WSO2\_API Manager

Registering application in the WSO2\_API Manager:

1 .Download wso2\_am(1.9.1) from the below site:

<http://wso2.com/products/api-manager/>

2 .Unzip the file in any location of your choice.(Eg: D:\WSO2\wso2-old\wso2am-1.9.1)

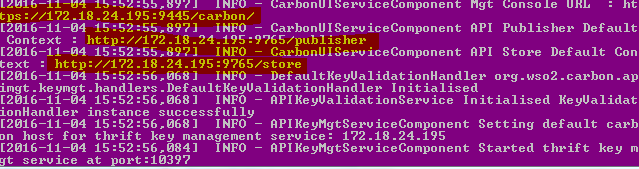
3 .Navigate to <wso2\_Home>\wso2am-1.9.1\bin\ and start the wso2server.bat (windows batch file)

It will open the command prompt, and server starts

Or

In command prompt navigate to wso2server.bat,





Wso2 server starts successfully!!!

4.In the browser ,copy those three url’s which we get when server starts

<https://172.18.24.195:9445/carbon/>

<https://172.18.24.195:9445/publisher/>

<https://172.18.24.195:9445/store/>

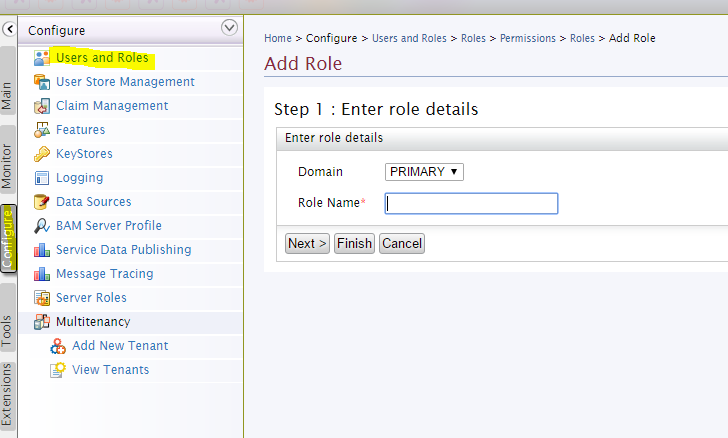
1. In <https://172.18.24.195:9445/carbon/>

