ASSESSMENT - 5

NGINX

1. **What is the advantage of using a “reverse proxy server”?**

ANS : A site can benefit in several ways by implementing a reverse proxy server:

- Avoid the expense of installing another web server. A reverse proxy server increases the capacity of existing servers.
- Serve more requests for static content and thus free up bandwidth to serve more dynamic content.
- Reduce operating expense by increasing bandwidth.
- Provide a single point of control over who can access HTTP servers, and which servers can be accessed.
- Decrease response time of web pages and accelerate download time, enhancing the experience of web site users.
- Provide another layer of protection by hiding the internal IP address.

2. **Why and where Nginx is a better choice than apache.**

ANS : 1. Based on event driven architecture.

2. Doesn’t create a new process for a new request.

3. Memory consumption is very low for serving static pages.

3. **What are worker nodes and worker connections? How to calculate the max server capacity using the above two?**

ANS : Defines the number of worker processes nginx is running. Default value is the number of processors in the system.

Defines maximum number of simultaneous connection.
Default value is 768.

Maximum number of connections = worker_processes * worker_connections

4. **From what directory will NGINX automatically load server (virtual host) configurations when using the default /etc/nginx/nginx.conf configuration?**

ANS : etc/nginx/sites-enabled/

5. **How to configure different log_format for different “location” block/directive?**

ANS :

```
        # Logging Settings
        ##
        log_format  custom '$remote_addr - $remote_user [$time_local] '
                                '"$request" $status $body_bytes_sent '
                                '"$http_referer" "$http_user_agent" '
                                '"$http_x_forwarded_for" $request_id '
                                '$geoip_country_name $geoip_country_code '
                                '$geoip_region_name $geoip_city ';

        access_log /var/log/nginx/access.log custom;
        error_log /var/log/nginx/error.log;
```

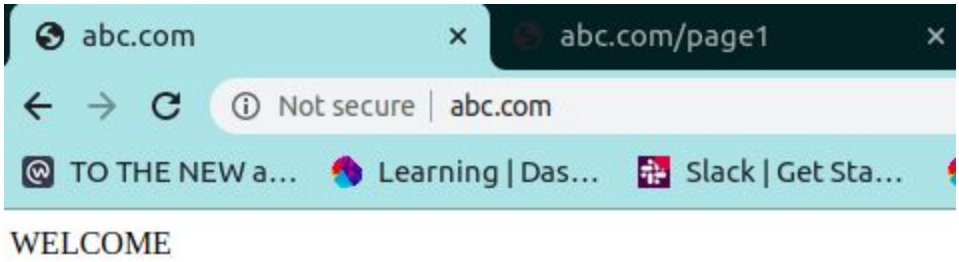
```
garima@garima:/etc/nginx$ cd /var/log/nginx/
garima@garima:/var/log/nginx$ ls
access.log  access.log.1  error.log
garima@garima:/var/log/nginx$ tail -n 1 access.log
10.1.225.131 - - [13/Feb/2020:12:17:19 +0530] "GET /HNAP1/ HTTP/1.1" 404 152 "-" "-"
garima@garima:/var/log/nginx$ sudo nginx -t
[sudo] password for garima:
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

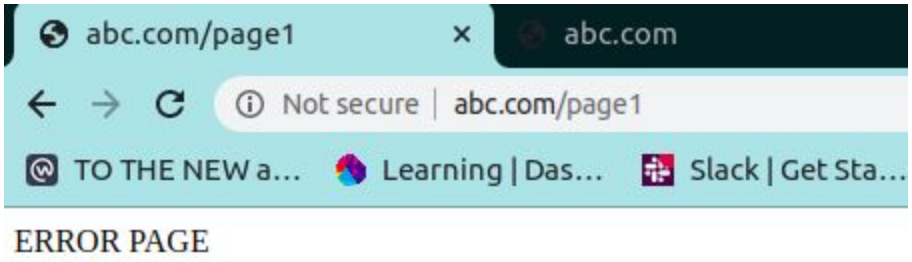
```
garima@garima:/var/log/nginx$ service nginx restart
garima@garima:/var/log/nginx$ tail -n 1 access.log
::1 - - [13/Feb/2020:15:19:55 +0530] "GET / HTTP/1.1" 304 0 "-" "Mozilla/5.0 (X1
1; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.130 Saf
ari/537.36" "-" 47878658f34860079a8852057c2a9ab3 - - - -
garima@garima:/var/log/nginx$
```

6. **Host a site <u>ABC.COM</u>**
    1. **Create an index page and a fail-safe page. If a page for URI is not available, the fail-safe page is served.**

