Practical Exercise: Deploy WSO2 API Manager Pattern #1

Training Objective

Learn how to set up a clustered environment of WSO2 API Manager.

Business Scenario

PizzaShack needs to set up a highly available API Manager environment.

High-Level Steps

- Load Balancer (Nginx), Database (MySQL) will be set up on your instructor's machine
- WSO2 Instructor will create teams of two participants
- Install prerequisites (MySQL Workbench, Oracle JDK 1.8.x)
- Run DB Scripts to create databases
- Change product configurations
- Import Certificates to client-truststore.jks
- Configure Analytics

Detailed Instructions

Instructor

Configure Nginx

1. Follow the instructions in the documentation.
2. Edit the Nginx config file:

```
upstream sslapi.am.wso2.com {
    server {node-1-ip-address}:9443;
    server {node-2-ip-address}:9443;
    #ip_hash;
    sticky learn create=$upstream_cookie_jsessionid
        lookup=$cookie_jsessionid
    zone=client_sessions:1m;
}

upstream sslgw.am.wso2.com {
    server {node-1-ip-address}:8243;
    server {node-2-ip-address}:8243;
}

server {
    listen 80;
    server_name api.am.wso2.com;
    rewrite ^/(.*) https://api.am.wso2.com/$1 permanent;
}

server {
    listen 443 ssl;
    server_name api.am.wso2.com;
    proxy_set_header X-Forwarded-Port 443;
```

```
    ssl_certificate /etc/nginx/ssl/{cert_name};
    ssl_certificate_key /etc/nginx/ssl/{key_name};
    location / {
            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://sslapi.am.wso2.com;
        }

        access_log /etc/nginx/log/am/https/access.log;
        error_log /etc/nginx/log/am/https/error.log;
}

server {
    listen 443 ssl;
    server_name gw.am.wso2.com;
    proxy_set_header X-Forwarded-Port 443;
    ssl_certificate /etc/nginx/ssl/{cert_name};
    ssl_certificate_key /etc/nginx/ssl/{key_name};
    location / {
            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://sslgw.am.wso2.com;
        }

        access_log /etc/nginx/log/gw/https/access.log;
        error_log /etc/nginx/log/gw/https/error.log;
}
```

Prepare your lab machine

Edit the file hosts inside C:\Windows\System32\drivers\etc or /etc/hosts and add the following configurations:

```
10.80.2.36      wso2-db
10.80.2.36      ldap-colaboradores
10.80.2.36      gateway.green.com apim.green.com sso.green.com # team 1
10.80.2.36      gateway.blue.com apim.blue.com sso.blue.com # team 2
10.80.2.36      gateway.yellow.com apim.yellow.com sso.yellow.com # team 3
10.80.2.36      gateway.black.com apim.black.com sso.black.com # team 4
10.80.2.36      gateway.white.com apim.white.com sso.white.com # team 5
10.80.1.38      green-1
10.80.1.93      green-2
10.80.3.38      blue-1
10.80.1.29      blue-2
10.80.2.32      white-1
10.80.0.122     white-2
```

