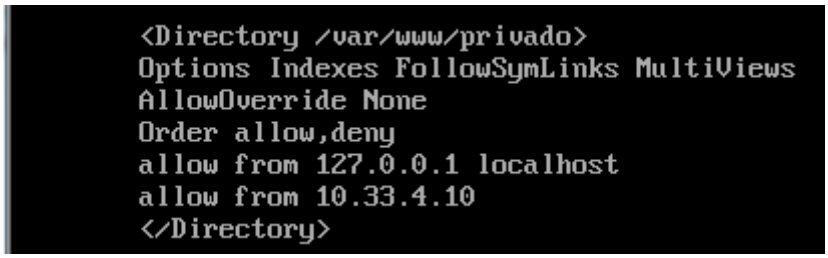
Es posible definir qué IPs/nombres de dominios pueden accede a un recurso del servidor utilizando las directivas Order, Allow y Deny dentro de secciones <Directory>, <File>, etc.

Cuando se realiza la conexión desde el cliente, el servidor comprueba si el acceso al recurso está autorizado. Si es así devuelve el recurso solicitado y si no, un código de error (403, Forbidden).

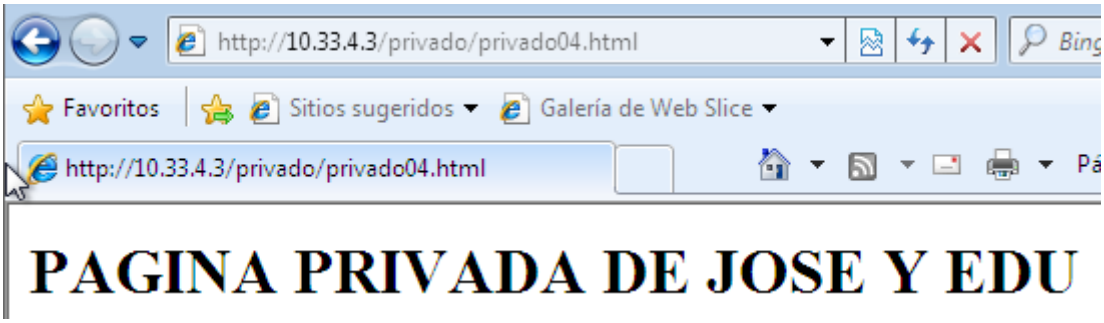
Iniciamos sesión y creamos el directorio /var/www/privado y dentro de este directorio creamos el fichero privado01.html



Editamos el fichero de configuración /etc/apache2/sites-available/default y utilizamos la sentencia <Directory> para denegar el acceso al directorio a todos los equipos excepto al local y a w704



Reiniciamos el servidor y comprobamos que podemos acceder al nuevo directorio desde el equipo w7 (con ip 10.33.4.10 permitida anteriormente).



A continuación procedemos a acceder desde otro equipo cuya Ip no esté en la lista anteriormente definida como por ejemplo debian04 (con ip 10.33.4.30)

A continuación, consultamos el directorio /etc/apache/mods-enabled para comprobar que el módulo auth_basic (relacionado con la autenticación)está habilitado.

```
root@ubuntusrv04:/etc/apache2/mods-enabled# ls
alias.conf            autoindex.conf  env.load          setenvif.load
alias.load            autoindex.load  mime.conf         status.conf
auth_basic.load       cgid.conf       mime.load         status.load
authn_file.load       cgid.load       negotiation.conf  userdir.conf
authz_default.load    deflate.conf    negotiation.load  userdir.load
authz_groupfile.load  deflate.load    reqtimeout.conf
authz_host.load       dir.conf        reqtimeout.load
authz_user.load       dir.load        setenvif.conf
```

Para usar la autenticación básica hay que crear un fichero accesible por Apache en el que se guardarán los usuarios y sus contraseñas. Para crear este fichero utilizaremos el comando htpasswd, observamos sus parametros más importantes:

```
root@ubuntusrv04:/etc/apache2# htpasswd
Usage:
        htpasswd [-cmdpsD] passwordfile username
        htpasswd -b[cmdpsD] passwordfile username password

        htpasswd -n[mdps] username
        htpasswd -nb[mdps] username password
 -c  Create a new file.
 -n  Don't update file; display results on stdout.
 -m  Force MD5 encryption of the password.
 -d  Force CRYPT encryption of the password (default).
 -p  Do not encrypt the password (plaintext).
 -s  Force SHA encryption of the password.
 -b  Use the password from the command line rather than prompting for it.
 -D  Delete the specified user.
On Windows, NetWare and TPF systems the '-m' flag is used by default.
On all other systems, the '-p' flag will probably not work.
```

Creamos el fichero "-c" y añadimos el usuario "mortadelo", con la contraseña "morta".

```
root@ubuntusrv04:/etc/apache2# htpasswd -c /etc/apache2/passwd mortadelo
New password:
Re-type new password:
Adding password for user mortadelo
```

Añadimos un nuevo usuario "filemon", con la contraseña "file".