```
garima@garima:~$ cd /var/www/html/
garima@garima:/var/www/html$ ls
abc.html  error.html  xyz.html
garima@garima:/var/www/html$ cat abc.html
WELCOME
garima@garima:/var/www/html$ cat error.html
ERROR PAGE
garima@garima:/var/www/html$ cd /etc/
garima@garima:/etc$ head -n 2 hosts
127.0.0.1       abc.com
127.0.1.1       garima
garima@garima:/etc$ cd nginx/sites-available/
garima@garima:/etc/nginx/sites-available$ ls
abc.com  default  xyz.com
```

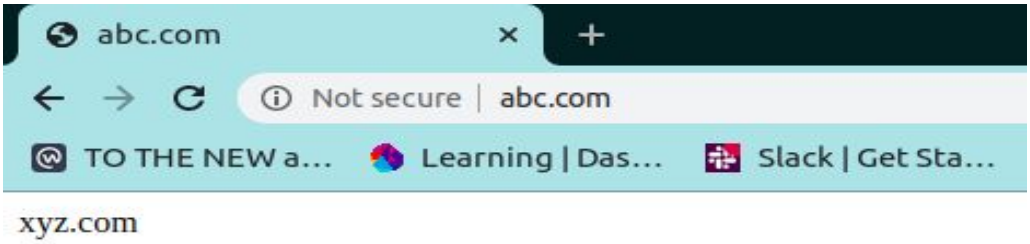
```
garima@garima:/etc/nginx/sites-available$ cat abc.com
server {
        listen 80 default_server;
        listen [::]:80 default_server;

        root /var/www/html;

        index abc.html ;

        error_page 404 /error.html;
        server_name abc.com;

        location / {
                # First attempt to serve request as file, then
                # as directory, then fall back to displaying a 404.
                try_files $uri $uri/ =404;
        }

}
garima@garima:/etc/nginx/sites-available$ cd ../sites-enabled/
garima@garima:/etc/nginx/sites-enabled$ ls
abc.com
```





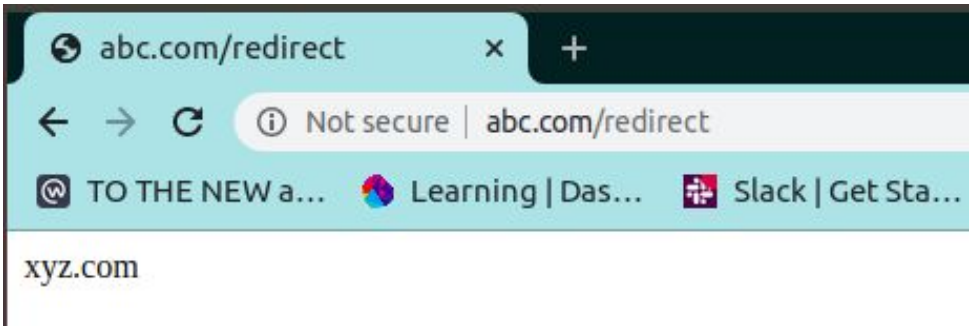
2. proxy pass to a website xyz.com on a particular URI.

