**INTIVION ACCOUNT REVIEW**

**29/05/2023**

**TOTAL SERVICES USING IN INTIVION ACCOUNT**

**Elastic Compute Cloud**

**Relational Database Service**

**Elastic File System**

**OpenSearch Service**

**Data Transfer**

**Elastic Load Balancing**

**Route 53**

**Secrets Manager**

**Simple Storage Service**

**WAF**

**CloudWatch**

**Cost Explorer**

**Key Management Service**

**DocumentDB (with MongoDB compatibility)**

**CloudFront**

**EC2 INSTANCES: -**

1. **EC2 servers** (Total): - **63** EC2 Instances.
2. **EC2 servers** (running): - There are **44** EC2 Instances are in running.
3. **EC2 servers** (stopped): - There are **19** EC2 Instances are stopped.

**LOAD BALANCER: -**

1. masterweb-alb
2. Masterweb-alb2

**TARGET GROUP: -**

1. faceforexdotcom-TG
2. forexelite-new
3. group-1-websites
4. group-2-websites
5. group-5-new-tg
6. intivion-com-prod-TG
7. procashier-TG
8. group-3-websites
9. group-4-websites-latest-TG
10. group-5-websites-latest-TG

**RDS: -** imlstlprod-mysql-aurora-master-cluster.

**EFS: -**

1. Crmlogs
2. Masterlogs

**ROUTE 53: -** There are total **193** Hosted Zones.

**S3 BUCKET: -** There are total **194** S3 Buckets.

**CLOUDFRONT: -** There are total **178** CloudFront distributions.

**WAF: -**masterdomain-WAF.

**Regular Tasks** **by Intivion** (environment creations)

1) DNS Setup (Route 53 and ACM)

2) Issue Certificate in both Region Ireland and North Virginia (two domains [ domain.com, \*.domain.com]).

3) S3 Bucket Creation (Bucket Name Should be www.website name).

4) Cors Policy setup in s3 (by copying from boostrade.com) .... same configurations (for reference).

5) load balancer setup (check which target group is attached to LB) add ACM certificate to the specific LB, set rules in LB.

6) CloudFront setup, for 1 domain we must have create 3 CloudFront distribution (Prod, Clientzone , assets and offload), for reference (copy from boostrade)

7) add all the domains in Route53 that you have created, for example (create records for domain.com, clientzone.domainname.com, www.domain.com, assets.domain.com, offload.domain.com)

7) **nginx** file creation: -

**Syntax: -** cd /etc/nginx/conf.d

create two nginx configuration files: -1) **domain name.conf** and 2) **clientzone domain name.conf**

e.g.:- bitmarket-trader.com.conf, clientzone.bitmarket-trader.com.conf)

**Syntax: -** vi domainname.conf

8) applogs creations (cd /var/www/applogs file creation according to domain mkdir filename, cd file creates sessions and weblog file inside domain)

9) Create an AMI of the server and update it in the Launch Configuration and autoscaling.

**Nginx config files**

# nginx configuration for new environment

1) change the directory to nginx (**cd /etc/nginx/**).

2) List the directory of nginx (**ls**) ...you will see conf.d file.

3) change the directory to the conf.d file (cd conf.d).

4) List the directory of conf.d file (**ls**)...you will get all the domain names that have been configured in nginx/conf.d.... (insert into any domain file and copy the config of that domain) ...for reference.

5) insert into your new domain (**sudo vi domain name.conf**)...paste here that you have copied conf file of domain.

6) same for clientzone (sudo vi clientzone.domain name.conf)

7) make the directory for applogs creation in applogs (mkdir domain name) ...for domain name and clientzone.domain name)

8) Check whether the domain conf file is successfully executed or not (**nginx -t**)

9) if (nginx –t) output will give **successful OK** then restart nginx server (**systemctl restart nginx**)

**Sample config.d file Creation**

1. Change the directory to **cd /etc/nginx/conf.d**
2. Create a new conf file for new domain name for example: - **vi xyz.com.conf** and **vi clientzone.xyz.com.conf** … (we have create 2 conf file for prod and for clientzone).
3. Here is a sample code for **xyz.com.conf** and for **clientzone.xyz.com.conf**
4. For **vi xyz.com.conf**

server {

listen 80;

server\_name **xyz.com**  **www.xyz.com**;

client\_max\_body\_size 2M;

add\_header Content-Security-Policy upgrade-insecure-requests;

access\_log /var/www/applogs/**xyz.com**/weblogs/access.log main;

error\_log /var/www/applogs/**xyz.com**/weblogs/error.log error;

root /var/www/**xyz.com**/html;

index index.php index.html index.htm;

location / {

try\_files $uri $uri/ /index.php?q=$uri&$args;

}

location = /xmlrpc.php {

deny all;

access\_log off;

log\_not\_found off;

return 444;

}

location ~ \.php$ {

root /var/www/**xyz.com**/html;

fastcgi\_pass unix:/var/run/php-fpm/www.sock;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

include fastcgi\_params;

}

}

1. For **vi clientzone.xyz.com.conf**

server {

listen 80;

server\_name **clientzone.xyz.com**  **www.clientzone.xyz.com**;

client\_max\_body\_size 2M;

add\_header Content-Security-Policy upgrade-insecure-requests;

access\_log /var/www/applogs/**clientzone.xyz.com**/weblogs/access.log main;

error\_log /var/www/applogs/**clientzone.xyz.com**/weblogs/error.log error;

root /var/www/**clientzone.xyz.com**/html;

index index.php index.html index.htm;

location / {

try\_files $uri $uri/ /index.php?q=$uri&$args;

}

location = /xmlrpc.php {

deny all;

access\_log off;

log\_not\_found off;

return 444;

}

location ~ \.php$ {

root /var/www/**clientzone.xyz.com**/html;

fastcgi\_pass unix:/var/run/php-fpm/www.sock;

fastcgi\_index index.php;

fastcgi\_param SCRIPT\_FILENAME $document\_root$fastcgi\_script\_name;

include fastcgi\_params;

}

}

1. Change the directory to **cd /var/www/**