SECURE NGINX FROM CLICKJACKING ATTACK

CLICKJACKING:

- It is a type of web attack where an attacker tricks a user into clicking on a hidden or disguised element on a webpage without their knowledge.
- This is achieved by placing the target website inside a transparent or invisible iframe on the attacker's website.
- An inline frame (iframe) is a HTML element that loads another HTML page within the document. It essentially puts another webpage within the parent page,commonly used for advertisements, embedded videos.
- As a result, when the user interacts with the attacker's page, they are actually performing actions on the target website without realizing it.

PREVENT CLICKJACKING:

- To prevent clickjacking in Nginx, you can use the X-Frame-Options header, which instructs the browser on how to display your website within iframes.

    1. X-Frame-Options: DENY - This option denies any framing of your website, preventing it from being loaded in iframes altogether.

    2. X-Frame-Options: SAMEORIGIN - This option allows your website to be loaded in iframes that originate from the same domain. It prevents your website from being embedded in iframes on other domains.

DEMO TO TEST CLICKJACKING ATTACK AND THEN PREVENT WEBSITE FROM CLICKJACKING :

STEP 1:

- Install nginx
    1. sudo apt update
    2. sudo apt install nginx -y

STEP 2:

- Move into /var/www and create a new directory with an html file inside it .

```
ubuntu@ip-172-31-18-120:~$ cd /var/www
ubuntu@ip-172-31-18-120:/var/www$ ls
html
ubuntu@ip-172-31-18-120:/var/www$ sudo mkdir victim
ubuntu@ip-172-31-18-120:/var/www$ cd victim
ubuntu@ip-172-31-18-120:/var/www/victim$ sudo nano index.html
ubuntu@ip-172-31-18-120:/var/www/victim$
```

- Add the following content inside the index html file.

```
<!DOCTYPE html>
<html>
<head>
    <title>Victim Website</title>
</head>
<body>
    <h1>Victim Website</h1>
    <p>This is the victim website that we want to protect from Clickjacking.</p>
</body>
</html>
```

STEP 3:

- Move inside sites-available and create a configuration file.

```
ubuntu@ip-172-31-18-120:/var/www/victim$ cd /etc/nginx/sites-available/
ubuntu@ip-172-31-18-120:/etc/nginx/sites-available$ ls
default
ubuntu@ip-172-31-18-120:/etc/nginx/sites-available$ sudo nano victim.conf
```

- Add the following code inside the configuration file.

```
server {
    listen 80;
    server_name 18.212.246.96;

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

- Make sure to change the server name with your own server name and also the location path.

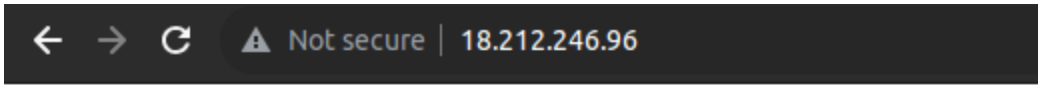
- Delete the default config file in sites-enabled and link the newly created config file into sites-enabled.

```
ubuntu@ip-172-31-18-120:~$ sudo ln -s /etc/nginx/sites-available/victim.conf /etc/nginx/sites-enabled/
ubuntu@ip-172-31-18-120:~$ cd /etc/nginx/sites-enabled/
ubuntu@ip-172-31-18-120:/etc/nginx/sites-enabled$ ls
default  victim.conf
ubuntu@ip-172-31-18-120:/etc/nginx/sites-enabled$ sudo rm default
ubuntu@ip-172-31-18-120:/etc/nginx/sites-enabled$ ls
victim.conf
```

- Test for syntax error in the config file and restart Nginx to save the configuration for the changes to take effect.

  sudo nginx -t
  sudo service nginx restart

- Paste the ip-address of the instance in the search bar,we should get the website being displayed.



Victim Website

This is the victim website that we want to protect from Clickjacking.

STEP 4:

- Install nginx on the attacker server as well and create an html file inside /var/www.

  1. sudo apt update
  2. sudo apt install nginx -y

STEP 5:

- Move into /var/www and create a new directory with an html file inside it .

