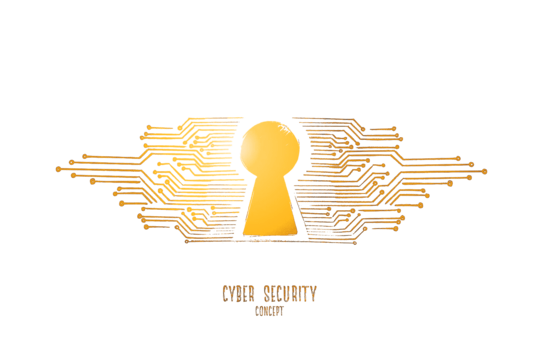
Bastionado de redes 

y sistemas.



# Hardening Apache

[**Hardening Apache 2**](#_xfpg6pduf6p7)

[Archivos de interés 2](#_sppmrpxi8ai)

[1. - Ocultar información sobre la versión. 3](#_y30kuxiw7gvx)

[2.- Probar que a realizar un enlace simbólico en el index y acceder a una zona no permitida 4](#_x8yg7fz6int4)

[2.1.- Corregir este hueco de seguridad 6](#_hco9l1k4rc6c)

[3.- Crear una zona privada con acceso usuario y contraseña 6](#_174vj5owzcv6)

[4.- Realizar ataque SYN FLOOD con hping3. 8](#_x2rdo3dugel5)

[6.- Impedir que el servidor apache se acceda desde determinadas IPs. 14](#_75r49pzfv0n7)

[Bibliografía 16](#_x4hh1070gnax)

| Archivos de interés |
| --- |
| /etc/apache2/ |
| **apache2.conf** |
| Configuración principal desde donde se cargarán todos los ficheros necesarios cuando se inicie el servidor web. |
| **ports.conf** |
| Determina puertos de escucha |
| **Mods-available, conf-available, sites-available** |
| Configuración para gestionar módulos, global y hosts virtuales de forma separada |
| **Mods-enable, conf-enabled, sites-enables** |
| Enlaces simbólicos desde \*-available (mods,conf y sites). |
| **envvars** |
| Variables de entornos usadas por apache2.conf |

Retomo la máquina para generar PKI’s de la práctica anterior.

| Es recomendable sobreescribir la página de inicio: |
| --- |
| **Sudo echo “funciono!” > /var/www/html/index.html** |
| 1. - Ocultar información sobre la versión. |
| 1. Abrimos el archivo de configuración de Apache |
| Centos  **nano /etc/httpd/conf/httpd.conf**  Ubuntu  **vim /etc/apache2/conf-enabled/security.conf** |
| 1. En la sección **ServerTokens** descomentar “ServerTokens OS” o aña de la línea: |
| **ServerTokens Prod** |
|  |
| 1. En el mismo archivo, editar la siguiente línea tal que: |
| **ServerSignature Off** |
|  |
| Esto ocultará mensajes del servidor como Internal Server Error (500) |
| 1. Reiniciar y comprobar |
| **# Shutdown -r now**  **service apache2 restart** |
| **curl --head localhost** |
|  |
| 2.- Probar que a realizar un enlace simbólico en el index y acceder a una zona no permitida |
| Generamos el enlace simbólico: |
| **Mkdir /tmp/test**  **echo “symbolic link data” > /test/test.txt**  **ln -s /tmp/test/test.txt /var/www/html/<dominio>** |
| Desde el browser de está manera no funcionará. Porque las carpetas que permiten links links simbólicos están predefinidas en:  **/etc/apache2/apache2.conf**  Por defecto se encuentra:   * / * /usr/share * /var/www/ |
| Por lo que según está configuración para implementar el acceso a un recurso compartido podríamos hacer:  **mkdir /rootlink/**  **echo “syslink” > /rootlink/roottest**  **ln -s /rootlink/roottest roottest /var/www/html/<dominio>/ roottest** |
| Resultado |
| Esto presenta una vulnerabilidad muy grave de configuración por defecto. Si por un casual se llegase a ejecutar código remoto y se crea un link simbólico a / con solamente:  **ln -s / barbaridad**  , se podŕa analizar el sistema en búsqueda de escalar privilegios. |
| 2.1.- Corregir este hueco de seguridad |
| Pues editando las rutas en las que sí queramos establecer enlaces y borrando las que no. A modo de lista blanca.  Por ejemplo:  <Directory /media/newhd/test\_web/>  Options Indexes FollowSymLinks  AllowOverride None  Require all granted  </Directory> |