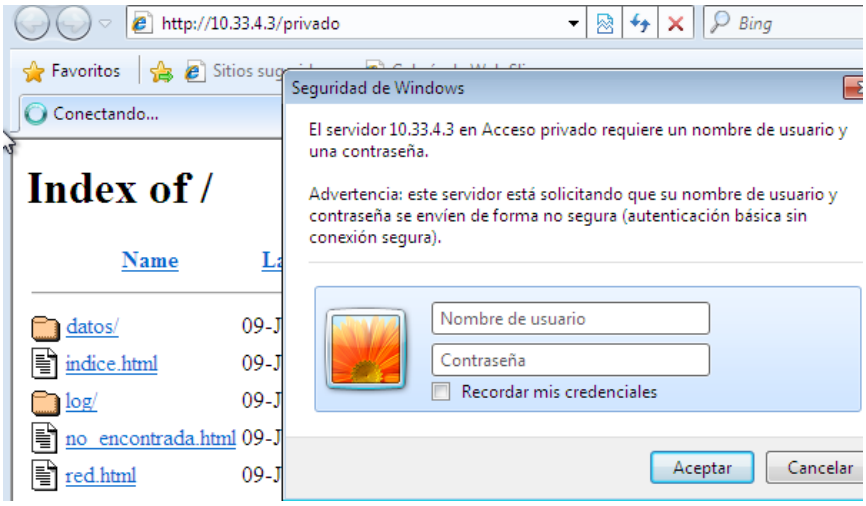
```
root@ubuntusrv04:/etc/apache2# htpasswd /etc/apache2/passwd filemon
New password:
Re-type new password:
Adding password for user filemon
```

A continuación editamos el fichero de configuración /etc/apache2/sites-available/default para permitir el acceso al directorio privado al usuario mortadelo, pero no a filemon.

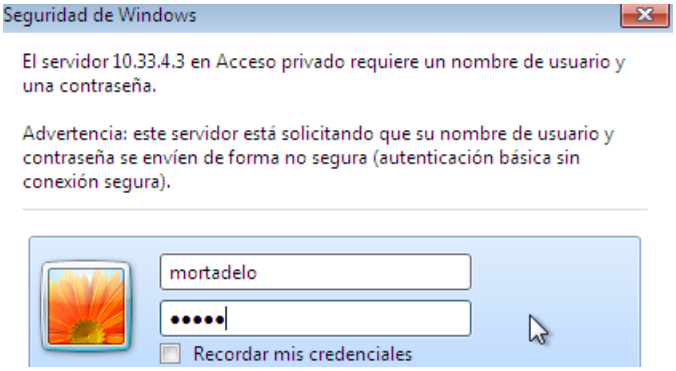
```
<Directory /var/www/privado>
Options Indexes FollowSymLinks MultiViews
AllowOverride None
Order allow,deny
allow from 127.0.0.1 localhost
allow from 10.33.4.10
AuthName "Acceso privado"
AuthType Basic
AuthUserFile /etc/apache2/passwd
Require user mortadelo
</Directory>
```

AuthName (nombre que le damos)
AuthType (tipo de autenticación)
AuthUserFile (directorio donde se encuentra el fichero de usu y contra)
Require user (usuarios permitidos "de dentro del fichero")

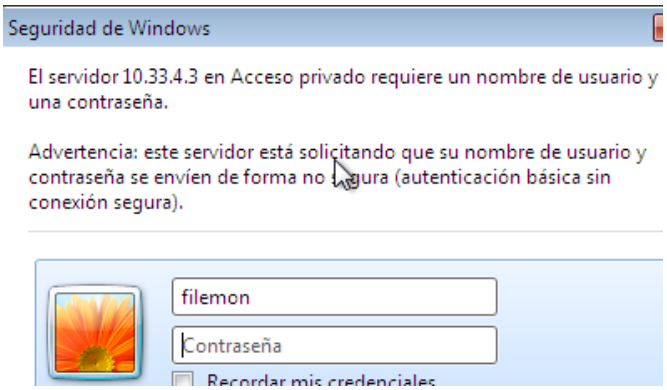
Comprobamos que al acceder al directorio privado de nuestro servidor, ahora nos solicita usuario y contraseña.



Ahora procedemos a comprobar que con mortadelo tenemos acceso:



Sin embargo, intentamos acceder con filemon (no lo incluimos en la directiva Require user) y observamos como el usuario filemon no puede acceder de ninguna forma.



Nota: Si el usuario está incluido en la directiva Require user, pero no está incluido en el fichero de la directiva AuthUserFile /etc/apache2/passwd, este tampoco podrá acceder, es decir, un usuario debe aparecer en ambos lugares para que tenga acceso.