Create Databases

1. Set up MySQL databases following the documentation.

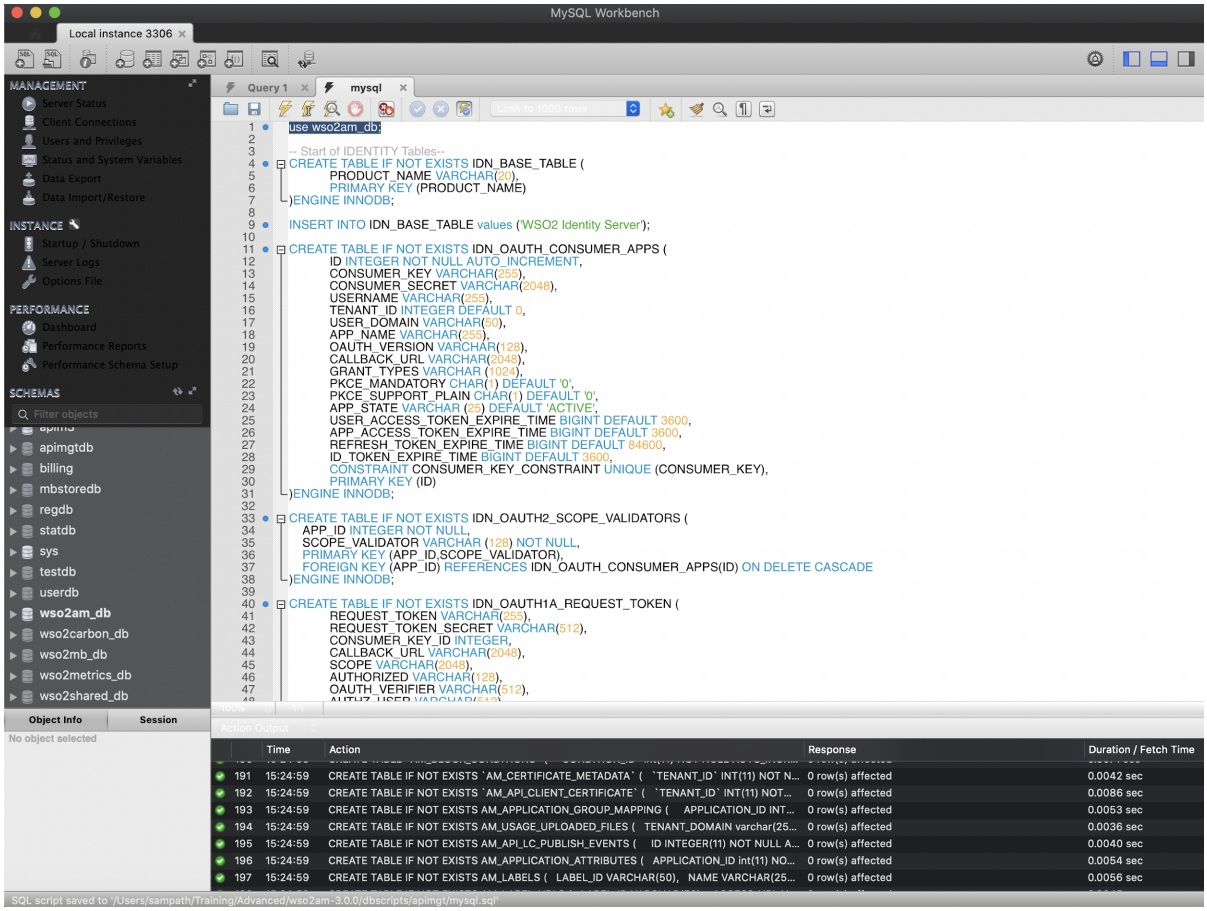
   ```
   create database wso2am_db;
   create database wso2shared_db;
   ```

Install and Configure WSO2 API Manager

1. Expand WSO2 API Manager binaries
2. From now on, this folder will be known as APIM-HOME pointing to c:\wso2\wso2am-4.2.0\
3. Run SQL scripts to create tables and other artifacts
   a. Open MySQL Workbench and connect to the server your WSO2 instructor provides you with
   b. Run scripts on databases created previously according to the following table:

DATABASE	SCRIPT
wso2am_db	<APIM-HOME>/dbscripts/apimgt/mysql.sql
wso2shared_db	<APIM-HOME>/dbscripts/mysql.sql

   c. When you open each script, remember to set the proper database to run it, for example see the following screenshot:



4. Edit the deployment.toml file to configure databases just created:

```
[database.shared_db]
type = "mysql"
url = "jdbc:mysql://localhost:3306/wso2shared_db?useSSL=false"
username = "root"
password = "root"

[database.apim_db]
type = "mysql"
url = "jdbc:mysql://localhost:3306/wso2am_db?useSSL=false"
username = "root"
password = "root"
```

5. Copy MySQL JDBC connector jar to <APIM-HOME>\repository\components\lib\
6. Edit the deployment.toml file to configure reverse proxy settings in the API Manager.

```
[transport.https.properties]
proxyPort = 443

[server]
hostname = "apim.green.com"
```

7. Edit the deployment.toml file to configure DNS names for URLs and other configs.

```
[[apim.gateway.environment]]
http_endpoint = "http://gateway.green.com"
https_endpoint = "https:/gateway.green.com"

[apim.devportal]
url = "https://apim.green.com/devportal"
```

8. You can configure Analytics by applying the configurations explained in this document.

9. Configure the Second API Manager node.

   Make a copy of the active instance configured above and use this copy as the second active instance.

10. Start the APIM Servers.

11. Validate your newly built environment by invoking APIs.

Expected Outcome

WSO2 API Manager set up on 2 machines as Active-Active configuration using two all-in-one instances fronted by Nginx and using MySQL as the Relational Database for configurations and product's data.