

Foundations of Computer Security: Milestone 1 Report

September 6, 2023

Aruba Sood

MT23022

Kajol Dwivedi

MT23041

Shantanu Mallik

MT23140

Vanshaj Sharma

MT23103

- Logging into VM using ssh

```
ssh iiitd@192.168.2.224
```

- Installing NGINX

```
sudo apt install nginx
```

- Copying react build files to nginx server root

while at the root of the react project folder, run

```
scp -r ./build/* iiitd@192.168.2.244:/var/www/html
```

This copies all the build files to the root of our nginx server

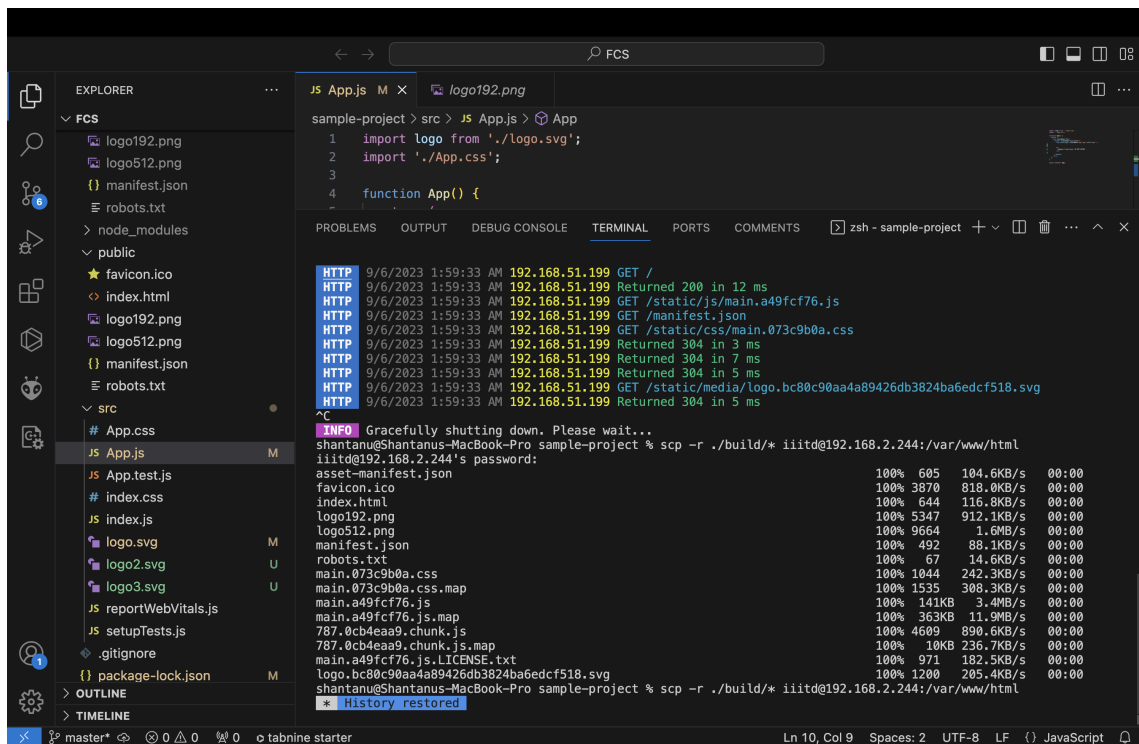


Figure 1: Copy Files to Server root

- Creating self signed ssl certificates

- creating an openssl certificate

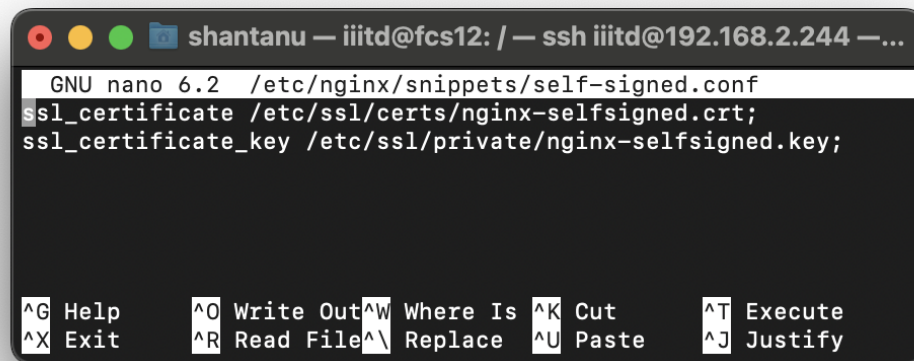
```
sudo openssl req -x509 -nodes -days 365 -newkey rsa:2048 -keyout /etc/ssl/private/nginx-selfsigned.key
-out /etc/ssl/certs/nginx-selfsigned.crt
```

[illegible]

– **Creating an nginx configuration snippet to point to the ssl certificate**

```
sudo nano /etc/nginx/snippets/self-signed.conf
```

This contains the paths to the ssl certificate and the ssl certificate key



```
GNU nano 6.2 /etc/nginx/snippets/self-signed.conf
ssl_certificate /etc/ssl/certs/nginx-selfsigned.crt;
ssl_certificate_key /etc/ssl/private/nginx-selfsigned.key;

^G Help    ^O Write Out ^W Where Is ^K Cut      ^T Execute
^X Exit    ^R Read File ^\ Replace  ^U Paste    ^J Justify
```

