What are nginx header security and its uses. And also implement in the test.conf file.

X-Frame-Options

The X-Frame-Options HTTP response header can be used to indicate whether or not a browser should be allowed to render a page in a <frame>, <iframe>, <embed> or <object>.

This header prevents clickjacking attacks by ensuring that the malicious content is not being embedded into the website.

X-XSS-Protection

The X-XSS-Protection header is used to filter out cross-site scripting (XSS) in modern browsers.

This is usually enabled by default, but using it will enforce it. It is supported by Internet Explorer 8+, Chrome, and Safari.

X-Content-Type-Options

The X-Content-Type-Options header prevents Internet Explorer and Google Chrome from sniffing a response away from the declared Content-Type. This helps reduce the danger of drive-by downloads and helps treat the content the right way.

Referrer Policy

The Referrer-Policy HTTP header controls how much referrer information (sent via the Referer header) should be included with requests.

Aside from the HTTP header, you can set this policy in HTML.

no-referrer-when-downgrade

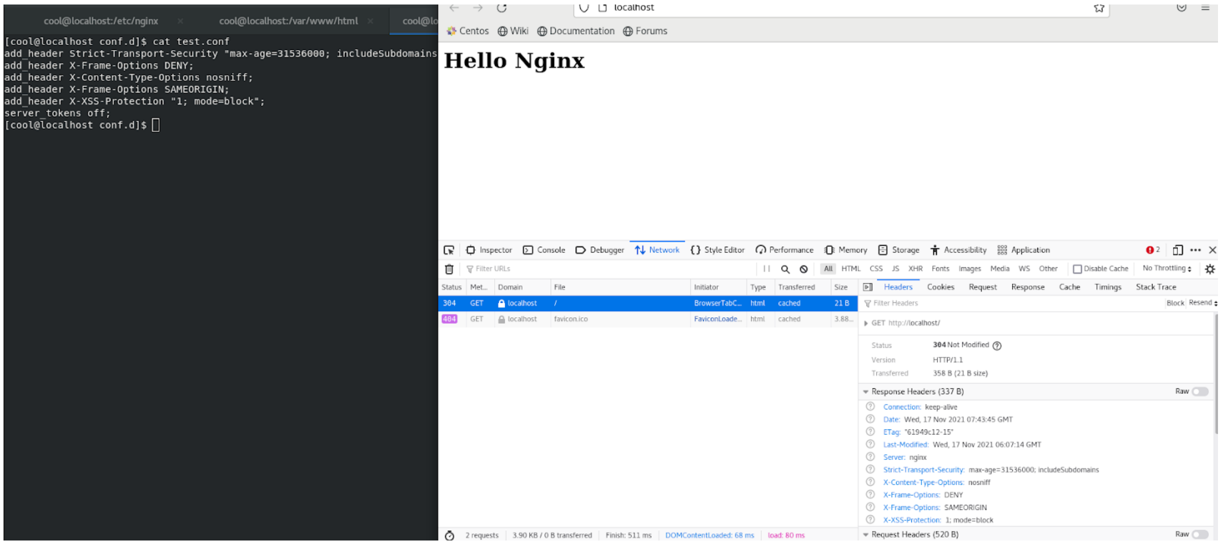
Send the origin, path, and query string in Referer when the protocol security level stays the same or improves (HTTP→HTTP, HTTP→HTTPS, HTTPS→HTTPS). Don't send the Referer header for requests to less secure destinations (HTTPS→HTTP, HTTPS→file).

Content Security Policy

The Content-Security-Policy is an HTTP security header that provides an additional layer of security.

This policy allows the browser to only loads the approved resources. Doing so helps in preventing the attacks like Cross-Site Scripting (XSS) and other code injection attacks

```
# security headers
add_header X-Frame-Options           "SAMEORIGIN" always;
add_header X-XSS-Protection          "1; mode=block" always;
add_header X-Content-Type-Options    "nosniff" always;
add_header Referrer-Policy           "no-referrer-when-downgrade" always;
add_header Content-Security-Policy   "default-src 'self' http: https:
data: blob: 'unsafe-inline'" always;
add_header Strict-Transport-Security "max-age=31536000; includeSubDomains"
always;
```



```
[cool@localhost conf.d]$ cat test.conf
add_header Strict-Transport-Security "max-age=31536000; includeSubdomains
add_header X-Frame-Options DENY;
add_header X-Content-Type-Options nosniff;
add_header X-Frame-Options SAMEORIGIN;
add_header X-XSS-Protection "1; mode=block";
server_tokens off;
[cool@localhost conf.d]$ 
```

Hello Nginx

Status	Met...	Domain	File	Initiator	Type	Transferred	Size
304	GET	localhost	/	BrowserTabC...	html	cached	21 B
404	GET	localhost	favicon.ico	FaviconLoade...	html	cached	3.88...

```
GET http://localhost/

Status       304 Not Modified
Version      HTTP/1.1
Transferred  358 B (21 B size)

Response Headers (337 B)
Connection: keep-alive
Date: Wed, 17 Nov 2021 07:43:45 GMT
ETag: "61949c12-15"
Last-Modified: Wed, 17 Nov 2021 06:07:14 GMT
Server: nginx
Strict-Transport-Security: max-age=31536000; includeSubdomains
X-Content-Type-Options: nosniff
X-Frame-Options: DENY
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
Request Headers (520 B)
```

2 requests 3.90 KB / 0 B transferred Finish: 511 ms DOMContentLoaded: 68 ms load: 80 ms

I added the following test.conf:

add_header Strict-Transport-Security "max-age=31536000; includeSubdomains";
add_header X-Frame-Options DENY;
add_header X-Content-Type-Options nosniff;
add_header X-Frame-Options SAMEORIGIN;
add_header X-XSS-Protection "1; mode=block";
server_tokens off; #hide the nginx server version