```
ubuntu@ip-172-31-18-120:~$ cd /var/www
ubuntu@ip-172-31-18-120:/var/www$ ls
html
ubuntu@ip-172-31-18-120:/var/www$ sudo mkdir victim
ubuntu@ip-172-31-18-120:/var/www$ cd victim
ubuntu@ip-172-31-18-120:/var/www/victim$ sudo nano index.html
ubuntu@ip-172-31-18-120:/var/www/victim$
```

- Add the following code inside the index file.

```
 GNU nano 6.2                                                    index.html
<!DOCTYPE html>
<html>
<head>
    <title>Attacker Website</title>
</head>
<body>
    <h1>Attacker Website</h1>
    <p>This is the attacker's website that will attempt to Clickjack the victim website.</p>
    <iframe src="http://18.212.246.96/" width="800" height="600"></iframe>
</body>
</html>
```

- Make sure to change the ip address to the victim server ip to implement clickjacking.

STEP 6:

- Move inside sites-available and create a configuration file.

```
ubuntu@ip-172-31-21-200:/var/www/attacker$ cd /etc/nginx/sites-available/
ubuntu@ip-172-31-21-200:/etc/nginx/sites-available$ ls
default
ubuntu@ip-172-31-21-200:/etc/nginx/sites-available$ sudo nano attacker.conf
ubuntu@ip-172-31-21-200:/etc/nginx/sites-available$
```

- Add the following code inside the configuration file.

```
  GNU nano 6.2
server {
    listen 80;
    server_name 52.90.128.68;

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

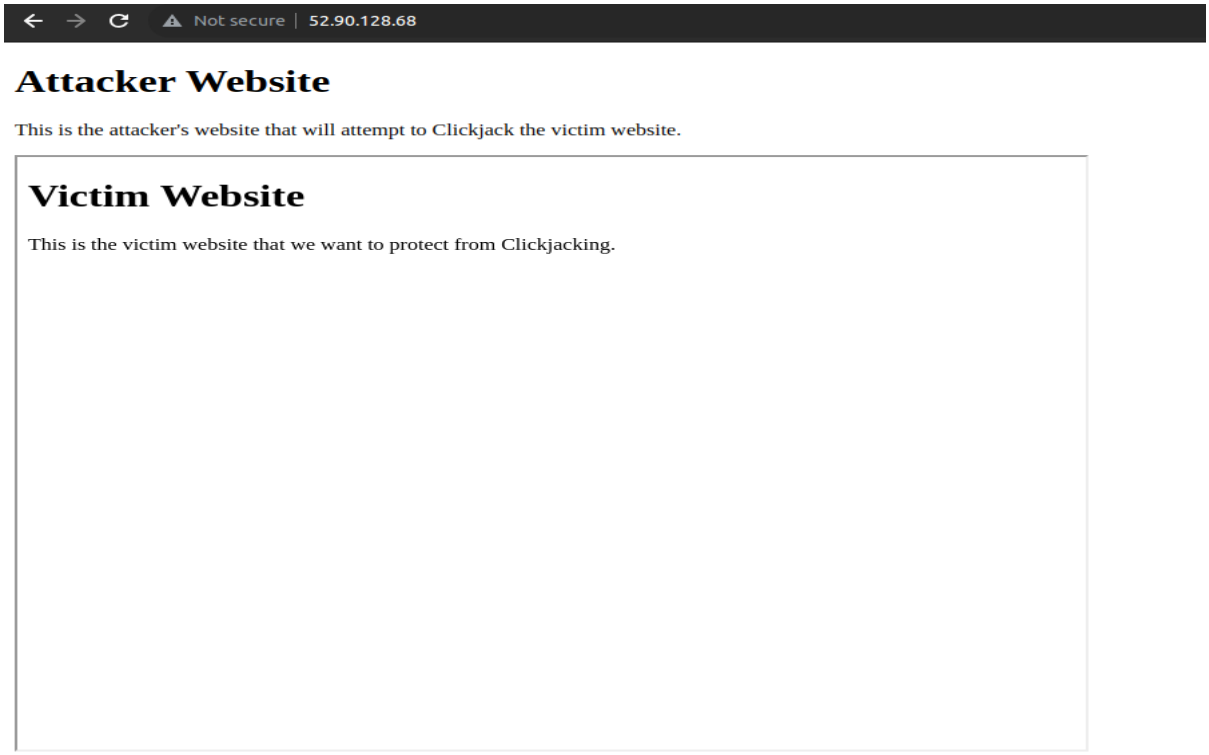
- Make sure to change the server name with your own server name and also the location path.

