Nginx Installation using dnf:

Sudo dnf install nginx

```
Installed:
  nginx-1:1.14.1-9.module_el8.0.0+184+e34fea82.x86_64                nginx-all-modules-1:1.14.1-9.module_el8.0.0+184+e34fea82.noarch
  nginx-filesystem-1:1.14.1-9.module_el8.0.0+184+e34fea82.noarch     nginx-mod-http-image-filter-1:1.14.1-9.module_el8.0.0+184+e34fea82.x86_64
  nginx-mod-http-perl-1:1.14.1-9.module_el8.0.0+184+e34fea82.x86_64  nginx-mod-http-xslt-filter-1:1.14.1-9.module_el8.0.0+184+e34fea82.x86_64
  nginx-mod-mail-1:1.14.1-9.module_el8.0.0+184+e34fea82.x86_64       nginx-mod-stream-1:1.14.1-9.module_el8.0.0+184+e34fea82.x86_64
```

Enabling Nginx services using :

sudo systemctl enable nginx
sudo systemctl start nginx

```
[cool@localhost ~]$ sudo systemctl enable nginx
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.
[cool@localhost ~]$ sudo systemctl start nginx
```

Adjusting Firewall Rules to allow to listen on port 80:

sudo firewall-cmd --permanent --add-service=http

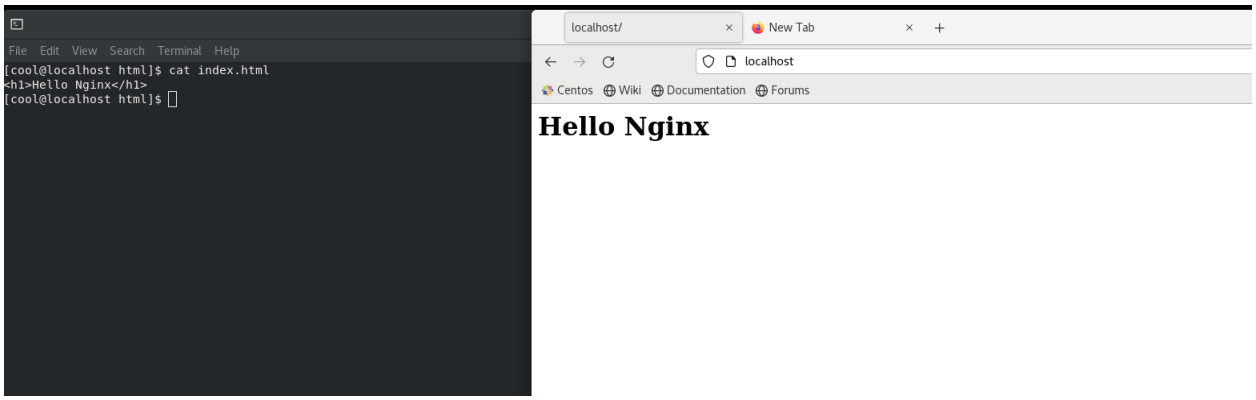
```
[cool@localhost ~]$ sudo firewall-cmd --permanent --add-service=http
success
[cool@localhost ~]$ sudo firewall-cmd --permanent --list-all
internal
  target: default
  icmp-block-inversion: no
  interfaces:
  sources:
  services: cockpit dhcpv6-client http mdns openvpn samba-client ssh
  ports: 1194/tcp 1194/udp
  protocols:
  masquerade: yes
  forward-ports:
  source-ports:
  icmp-blocks:
  rich rules:
```

sudo firewall-cmd --reload

For hosting a index.html saying Hello Nginx:

Modified the index.html file in /usr/share/nginx/html and editing the content to:

<h1>Hello Nginx</h1>



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

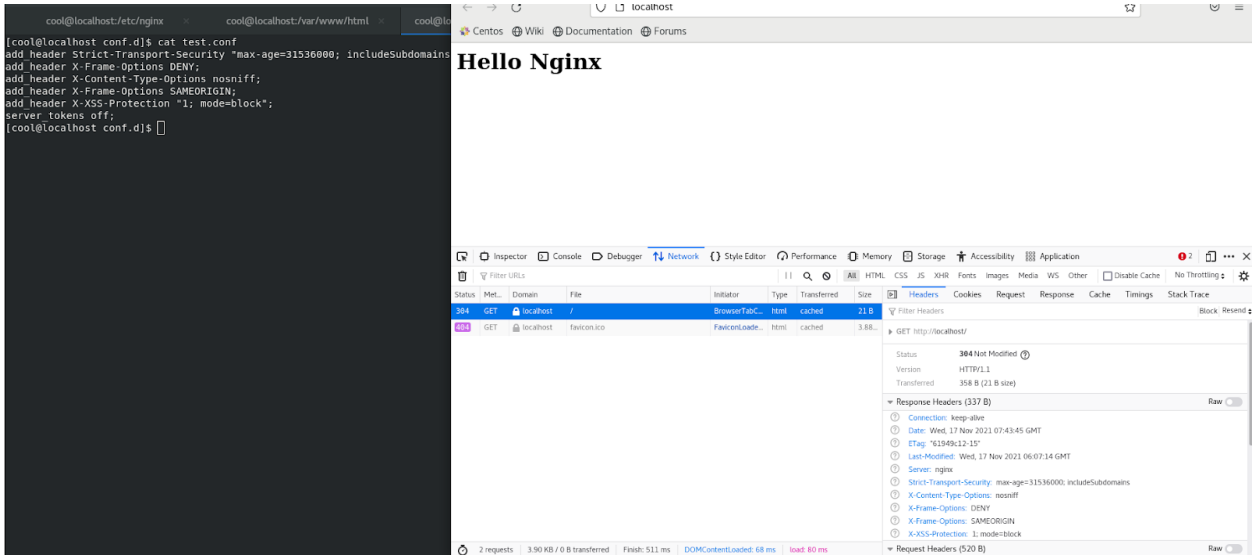
Send the origin, path, and query string in Referer when the protocol security level stays the same or improves (HTTP→HTTP, HTTP→HTTPS, HTTPS→HTTPS). Don't send the Referer header for requests to less secure destinations (HTTPS→HTTP, HTTPS→file).

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;
```

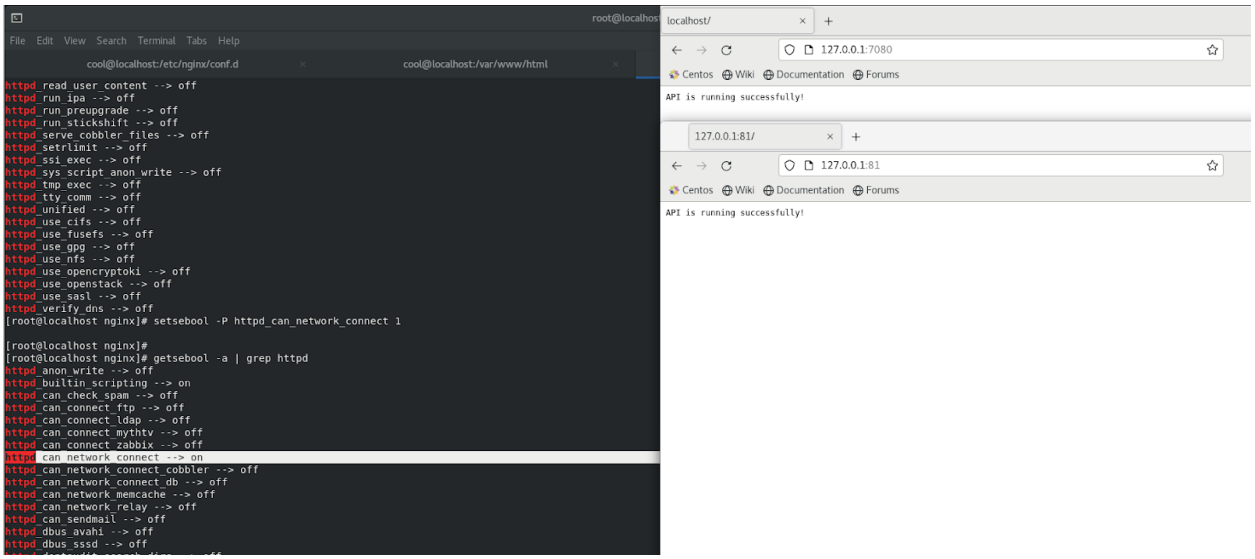


I added the following test.conf:

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

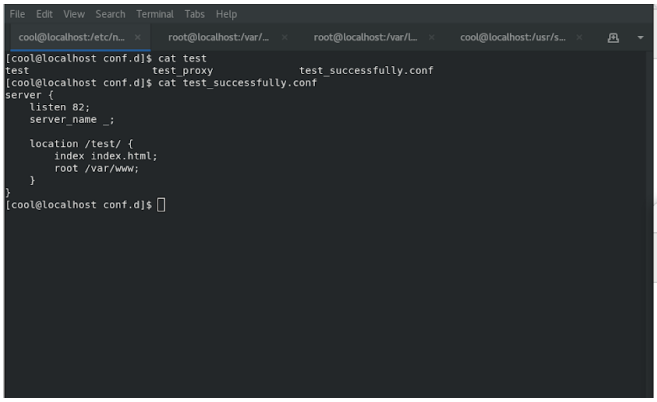
Nginx Reverse proxy all http requests to nodes js api.

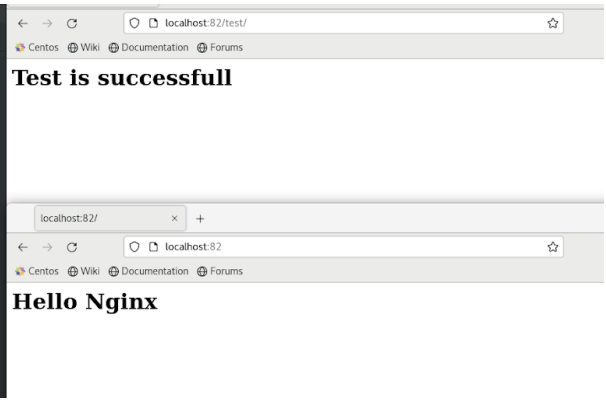
```
server {
    listen 81;
    server_name _;

