Configuración de Headers de Seguridad en Nginx con DirectAdmin

Introducción

Esta guía documenta cómo configurar headers de seguridad en servidores con DirectAdmin, tanto a nivel global como para dominios específicos que requieran exclusiones.

Última actualización: 25/12/2024 GMT+1

Configuración Global

La configuración global se aplica a **todos los dominios** del servidor mediante el archivo:

```
/usr/local/directadmin/data/templates/custom/cust_nginx.CUSTOM.post
```

Crear el directorio custom (si no existe)

```
mkdir -p /usr/local/directadmin/data/templates/custom
```

Ejemplo de configuración global

```
# Security Headers - Global
add_header X-Content-Type-Options "nosniff" always;
add_header X-Frame-Options "SAMEORIGIN" always;
add_header X-XSS-Protection "1; mode=block" always;
add_header Referrer-Policy "strict-origin-when-cross-origin" always;
add_header Permissions-Policy "geolocation=(), microphone=(), camera=()" always;
```

```
add_header Strict-Transport-Security "max-age=31536000; includeSubDomains" always;
add_header Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline'
'unsafe-eval' https:; style-src 'self' 'unsafe-inline' https:; img-src 'self' data: https:;
font-src 'self' data: https:; connect-src 'self' https:; frame-src 'self'
https://www.youtube.com https://www.google.com; frame-ancestors 'self'; base-uri 'self'; form-
action 'self';" always;

# Custom error pages for nginx-only domains
error_page 400 /error/400.html;
error_page 401 /error/401.html;
error_page 403 /error/403.html;
error_page 404 /error/404.html;
error_page 405 /error/405.html;
error_page 406 /error/406.html;
error_page 413 /error/413.html;
error_page 422 /error/422.html;
error_page 429 /error/429.html;
error_page 431 /error/431.html;
error_page 500 /error/500.html;
error_page 502 /error/502.html;
error_page 503 /error/503.html;
error_page 504 /error/504.html;

location ^~ /error/ {
    alias |DOCROOT|/error/;
    internal;
    # Repetir headers en este location block
    add_header X-Content-Type-Options "nosniff" always;
    add_header X-Frame-Options "SAMEORIGIN" always;
    add_header X-XSS-Protection "1; mode=block" always;
    add_header Referrer-Policy "strict-origin-when-cross-origin" always;
    add_header Permissions-Policy "geolocation=(), microphone=(), camera=()" always;
    add_header Strict-Transport-Security "max-age=31536000; includeSubDomains" always;
    add_header Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline'
'unsafe-eval' https:; style-src 'self' 'unsafe-inline' https:; img-src 'self' data: https:;
font-src 'self' data: https:; connect-src 'self' https:; frame-src 'self'
https://www.youtube.com https://www.google.com; frame-ancestors 'self'; base-uri 'self'; form-
action 'self';" always;
}
```

Aplicar cambios

Después de crear o modificar el archivo, reconstruir las configuraciones de nginx:

```
cd /usr/local/directadmin/custombuild
./build rewrite_confs
```

Personalización por Dominio

Para dominios que requieren configuraciones diferentes (CSP más permisivo, exclusiones de headers, etc.), DirectAdmin permite crear archivos de personalización por dominio.

Ubicación de archivos

```
/usr/local/directadmin/data/users/<USUARIO>/domains/<DOMINIO>.cust_nginx
/usr/local/directadmin/data/users/<USUARIO>/domains/<DOMINIO>.cust_nginx.2
/usr/local/directadmin/data/users/<USUARIO>/domains/<DOMINIO>.cust_nginx.3
/usr/local/directadmin/data/users/<USUARIO>/domains/<DOMINIO>.cust_nginx.4
```

Archivo de configuración personalizada en home del usuario

Para configuraciones más complejas, se puede crear un archivo `.conf` en el directorio del usuario e incluirlo:

```
/home/<USUARIO>/nginx/<nombre>.conf
```

Puntos de Inserción en DirectAdmin

DirectAdmin utiliza tokens en la plantilla `nginx_server.conf` para insertar configuraciones personalizadas:

Token	Archivo por Dominio	Ubicación en nginx
`CUSTOM1`	`.cust_nginx`	Antes del bloque `server {}`
`CUSTOM`	(interno)	Inicio del bloque `server {}`
`CUSTOM2`	`.cust_nginx.2`	Dentro de `location / {}`
`CUSTOM3`	`.cust_nginx.3`	Después de locations, antes de webapps
`CUSTOM4`	`.cust_nginx.4`	Final del bloque `server {}`

Archivo .cust_nginx (CUSTOM1)

Se usa principalmente para:

- Cambiar el DOCROOT (aplicaciones Laravel, etc.)
- Configuraciones que van antes del bloque server

Ejemplo - Cambiar DOCROOT para Laravel:

```
|?DOCROOT=`HOME`/domains/`DOMAIN`/mi-app/public|
```

Archivo .cust_nginx.3 (CUSTOM3)

Se usa para:

- Configuraciones de cache
- Location blocks adicionales
- Headers específicos para tipos de archivo

Ejemplo - Cache de assets:

```
location ~* \.(js|css|png|jpg|jpeg|gif|svg|ico)$ {
    expires 30d;
    add_header Cache-Control "public, no-transform";
}
location ~* \.(jpg|jpeg|gif|png|svg)$ {
    expires 365d;
}
location ~* \.(pdf|css|html|js|swf)$ {
    expires 2d;
}
```