Figure 3: NGINX configuration snippet

– Adding ssl to our nginx configuration

sudo nano /etc/nginx/sites-available/default



```
GNU nano 6.2 /etc/nginx/sites-available/default
##
# You should look at the following URL's in order to grasp a solid understanding
# of Nginx configuration files in order to fully unleash the power of Nginx.
# https://www.nginx.com/resources/wiki/start/
# https://www.nginx.com/resources/wiki/start/topics/tutorials/config_pitfalls/
# https://wiki.debian.org/Nginx/DirectoryStructure
#
# In most cases, administrators will remove this file from sites-enabled/ and
# leave it as reference inside of sites-available where it will continue to be
# updated by the nginx packaging team.
#
# This file will automatically load configuration files provided by other
# applications, such as Drupal or Wordpress. These applications will be made
# available underneath a path with that package name, such as /drupal8.
#
# Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
##

# Default server configuration
#
server {
    listen 80 default_server;
    listen [::]:80 default_server;
    listen 443 ssl http2 default_server;
    listen [::]:443 ssl http2 default_server;

    server_name 192.168.2.244;
    include snippets/self-signed.conf;
    include snippets/ssl-params.conf;
    # SSL configuration
    #
    #listen 443 ssl default_server;
    #listen [::]:443 ssl default_server;
    #
    # Note: You should disable gzip for SSL traffic.
    # See: https://bugs.debian.org/773332
    #
    # Read up on ssl_ciphers to ensure a secure configuration.
    # See: https://bugs.debian.org/765782
    #
    # Self signed certs generated by the ssl-cert package
    # Don't use them in a production server!
    #
    # include snippets/snakeoil.conf;

    root /var/www/html;

    # Add index.php to the list if you are using PHP
    index index.html index.htm index.nginx-debian.html;

    server_name _;

    location / {
[ Read 96 lines ]
^G Help      ^O Write Out ^W Where Is  ^K Cut       ^T Execute   ^C Location
^X Exit      ^R Read File ^_ Replace   ^U Paste     ^J Justify   ^_ Go To Line
```

Figure 4: NGINX SSL configuration

– **Setting the changes in nginx**

```
sudo nginx -t
```

```
sudo systemctl restart nginx
```

This part has been referred from:

(<https://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-nginx-in-ubuntu-16-04>)

• **Final deployment**

To access the website, visit `https://192.168.2.244/` while being connected to the IIITD network.

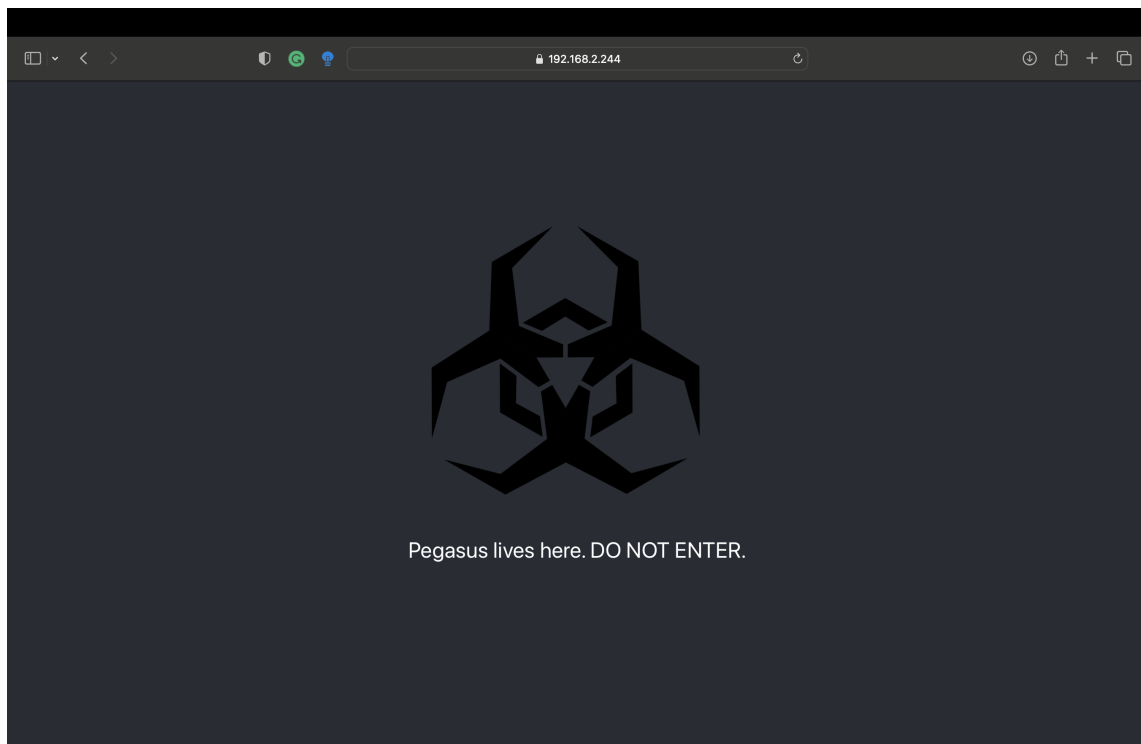


Figure 5: Pegasus Home