```
garima@garima:/etc/nginx/sites-enabled$ sudo ln -s /etc/nginx/sites-available/xyz.com .
garima@garima:/etc/nginx/sites-enabled$ ls
abc.com  xyz.com
garima@garima:/etc/nginx/sites-enabled$ sudo vim abc.com
garima@garima:/etc/nginx/sites-enabled$ sudo systemctl restart nginx
garima@garima:/etc/nginx/sites-enabled$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
garima@garima:/etc/nginx/sites-enabled$ cat abc.com
server {
        listen 80 default_server;
        listen [::]:80 default_server;

        root /var/www/html;

        index abc.html ;

        error_page 404 /error.html;
        server_name abc.com;

        location / {
                proxy_pass http://127.0.0.1:81/;
                try_files $uri $uri/ =404;
        }

}
garima@garima:/etc/nginx/sites-enabled$
```



3. redirect to above URI on /redirect/

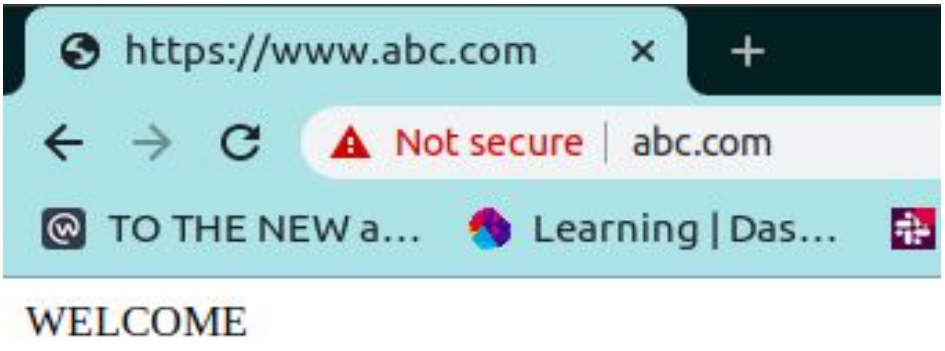
```
File  Edit  View  Search  Terminal  Tabs  Help
garima@garima: /var/www/html
garima@garima:/etc/nginx/sites-enabled$ sudo vim abc.com
garima@garima:/etc/nginx/sites-enabled$ cat abc.com
server {
        listen 80 default_server;
        listen [::]:80 default_server;

        root /var/www/html;

        index abc.html ;

        error_page 404 /error.html;
        server_name abc.com;

        location /redirect {
                proxy_pass http://127.0.0.1:81/;
                #try_files $uri $uri/ =404;
        }

}
garima@garima:/etc/nginx/sites-enabled$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
garima@garima:/etc/nginx/sites-enabled$ sudo systemctl restart nginx
garima@garima:/etc/nginx/sites-enabled$
```



4. **perform an HTTP to HTTPS redirection including non-www to www redirection.**

```
garima@garima:/etc/nginx$ sudo apt-get install openssl
Reading package lists... Done
Building dependency tree
Reading state information... Done
openssl is already the newest version (1.1.1-1ubuntu2.1~18.04.5).
0 upgraded, 0 newly installed, 0 to remove and 26 not upgraded.
garima@garima:/etc/nginx$ sudo openssl req -x509 -nodes -days 365 -newkey rsa:20
48 -keyout /etc/nginx/ssl/private.key -out /etc/nginx/ssl/public.pem
Can't load /home/garima/.rnd into RNG
140427662741952:error:2406F079:random number generator:RAND_load_file:Cannot ope
n file:../crypto/rand/randfile.c:88:Filename=/home/garima/.rnd
Generating a RSA private key
.....................................................+++++
.....................................................+++++
writing new private key to '/etc/nginx/ssl/private.key'
```

```
garima@garima:/etc/nginx/sites-enabled$ sudo vim abc.com
garima@garima:/etc/nginx/sites-enabled$ cat abc.com
server {
        listen 80 ;

        server_name abc.com www.abc.com;

        return 302 https://www.abc.com;

}
server{

        listen 443 ssl;
        server_name www.abc.com;

        root /var/www/html;
        index abc.html;
        ssl_certificate /etc/nginx/ssl/public.pem;
        ssl_certificate_key /etc/nginx/ssl/private.key;
}
garima@garima:/etc/nginx/sites-enabled$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
garima@garima:/etc/nginx/sites-enabled$ sudo systemctl restart nginx
garima@garima:/etc/nginx/sites-enabled$
```



5. Allow access to a set of particular IPs on a location block and return 405 to other IPs no matter if the page in that location exists.