    location / {
        proxy_pass         http://127.0.0.1:7080;
    }
}
```



Create a test2.conf and listen on port 82 and to “ location /test/” with the message “ test is successful”.

```
server {
    listen 82;
    server_name _;

    location /test/ {
            index index.html;
            root /var/www;
    }
}
```





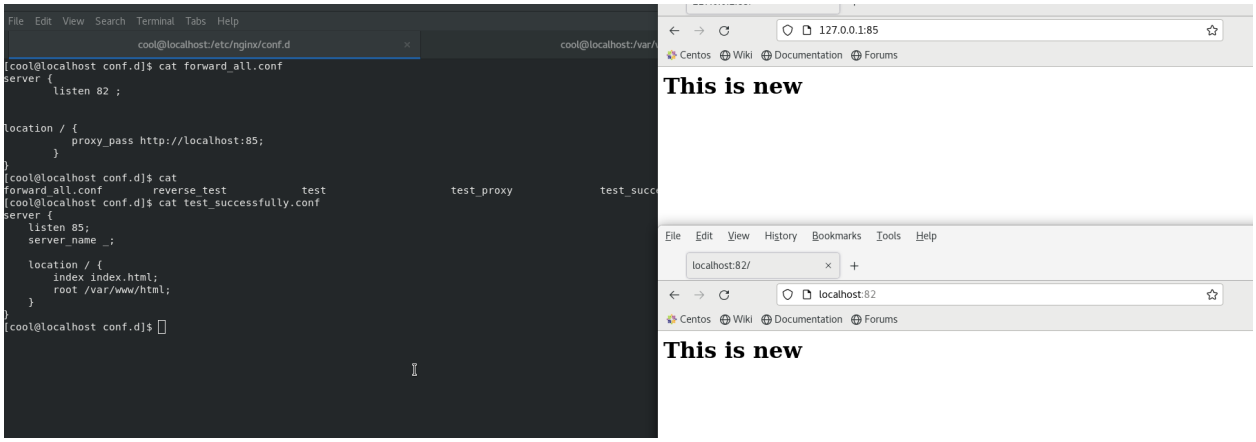
Reverse proxy all http traffic of port 82 to port 85.

sudo semanage port -a -t http_port_t -p tcp 85

- To add the 85 port in http_port

chcon -vR system_u:object_r:httpd_sys_content_t:s0 /var/www/Folder_name/

- To allow custom document root to be served as HTTP content.



Install LEMP stack (avoid installing mysql) and open info.php on port 80 and print message info.php.

LEMP Stands For:

- L- Linux Operating System (Using ubuntu)
- E- Nginx Server (nginx is already installed)
- M- MySQL Database (not required as per the requirement)
- P- PHP (installing php)

First added the repository and installed the php and php fpm:

sudo add-apt-repository ppa:ondrej/php

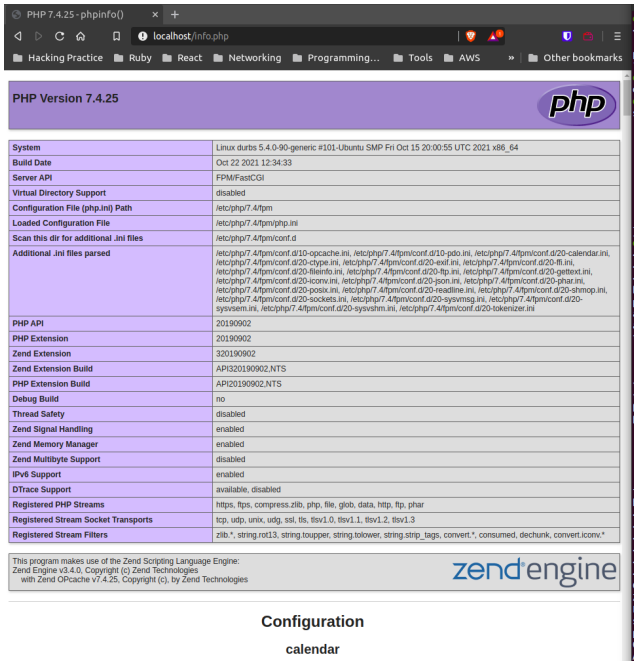
Sudo apt install php php-fpm

Reconfigured the default configure file in /etc/nginx/site-enabled:

```
server {
        listen 80;
        listen [::]:80;
        root /var/www/html;
        index info.php;
        server_name localhost;
    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php7.4-fpm.sock;
    }
}
```

After that created the symbolic link between: site-available and site enabled.

```
Sudo ln -s /etc/nginx/site-available/default /etc/nginx/site-enabled/
```




```
durbs@durbs:~$ cat /var/www/html/info.php
<?php

phpinfo();
?>
durbs@durbs:~$ cat /etc/nginx/sites-available/d
default         django_deploy_lf
durbs@durbs:~$ cat /etc/nginx/sites-available/default
server {
        listen 80;
        listen [::]:80;
        root /var/www/html;
        index info.php;
        server_name localhost;
    location ~ \.php$ {
        include snippets/fastcgi-php.conf;
        fastcgi_pass unix:/run/php/php7.4-fpm.sock;
    }
}
durbs@durbs:~$ curl http://localhost/info.php
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"><head>
<style type="text/css">
body {background-color: #fff; color: #222; font-family: sans-serif;}
pre {margin: 0; font-family: monospace;}
a:link {color: #009; text-decoration: none; background-color: #fff;}
a:hover {text-decoration: underline;}
table {border-collapse: collapse; border: 0; width: 934px; box-shadow: 1px 2px 3px #ccc;}
.center {text-align: center;}
.center table {margin: 1em auto; text-align: left;}
.center th {text-align: center !important;}
td, th {border: 1px solid #666; font-size: 75%; vertical-align: baseline; padding: 4px 5px;}
th {position: sticky; top: 0; background: inherit;}
h1 {font-size: 150%;}
h2 {font-size: 125%;}
.p {text-align: left;}
.e {background-color: #ccf; width: 300px; font-weight: bold;}
.h {background-color: #99c; font-weight: bold;}
.v {background-color: #ddd; max-width: 300px; overflow-x: auto; word-wrap: break-word;}
.v i {color: #999;}
img {float: right; border: 0;}
hr {width: 934px; background-color: #ccc; border: 0; height: 1px;}
</style>
<title>PHP 7.4.25 - phpinfo()</title><meta name="ROBOTS" content="NOINDEX,NOFOLLOW,NOARCHIVE" /></head>
<body><div class="center">
<table>
<tr class="h"><td>
<a href="http://www.php.net/"><img border="0" src="
CAYAAAA+j9gsAAAAGXRFWHRTb2Z0d2FyZQBBZG9lZSBJbWFnZVJlYWR5ccllPAAAD4BJREFUeNrsnXtwXFUdx8/
ZyKon/gYSqizjgM2KQMfzFAOloOA5KEh+j4R9oZH7zT6MAMKrNphZFSQreKHRgZmspLHSCJ2Co6tBtJk7Zps7tJsSt95FS/33PvWU4293
F29ybdlPzaM3df2XPv+ZzF4/zOuWc1tkjl+T0WQ35QC65BSlD6WKW4rusCm9F1ps/o5mPrlOf8dd0YoNfl0nt4ntB1PT4zYwzQkf3k89/
sW4xtpS0CmE0SyPUFUJXFMIxZcM0jAZ4xrKMudQT796HBF6n6EaUjkP0v19K9OEHWqJLkNW1s8mC2WgVTwGAqWTafJzTWTKZmQuZ/k1M
pAi2+eys6mpWfVaAPzcILu8EVKoCAaYFtPxrAXo8qyNwzZc7gSgzgN9Hx0Ecn3j8xr4lyHOhNrlpaJIgptM5DjCdzrJ0Jmce6bWFkOpqs
0MErA4gXIBuAmYS3gFmOPCcdaTXCbq+n16PPLXjewMfGcgEttECeouTpk5MplhyKsPBTlXNYyULtwIW7Cx1vlwuJyDLR9L0mQlVPb27fh
A54yBbGttMpc1OWwF1cmKaH2FSF7vAjGezOZZJZ9j0dIZlMhnuRiToMO0c+N4X7oksasqEt9XS2KZCHZoem2Ixq5zpAuDTqTR14FMslZv
```