| 3.- Crear una zona privada con acceso usuario y contraseña |
| --- |
| htpasswd -c /etc/apache2/.htpasswd ivan |
| cat /etc/apache2/.htpasswd |
|  |
| Vim /etc/apache2/dominios-ssl.conf |
|  |
| \*\*Crear la carpeta restringida bajo la carpeta raíz de virtualhost de Apache |
| mkdir /var/www/html/dominio1/restrictedFolder |
| sudo service apache2 restart |
| https://localhost/restrictedFolder |
|  |
|  |
| 4.- Realizar ataque SYN FLOOD con hping3. |
| Ejemplo de uso de hping3 sobre un equipo genérico dentro de la red:  hping3 -S -p 443 --flood 192.168.169.178:   1. sudo: Ejecuta el comando con privilegios de superusuario. 2. hping3: Es el nombre del programa, en este caso, la herramienta de envío de paquetes. 3. -S: Especifica el tipo de paquete que se enviará, en este caso, paquetes TCP SYN. Los paquetes SYN son parte del proceso de establecimiento de conexión en el protocolo TCP. 4. -p 443: Indica el puerto al que se enviarán los paquetes. En este caso, se utiliza el puerto 443, que es comúnmente asociado con conexiones seguras HTTPS. 5. --flood: Esta opción indica que la herramienta debe inundar el host de destino con paquetes lo más rápido posible, sin esperar respuestas. 6. 192.168.169.178: Es la dirección IP del host de destino al que se enviarán los paquetes. |
| 5.- Realizar hotlinking y evitar hotlinking. |
| Hotlinking externo |
|  |
|  |
|  |
| Hotlinking hacia nuestro servidor |
| Añadimos una imagen local |
|  |
|  |
| Accedemos desde nuestra ip para actuar como un tercero |
|  |
| Esto significa que somos sensibles al hotlinking |
|  |
| Como evitarlo |
| Una forma de evitarlo es mediante el módulo **a2emod** |
|  |
|  |
|  |
| .htacess |
|  |
| 1. <IfModule mod\_rewrite.c>: Verifica si el módulo de Apache llamado mod\_rewrite está habilitado. Este módulo es necesario para la manipulación de URLs mediante reglas de reescritura. 2. RewriteEngine On: Habilita el motor de reescritura de URLs. Es necesario para que las reglas de reescritura tengan efecto. 3. # Lista de dominios permitidos: Un comentario que indica que a continuación se especificarán los dominios permitidos. 4. RewriteCond %{HTTP\_REFERER} !^$: Esta línea establece una condición basada en el encabezado Referer del navegador. El Referer indica desde qué URL se hizo la solicitud. Esta condición verifica si el Referer no está vacío, lo que significa que la solicitud debe provenir de algún lugar. 5. RewriteCond %{HTTP\_REFERER} !^http(s)?://(www\.)?dominio1.com [NC]: Esta condición verifica si el Referer no coincide con tu propio dominio. Reemplaza tudominio.com con tu propio nombre de dominio. El modificador [NC] indica que la comparación no distingue entre mayúsculas y minúsculas. 6. RewriteRule \.(jpg|jpeg|png|gif)$ - [NC,F,L]: Esta regla se aplica sólo si las condiciones anteriores se cumplen. La regla bloquea las solicitudes de archivos con extensiones .jpg, .jpeg, .png y .gif. Si una solicitud coincide con esta regla, se le enviará un código de estado HTTP 403 (Forbidden), y la regla terminará con la flag [F].   La flag [L] indica que no se deben procesar más reglas si esta se cumple. |
| Vista desde máquina externa |
|  |
|  |
| Desde una máquina externa |
|  |
| 6.- Impedir que el servidor apache se acceda desde determinadas IPs. |
| Veo especialmente útil el bloqueo para ip’s de administración. |
| Por lo que, filtrar los accesos desde ip’s diferentes al 127.0.0.0 y una ip designada para la gestión que corresponderá con mi teléfono móvil. |
| **echo gestion > /var/www/html/dominio1/gestion.html** |
| **/etc/apache2/sites-enabled/dominios-ssl.conf/** |
| Añadimos las siguientes líneas al host virtual |
| <Location />  <RequireAll>  Require all granted  Include /etc/apache2/blacklist  </RequireAll>  </Location>  </VirtualHost> |
|  |
| Hay que crear la lista negra |
| **touch /etc/apache2/blacklist** |
| **Sudo chown www-data.www-data /etc/apache2/blacklist** |
|  |
| Ip fuera de la lista negra ( 192.168.244.103 ) |
|  |
| Desde una ip en la lista negra (192.168.244.152) |
|  |

## Bibliografía

* <https://www.incibe.es/en/incibe-cert/publications/webinars/apache-basic-hardening>
* <https://www.inmotionhosting.com/support/server/apache/hide-apache-version-and-linux-os/>