Team Avengers

Agenda

This is the second set in the series of Avengers training. This will be given with 1 use case. This consists of 50 Points.

1. Work with target machine

Use Case

In the target machine we need to install Nginx for serving web content.

Task

Write a shell script for the following.

1. You can provision a VM locally or AWS.
2. Connect to the provisioned machine.
3. Install nginx (you need to understand about nginx).
4. Copy the following content into the directory /etc/nginx/sites-enabled/ as file `default`.

```
map $sent_http_content_type $charset {
   ~^text/ UTF-8;
   ~^image/ UTF-8;
   ~^application/ UTF-8;
}

map $http_upgrade $connection_upgrade {
      default upgrade;
      '' close;
}

limit_req_zone $binary_remote_addr zone=mylimit:10m rate=3000r/s;

server {
   server_name 127.0.0.1;
   listen 8443 ssl;
   ssl_protocols TLSv1.2 TLSv1.3;
   ssl_certificate /opt/cisco/merck/certs/merck-web.crt;
   ssl_certificate_key /opt/cisco/merck/certs/merck-web.key;

   ssl_stapling on;
   ssl_stapling_verify on;
   ssl_trusted_certificate /opt/cisco/merck/certs/merck-web.crt;
```

```
    proxy_hide_header X-Content-Type-Options;

    add_header X-Content-Type-Options nosniff;

    set $CSP "default-src 'self'";
    set $CSP "${CSP}; frame-ancestors";
    set $CSP "${CSP}; block-all-mixed-content";
    set $CSP "${CSP}; base-uri 'self'";
    set $CSP "${CSP}; worker-src blob:";
    set $CSP "${CSP}; img-src 'self' data:";
    set $CSP "${CSP}; connect-src 'self' wss:";
    set $CSP "${CSP}; style-src 'self' 'unsafe-inline'";
    set $CSP "${CSP}; script-src 'self' 'unsafe-inline' 'unsafe-eval'";
    add_header Content-Security-Policy $CSP;

    charset $charset;
    charset_types *;
    client_max_body_size 100m;

    location /merck {
        # Added on 06/12/2020
        add_header Cache-Control 'private, no-store, max-age=0';
        add_header X-Frame-Options sameorigin always;
        add_header Strict-Transport-Security "max-age=31536000; includeSubDomains"
always;
        limit_req zone=mylimit burst=3000 nodelay;

        root /opt/cisco;
        index index.html;
        try_files $uri $uri/ /merck/index.html;
    }

    error_page 404 /merck/404.html;
    location = /404.html {
        root /opt/cisco;
    }

    access_log /cohesion/log/access.log main;
    error_log /cohesion/log/error.log;

}
```

5. You can see the location block /merck in the above file. Now add the following blocks after the `/merck` block.

```
location /websocket {
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $http_host;
        proxy_read_timeout 5m;
        proxy_send_timeout 5m;
        proxy_pass https://cohesion-backend:8192;

        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection $connection_upgrade;
    }

    location /carbon {
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $http_host;
        proxy_read_timeout 5m;
        proxy_send_timeout 5m;
        proxy_pass https://cohesion-wso2:9443/carbon;
    }

    location /store {
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $http_host;
        proxy_read_timeout 5m;
        proxy_send_timeout 5m;
        proxy_pass https://cohesion-wso2:9443/store;
    }

    location /publisher {
        proxy_set_header X-Forwarded-Host $host;
        proxy_set_header X-Forwarded-Server $host;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header Host $http_host;
        proxy_read_timeout 5m;
        proxy_send_timeout 5m;
```

```
    proxy_pass https://cohesion-wso2:9443/publisher;
  }

  location /registry {
    proxy_set_header X-Forwarded-Host $host;
    proxy_set_header X-Forwarded-Server $host;
    proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
    proxy_set_header Host $http_host;
    proxy_read_timeout 5m;
    proxy_send_timeout 5m;
    proxy_pass https://cohesion-wso2:9443/registry;
  }

  location /token {
     proxy_set_header X-Forwarded-Host $host;
     proxy_set_header X-Forwarded-Server $host;
     proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
     proxy_set_header Host $http_host;
     proxy_read_timeout 5m;
     proxy_send_timeout 5m;
     proxy_pass https://cohesion-wso2:8243/token;
  }

  location /revoke {
     proxy_set_header X-Forwarded-Host $host;
     proxy_set_header X-Forwarded-Server $host;
     proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
     proxy_set_header Host $http_host;
     proxy_read_timeout 5m;
     proxy_send_timeout 5m;
     proxy_pass https://cohesion-wso2:8243/revoke;
  }

  location /api {
     proxy_set_header X-Forwarded-Host $host;
     proxy_set_header X-Forwarded-Server $host;
     proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
     proxy_set_header Host $http_host;
     proxy_read_timeout 5m;
     proxy_send_timeout 5m;
     proxy_pass https://cohesion-wso2:8243/api;
  }
  ```
```

5. Now restart the nginx server
```

6. Capture the shell script output to the following file.
/home/ubuntu/logs/mylog.log
7. Replace the IP 127.0.0.1 with 0.0.0.0
8. Write a cron script which look for the log file size and if the size is more than 10MB then delete the log file
and create a log file with empty content.