Archivo .cust_nginx.4 (CUSTOM4)

Se usa para:

- Incluir archivos de configuración externos
- Location blocks complejos
- Reglas de rewrite específicas

Ejemplo - Incluir configuración WHMCS:

```
include /home/intranet/nginx/whmcs.conf;
```

Ejemplos Prácticos

Ejemplo 1: Dominio con CSP relajado

Para un dominio que necesita permitir scripts de terceros (ej: pasarelas de pago, widgets):

Archivo: `/usr/local/directadmin/data/users/usuario/domains/tienda.com.cust_nginx.4`

```
# Sobrescribir CSP global con uno más permisivo
add_header Content-Security-Policy "default-src 'self'; script-src 'self' 'unsafe-inline'
'unsafe-eval' https: blob:; style-src 'self' 'unsafe-inline' https:; img-src 'self' data:
https: blob:; font-src 'self' data: https:; connect-src 'self' https: wss:; frame-src 'self'
https:; frame-ancestors 'self'; base-uri 'self'; form-action 'self' https:;" always;
```

Ejemplo 2: Dominio que permite iframes externos

Para un dominio que necesita ser embebido en otros sitios:

Archivo: `/usr/local/directadmin/data/users/usuario/domains/widget.com.cust_nginx.4`

```
# Permitir que el sitio sea embebido
add_header X-Frame-Options "" always;
add_header Content-Security-Policy "default-src 'self'; frame-ancestors https://sitio-
```

```
padre.com https://otro-sitio.com;" always;
```

Ejemplo 3: Aplicación Laravel con DOCROOT personalizado

Archivo: `/usr/local/directadmin/data/users/develop/domains/larafactu.com.cust_nginx`

```
|?DOCROOT=`HOME`/domains/`DOMAIN`/larafactu/public|
```

Ejemplo 4: BookStack Wiki

Archivo: `/usr/local/directadmin/data/users/castris/domains/wiki.castris.com.cust_nginx`

```
|?DOCROOT=`HOME`/domains/`DOMAIN`/BookStack/public|
```

Ejemplo 5: Configuración compleja con archivo externo

Crear archivo de configuración en home del usuario:

Archivo: `/home/intranet/nginx/whmcs.conf`

```
# WHMCS CONFIG
location ~ /announcements/?(.*)$ {
    rewrite ^/(.*)$ /index.php?rp=/announcements/$1;
}

location ~ /download/?(.*)$ {
    rewrite ^/(.*)$ /index.php?rp=/download$1;
}

location ~ /knowledgebase/?(.*)$ {
    rewrite ^/(.*)$ /index.php?rp=/knowledgebase/$1;
}

# ... más reglas de rewrite ...
```

```
# Security Advisory
location ^~ /vendor/ {
    deny all;
    return 403;
}
```

Archivo: `/usr/local/directadmin/data/users/intranet/domains/intranet.castris.com.cust_nginx.4`

```
include /home/intranet/nginx/whmcs.conf;
```

Ejemplo 6: Deshabilitar todos los headers de seguridad (NO RECOMENDADO)

Solo para debugging o situaciones muy específicas:

Archivo: `/usr/local/directadmin/data/users/usuario/domains/debug.com.cust_nginx.4`

```
# TEMPORAL - Solo para debugging
add_header X-Content-Type-Options "" always;
add_header X-Frame-Options "" always;
add_header X-XSS-Protection "" always;
add_header Referrer-Policy "" always;
add_header Permissions-Policy "" always;
add_header Strict-Transport-Security "" always;
add_header Content-Security-Policy "" always;
```

Troubleshooting

Verificar sintaxis de nginx

```
nginx -t
```

Ver configuración generada para un dominio

```
cat /etc/nginx/conf.d/usuario.dominio.com.conf
```

Reconstruir todas las configuraciones

```
cd /usr/local/directadmin/custombuild
./build rewrite_confs
```

Recargar nginx sin reiniciar

```
systemctl reload nginx
```

Verificar headers de un sitio

```
curl -I https://dominio.com
```

Logs de errores

```
tail -f /var/log/nginx/domains/dominio.com.error.log
```

Notas importantes

1. **Orden de precedencia**: Los headers definidos en location blocks más específicos sobrescriben los globales.
2. **Keyword always**: Usar siempre `always` al final del `add_header` para que se aplique en todas las respuestas (incluyendo errores).
3. **Reconstruir después de cambios**: Siempre ejecutar `./build rewrite_confs` después de modificar archivos `.cust_nginx*`.
4. **Permisos**: Los archivos deben tener permisos correctos para que DirectAdmin pueda leerlos.
5. **Variables de DirectAdmin**: Se pueden usar variables como `|DOCROOT|`, `|DOMAIN|`, `|HOME|` en las configuraciones.

Variables Disponibles en Templates

Variable	Descripción
`|DOMAIN|`	Nombre del dominio
`|HOME|`	Directorio home del usuario
`|DOCROOT|`	Document root del dominio
`|IP|`	IP del servidor
`|PORT_80|`	Puerto HTTP
`|PORT_443|`	Puerto HTTPS

Referencias

- DirectAdmin Custom Nginx Templates
- Mozilla Observatory - Para verificar headers de seguridad
- Security Headers - Análisis de headers

Aviso

Esta documentación se entrega tal y como está, basada en configuración del servidor dar.tabratino.com.

Revision #6
Created 2025-12-25 07:05:37 UTC by Abkrim
Updated 2026-01-18 05:50:58 UTC by Abkrim