- Delete the default config file in sites-enabled and link the newly created config file into sites-enabled.

```
ubuntu@ip-172-31-18-120:~$ sudo ln -s /etc/nginx/sites-available/victim.conf /etc/nginx/sites-enabled/
ubuntu@ip-172-31-18-120:~$ cd /etc/nginx/sites-enabled/
ubuntu@ip-172-31-18-120:/etc/nginx/sites-enabled$ ls
default  victim.conf
ubuntu@ip-172-31-18-120:/etc/nginx/sites-enabled$ sudo rm default
ubuntu@ip-172-31-18-120:/etc/nginx/sites-enabled$ ls
victim.conf
```

- Test for syntax error in the config file and restart Nginx to save the configuration for the changes to take effect.
    1. sudo nginx -t
    2. sudo service nginx restart

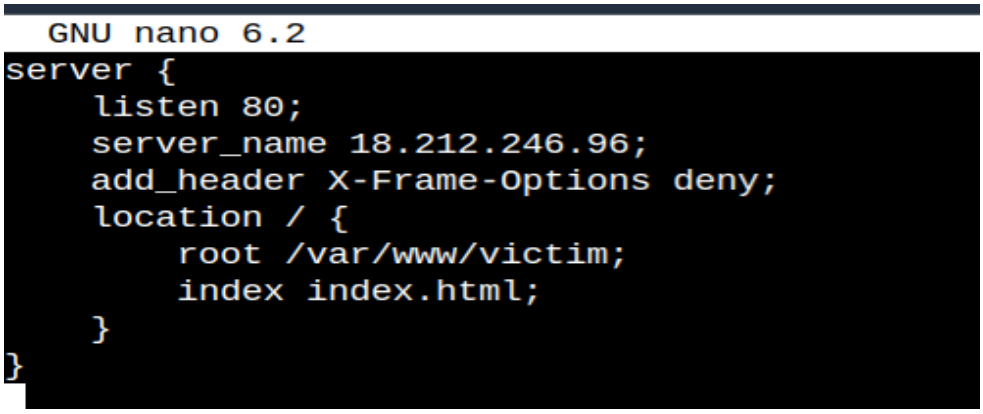
- Paste the ip-address of the instance in the search bar,we should get the website being displayed.



- We can see that the victim webpage is being displayed inside of the attacker webpage, this is called clickjacking, so in order to prevent this, we can use X-Frame options inside the victim server html file.
- This will prevent other websites from displaying content of other websites.