**login as**

username: admin

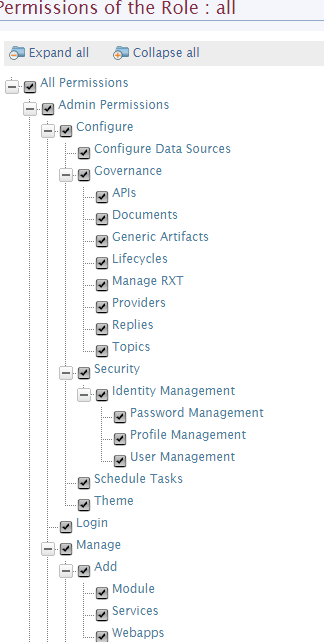
password: admin

1.click on **configure -> users and roles -> roles -> add new role**



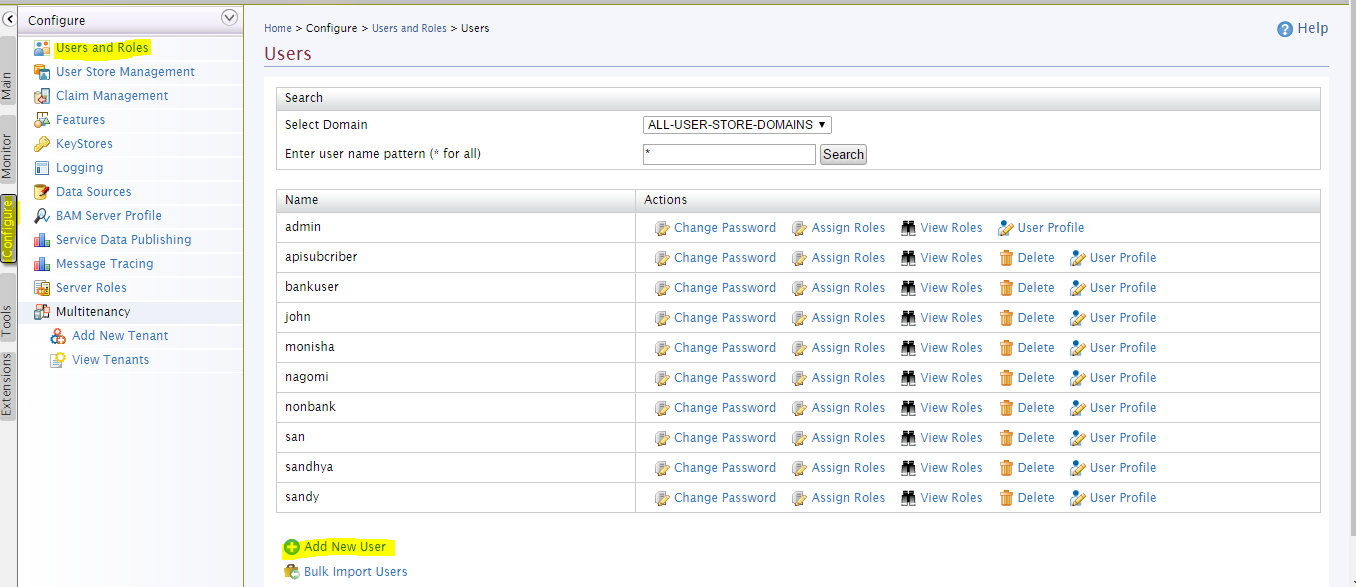
We can create new role

And assign permissions for that roles and click on **-> update**



1.click on **configure -> users and roles -> users -> addnewuser**

we can add new users



Can able to create new user **-> next**



Where we can assign roles for the specific user and click on finish

New user created successfully!!!

In publisher , <https://172.18.24.195:9445/publisher/>

login as (default)

username: admin

password: admin

or else login with the new users which has been created by following the steps given above

Registering the service

1.click Add->Design new API->start creating

**Fill in DESIGN API**

**General Details**

1.Name : Display name in the store

(eg:PizzaShack)

2.Context : URI context path of the API (case sensitive).  
 The supported formats are..   
 1./foo   
 2./foo/bar   
 3. /foo/{version}/bar (case sensitive) - allows the version to be within the context

(eg:/pizza)

3.Version : version number for the particular application

(eg:1.0.0)

4.Visibility : (eg:public)

**Public :** The API is accessible to everyone and can be advertised in multiple stores - a central store and/or non-WSO2 stores.  
 **Visible to my domain :** The API is visible to all users who are registered in the API's tenant domain.  
 **Restricted by roles :** The API is visible only to specific user roles in the tenant store that you specify.

5.Thumbnail Image: Display image

6.Description: we can describe about the application which we are registering

7.Tags: Is the keyword which is used to search

API\_Definition

URL Pattern**:**

**In url pattern:**

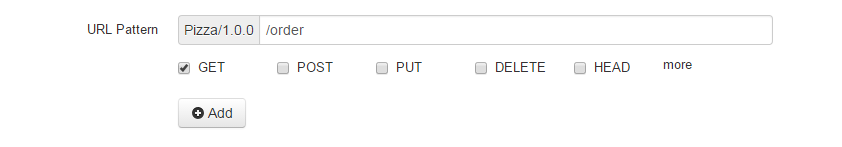
**We have to mention the pattern in which url can be displayed**

