TECNOLÓGICO NACIONAL DE MÉXICO

Instituto Tecnológico de Mexicali

Ing. Sistemas Computacionales

Desarrollo de Aplicaciones Web

Docente: Bogarin Valenzuela Jose Ramon

“Investigación U3 – Archivo Nginx.conf”

Alumna: Rodriguez Herrera Maria Fernanda

Mexicali, B.C. **31-Oct-2025**

Archivo ninx.conf

nginx.conf es el archivo de configuración por defecto del servidor Nginx, el cual contiene la gestión de conexiones, la configuración de los host virtuales, la seguridad, así como el manejo de archivos estáticos. La configuración básica para este archivo es:

```
server {
    listen 80;
    server_name localhost;

    # Directorio raíz donde están los archivos estáticos
    root /usr/share/nginx/html;
    index index.html;

    # Configuración de logs
    access_log /var/log/nginx/access.log;
    error_log /var/log/nginx/error.log;

    # Compresión gzip para mejorar el rendimiento
    gzip on;
    gzip_vary on;
    gzip_min_length 1024;
    gzip_types text/plain text/css text/xml text/javascript application/x-
javascript application/xml+rss application/javascript application/json;
}
```

Donde se inicia el bloque del ***server***, que contiene el puerto de escucha ***80***, seguido del nombre del servidor que es ***localhost***. Luego tenemos la ruta para los archivos estáticos donde nginx los buscará, que es ***root /usr/share/ninx/html***, seguido del archivo por defecto que es ***index index.html***. Después se tiene la configuración de logs, que es donde se guarda la información de las solicitudes HTTP en el archivo ***access.log***, así como el registro de errores (fallos en la configuración, rutas no encontradas, etc.) que es el archivo ***error.log***. Y por último la compresión ***gzip*** para la mejora del rendimiento, que es más que nada (como su nombre lo dice) comprimir las respuestas HTTP para reducir el tamaño y así mejorar la respuesta de carga del navegador, archivos que tengan una longitud mínima de 1024 b, así como archivos html, css, xml, json, etc.

Pero, ¿qué pasa si nuestro archivo no está protegido o no contiene reglas de restricción? Esto puede llevar a poner en riesgo la seguridad y privacidad de nuestra aplicación, por lo que, se podría acceder a archivos sensibles, exponer código fuente/html, realizar ataques por mapeo de rutas, etc., por ello, es importante especificar en este archivo las rutas que queremos proteger.

Por ejemplo, la sintaxis general para bloquear cualquier tipo de archivos es:

```
location ~ ruta_del_archivo_a_bloquear {
    deny all;
    return 404;
}
```

Aquí se especifica por bloques, con la localización del archivo con ***location ~*** seguido de la ruta donde se encuentra, y dentro de este con ***deny all*** se bloquea el acceso para todos los usuarios y devolverá un error ***404*.** Por ejemplo, se puede bloquear el acceso a archivos de entorno ***(/\.env)***, archivos git ***(/\.git)***, archivos de sistema y configuración ***(/\.)***, archivos .map ***(\.map$)***, entre otros.

Referencias:

Breus, V. (2023, May 18). *Nginx Seguridad: Cómo controlar los recursos y establecer límites para un servidor más seguro*. Serverspace.io; ITGLOBAL.COM NL. https://serverspace.io/es/support/help/nginx-web-server-security-resources-limits/

Como Bloquear el Acceso a un Archivo de un Servidor Nginx (Casos de Uso). (2019, November 17). Nube Colectiva | La Mejor Comunidad de Programación. https://nubecolectiva.com/blog/como-bloquear-el-acceso-a-un-archivo-de-un-servidor-nginx-casos-de-uso/