```
server {
        listen 80 ;

        server_name abc.com;
        root /var/www/html/;
        index abc.html;
        allow 10.1.211.99;
        deny all;
        error_page 403 404 =405 /error.html;
        location /error.html{
                return 405;
        }
}
```

6. Place your images at /var/www/html/images. Only accept jpg/png/jpeg. Discard rest

7. Create a load balancer with 5 backends. Explain different types of load balancing methods.

i) Round Robin

ii) Least connection:

```
Upstream backend{
Least_conn;
Server backend1.example.com;
Server backend2.example.com;
}
```

iii) IP hash:

```
Upstream backend{
ip_hash;
Server backend1.example.com;
Server backend2.example.com;
}
```

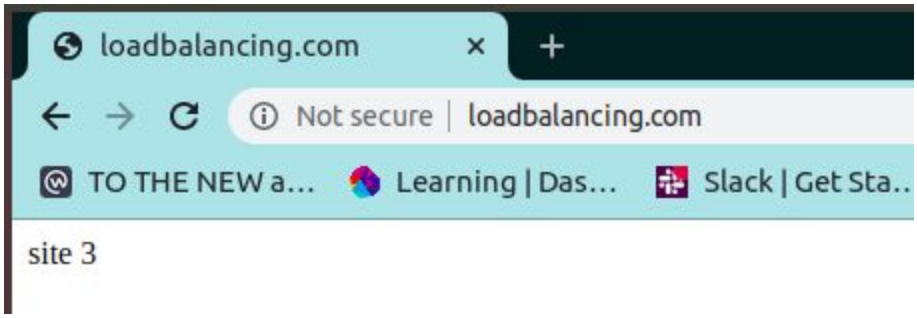
iv) Least time:

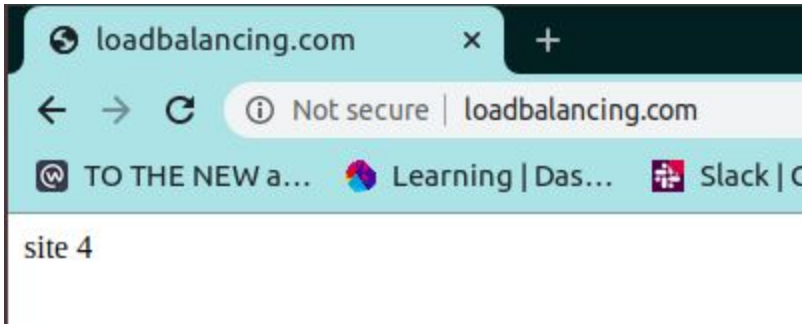
```
Upstream backend{
Least_time header;
Server backend1.example.com;
Server backend2.example.com;
}
```

```
garima@garima: /etc/nginx/sites-enabled ×        garima
garima@garima:~$ cd /var/www/html/
garima@garima:/var/www/html$ sudo vim index1.html
[sudo] password for garima:
garima@garima:/var/www/html$ sudo vim index2.html
garima@garima:/var/www/html$ sudo vim index3.html
garima@garima:/var/www/html$ sudo vim index4.html
garima@garima:/var/www/html$ sudo vim index5.html
garima@garima:/var/www/html$
```

```
garima@garima:/etc/nginx/sites-enabled$ sudo vim load-balancing
[sudo] password for garima:
garima@garima:/etc/nginx/sites-enabled$ sudo vim load-balancing
garima@garima:/etc/nginx/sites-enabled$ cd ../sites-available/
garima@garima:/etc/nginx/sites-available$ sudo vim balancing
garima@garima:/etc/nginx/sites-available$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
garima@garima:/etc/nginx/sites-available$ sudo systemctl restart nginx
garima@garima:/etc/nginx/sites-available$ cd ../sites-enabled/
garima@garima:/etc/nginx/sites-enabled$ sudo ln -s /etc/nginx/sites-available/ba
lancing .
garima@garima:/etc/nginx/sites-enabled$ sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
garima@garima:/etc/nginx/sites-enabled$ sudo systemctl restart nginx
garima@garima:/etc/nginx/sites-enabled$
```