/{context}/{version}/{url pattern}

**Context , version** will be taken from the as soon as given in general details section

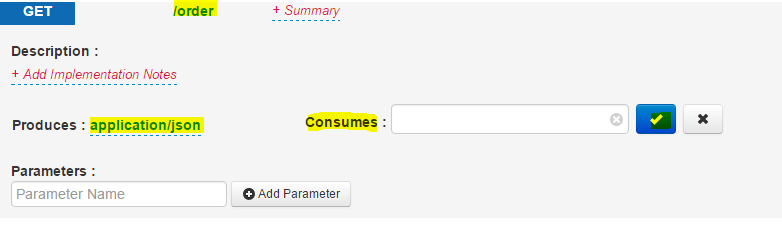
Eg: /pizza/1.0.0/order

Give the url pattern name then mention the method (get,post.put,delete,head,options) and click on ->**add**



**Click on the url pattern which we are added , and mention the media type (produces and consumes ) used in the services which we have created**

**And click on -> next:Implement>**



**Implement**

**Managed API**

**1.EndPoints**

**HTTP Endpoint**

A REST service endpoint based on a URI template

**Address Endpoint**

The direct URL of the service

**Failover Group**

The endpoints that the service tries to connect to in case of a failure. This happens in a round robin manner.

**Load Balance**

The endpoints where the incoming requests are directed to in a round robin manner. They automatically handle fail-over as well.

* Default endpoint(HTTP Endpoint)

Production Endpoint: have to mention the tomcat url in which the service is running

<http://localhost:18081/PizzaShack/pizza/order>

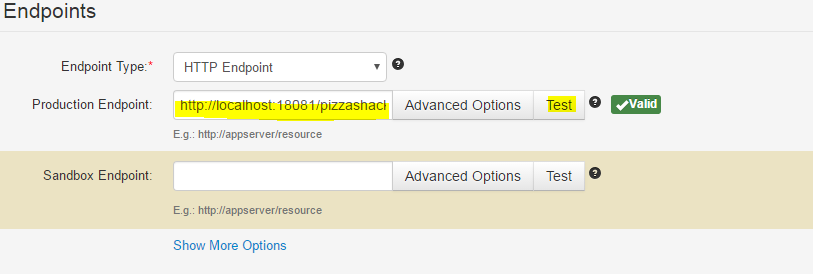
is the url in which order service is running ,that order is the url pattern ,mentioned in the previous section (url pattern) which will append at the end automatically.

Hence we have to give the production endpoint by leaving the url pattern

Which means

[**http://localhost:18081/PizzaShack/pizza**](http://localhost:18081/PizzaShack/pizza)

**and click on -> test**



It will validate the url ,whether the service is running and returns validate 

Else returns 

Click ->next

**Manage**

**Configurations**

**1.Make it has default version**

Marks one API version in a group as the default, so that it can be invoked without specifying the version number in the URL. For example, if you mark http://host:port/youtube/2.0 as the default API, requests made to http://host:port/youtube/ are automatically routed to version 2.0. If you mark an unpublished API as the default, the previous default, published API will still be used as the default until the new default API is published

**By default it will be unchecked**

**2.Tier availability**

Tiers limit the number of requests a subscriber can make to an API. An API can be exposed over one or more tiers. The Gold, Silver and Bronze tiers allow 20, 5 and 1 request/s respectively per minute.

**We can choose the limitations of our own**

**3.Transports:**

HTTP is less secure than HTTPS

and makes your API vulnerable to security threats.

