

Simulation Reverse Proxy Via Nginx

Steps to start different essential processes on server:

1. Open a terminal and start a node JS process on port 8080. (Open port 8080 via security groups)

```
const express=require('express')
const app=express()

app.get('/app1',(req,res)⇒{
  res.send("This is from App1!")
})

app.listen(8080,()⇒{
  console.log("Application is running in port 8080!")
})
```

2. Open another terminal and start another node JS process on port 8081. (Open port 8081 via security groups)

```
const express=require('express')
const app=express()

app.get('/app2',(req,res)⇒{
  res.send("This is from App2!")
})

app.listen(8081,()⇒{
  console.log("Application is running in port 8081!")
})
```

3. Open another terminal and start an nginx process on port 80 using following commands:

```
sudo apt update
```

```
sudo apt install nginx
```

4. Try accessing the processes running on the port 80
(http://{public_ip/domain_name}:80), 8080
(http://{public_ip/domain_name}:8080), 8081
((http://{public_ip/domain_name}:8081) and see if the processes are accessible.
5. If the processes are accessible, then configure reverse proxy on port 80.

Step 1: `sudo vi /etc/nginx/nginx.conf` (Modify the permissions if needed)

Step 2:

```
events {  
    # Event directives...  
}  
  
http {  
    server {  
        listen 80;  
        server_name app1.deepak.com;  
  
        location / {  
            proxy_pass http://localhost:8080;  
            proxy_http_version 1.1;  
            proxy_set_header Upgrade $http_upgrade;  
            proxy_set_header Connection 'upgrade';  
            proxy_set_header Host $host;  
            proxy_cache_bypass $http_upgrade;  
        }  
    },  
}
```

```
server {  
    listen 80;  
    server_name app2.deepak.com;  
  
    location / {  
        proxy_pass http://localhost:8081;  
        proxy_http_version 1.1;  
        proxy_set_header Upgrade $http_upgrade;  
        proxy_set_header Connection 'upgrade';  
        proxy_set_header Host $host;  
        proxy_cache_bypass $http_upgrade;  
    }  
}  
  
}
```

Step 3: `sudo nginx -s reload`

6. Edit your hosts file as administrator/root and put below entries.

```
public_ip  app1.deepak.com  
public_ip  app2.deepak.com
```

6. visit the below urls and see if you able to access your applications.

```
http://app1.deepak.com/app1  
http://aap2.deepak.com/app2
```