```
File  Edit  View  Search  Terminal  Help
127.0.0.1       localhost abc.com www.abc.com xyz.com loadbalancing.com
127.0.1.1       garima
```

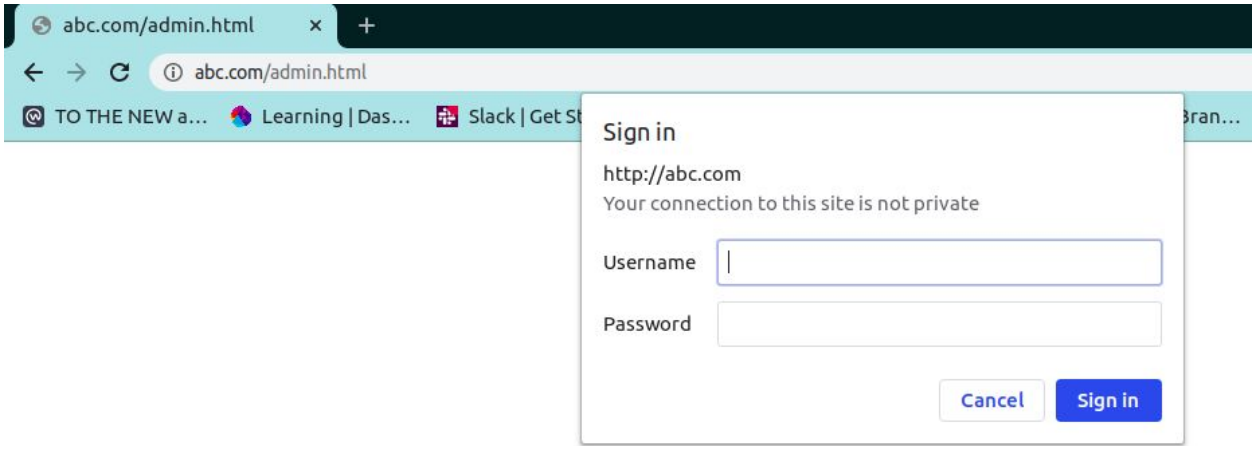





```
garima@garima:/etc/nginx/sites-enabled$ cat load-balancing
upstream balance{
        server 127.0.0.1:82;
        server 127.0.0.1:83;
        server 127.0.0.1:84;
        server 127.0.0.1:85;
        server 127.0.0.1:86;
}
server{
        listen 80;
        server_name loadbalancing.com;
        location /{
        proxy_pass http://balance;
        }

}
```

```
garima@garima:/etc/nginx/sites-enabled$ cat balancing
server{
        listen 82;
        root /var/www/html;
        index index1.html;
        server_name 127.0.0.1;
}
server{
        listen 83;
        root /var/www/html;
        index index2.html;
        server_name 127.0.0.1;
}
server{
        listen 84;
        root /var/www/html;
        index index3.html;
        server_name 127.0.0.1;
}
server{
        listen 85;
        root /var/www/html;
        index index4.html;
        server_name 127.0.0.1;
}
server{
        listen 86;
        root /var/www/html;
        index index5.html;
        server_name 127.0.0.1;
}
```

8. Setup Basic Auth (Popup asking for username and password) in a particular location block. (The Basic Auth should not be asked for TTN IP)




```
garima@garima:/etc/nginx/sites-enabled$ cat abc.com
server {
        listen 80 ;

        server_name abc.com;
        root /var/www/html/;
        index abc.html;
        error_page 404 /error.html;
        location = /admin.html{
                auth_basic "login required";
                auth_basic_user_file /etc/nginx/.htpasswd;
        }

}
```

```
garima@garima:/etc/nginx/sites-enabled$ sudo apt-get install apache2-utils
Reading package lists... Done
Building dependency tree
Reading state information... Done
apache2-utils is already the newest version (2.4.29-1ubuntu4.11).
0 upgraded, 0 newly installed, 0 to remove and 26 not upgraded.
```

```
garima@garima:/etc/nginx/sites-enabled$ sudo htpasswd -c /etc/nginx/.htpasswd admin
New password:
Re-type new password:
Adding password for user admin
garima@garima:/etc/nginx/sites-enabled$
```