**By default both are checked**

**4.Response caching:**

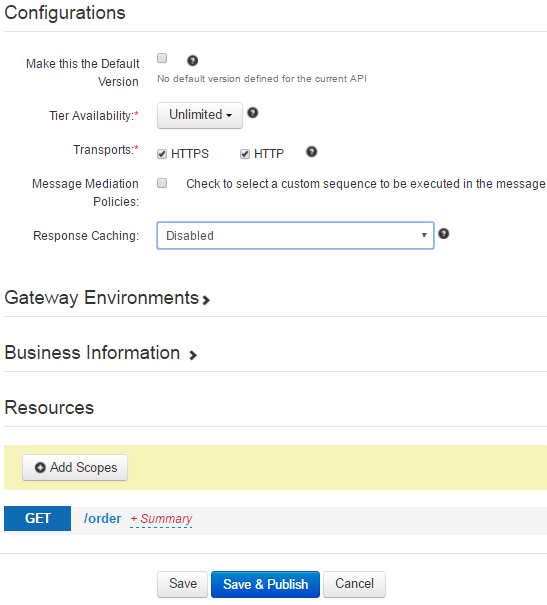
This option determines whether to cache the response messages of the API. Caching improves performance, because the backend server does not have to process the same data for a request multiple times. To offset the risk of stale data in the cache, set an appropriate timeout period when prompted.

**By default it is disabled**

Resources

If we want can add the scope.if we add that service will be available for particular group of users alone

**Then click-> save and publish**



**Now,the API is published successfully!!!**

**In the store**

<https://172.18.24.195:9445/store/>

we have to login if we have permissions to access the store ,or else we can access via default login

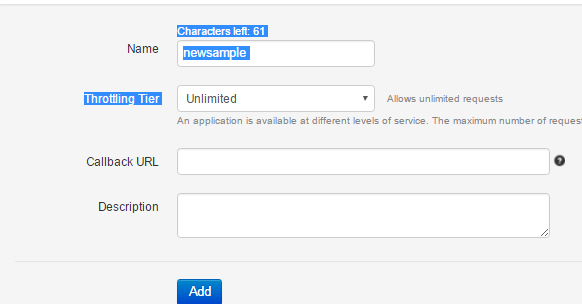
username : admin

password : admin

and can able to access the api

Myapplication

Have to create new application by using which we can access the set of api’s



Application added successfully!!!

Steps to subscribe the API’s

1.click on the API which we want .

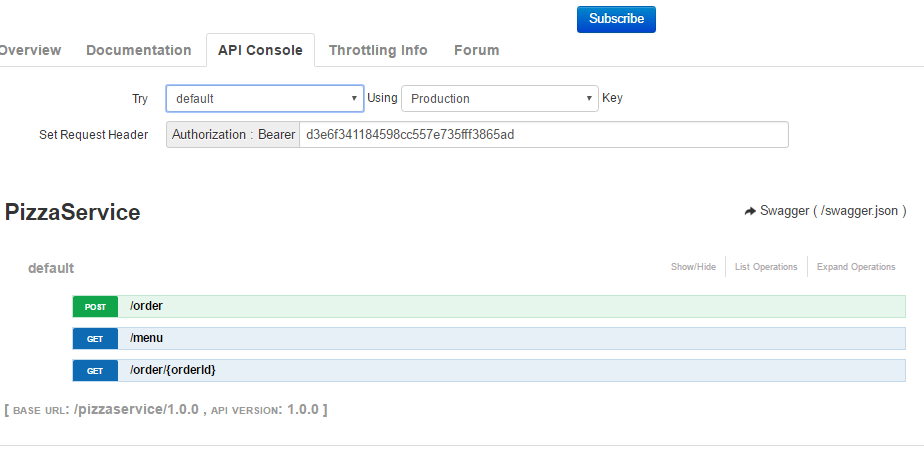
2.select the application which we have created –> click ->subscribe

It will go to the my subscription page

3.click -> generate keys ,It will generate consumer key,consumer secret key ,access token and it also show the time limit until which the key is valid

4.click the api which listed below for that particular application

In the APIConsole ,it will display like this



If we click on the methods ,it will display the output there it self

Now the service registered and subscribed successfully!!!

WSO2 API and WSO2 IS Integration

<https://docs.wso2.com/display/CLUSTER44x/Configuring+the+Pre-Packaged+Identity+Server+5.1.0+with+API+Manager+1.10.0>