STEP 7:

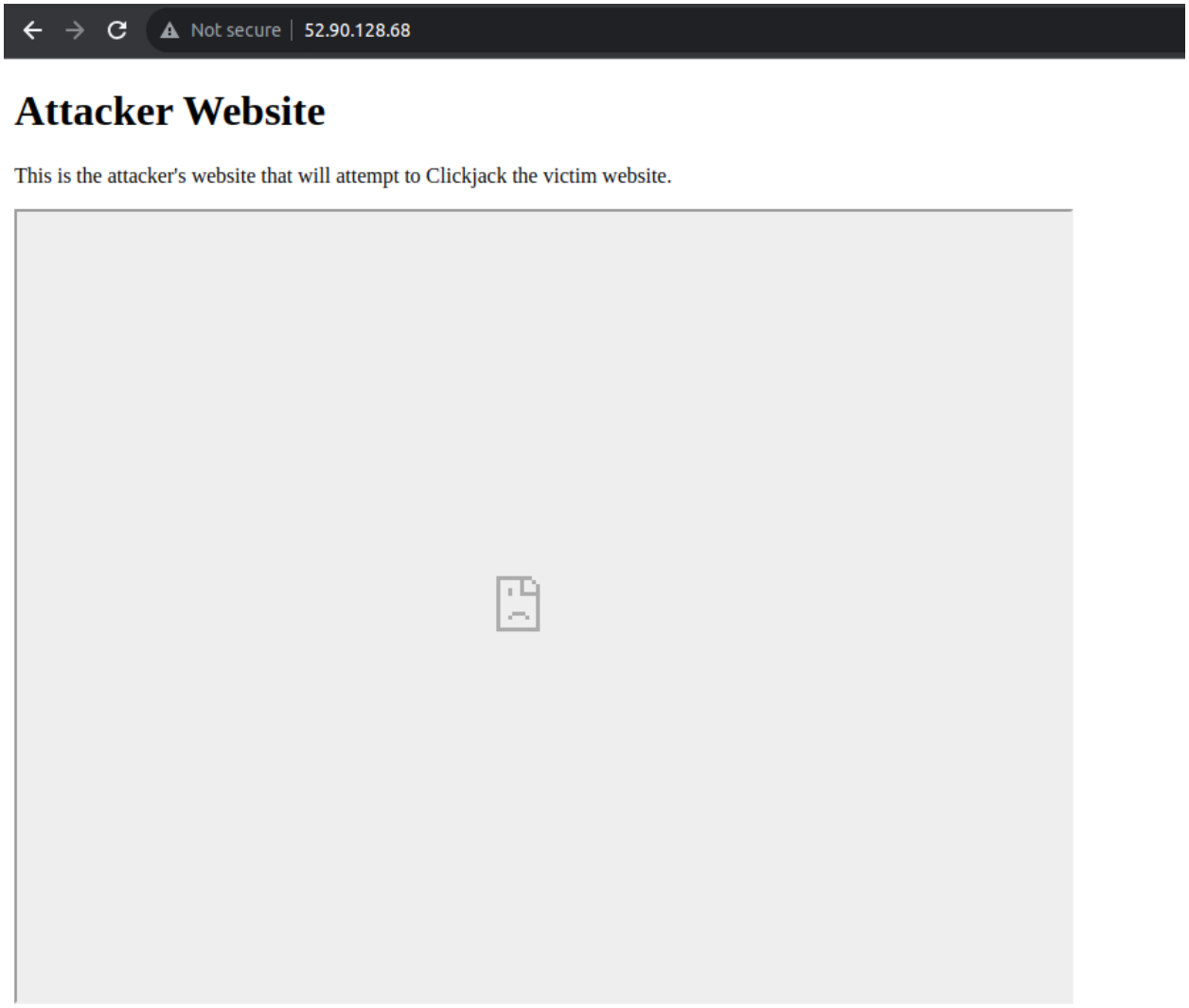
- Add the following line inside the config file of the victim server.
  add_header X-Frame-Options deny;



- After adding the line,save the changes and check for syntax error and restart nginx.
  1. sudo nginx -t
  2. sudo service nginx reload

STEP 8:

- Paste the ip of the attacker server again and check if we can see the victim server being displayed.



- We can see that the attacker server cannot access the victim server,since the victim server is configured with X-Frame Options.
- This shows that we can prevent our websites from clickjacking.