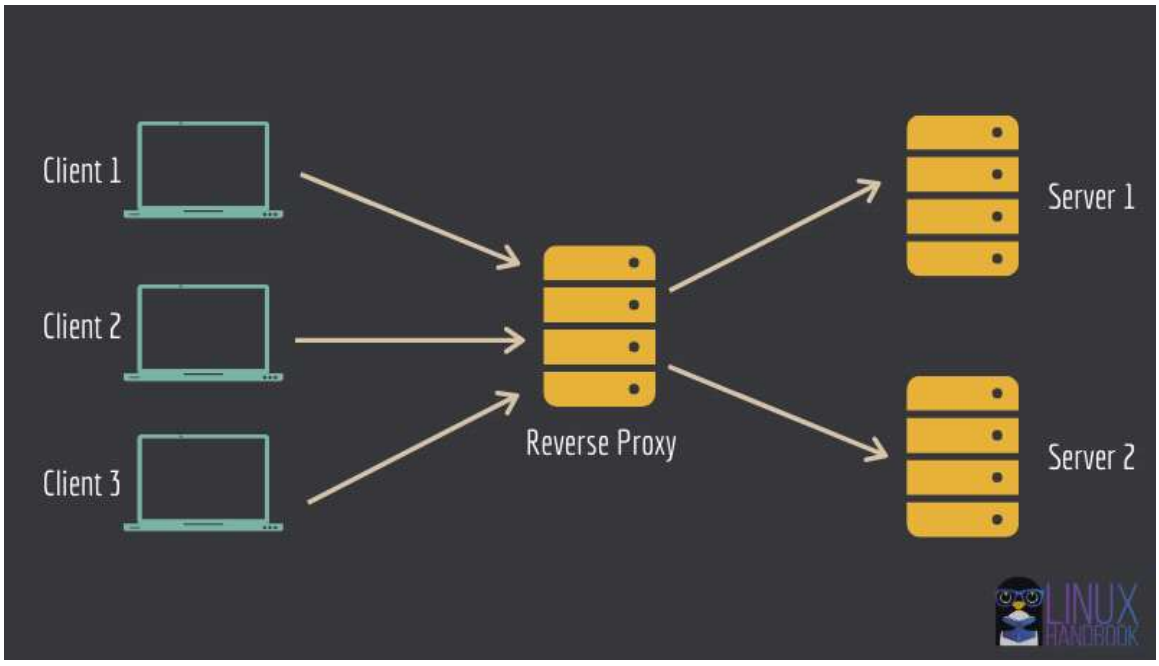
Configuring Nginx as Reverse Proxy and Load Balancer



Reverse Proxy:

- In reverse proxy we set different servers in private subnet so that they are not in direct contact with public network. for accessing public network their is need for reverse proxy that help to secure that servers from direct contact.

- it also store cache.

- reverse proxy can be configure as loadbalancer.

- we can use to prevent the web server by making it in private subnet.

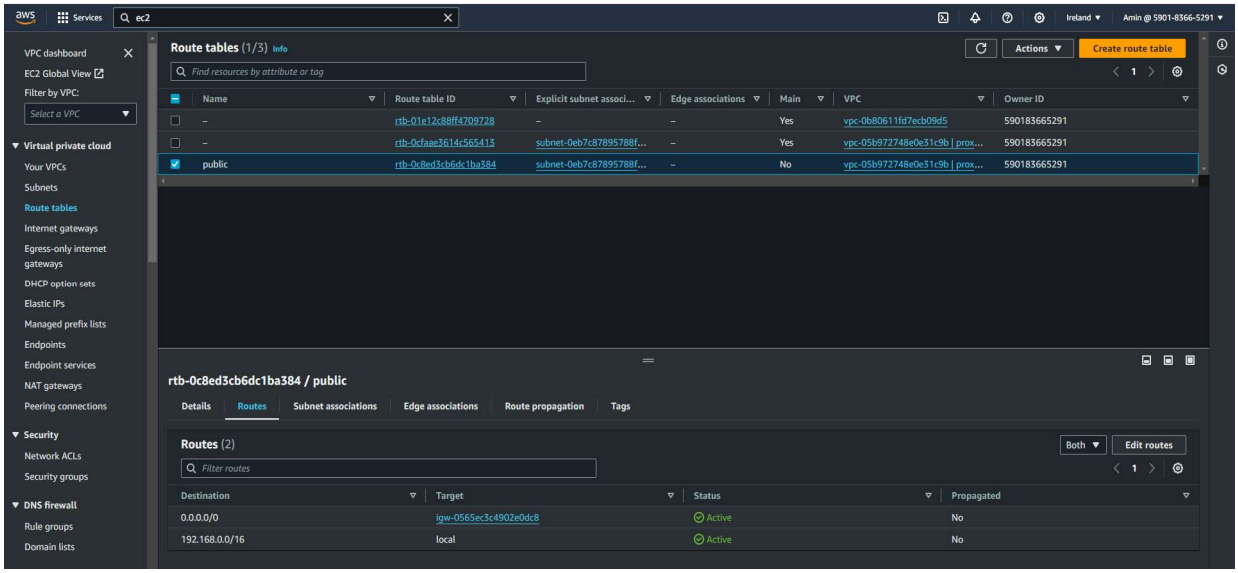
==========

Step 1:

- First create the private subnet and public subnet

- To make subnet public and private, create route table in which first route table has public subnet and attached route of IG.

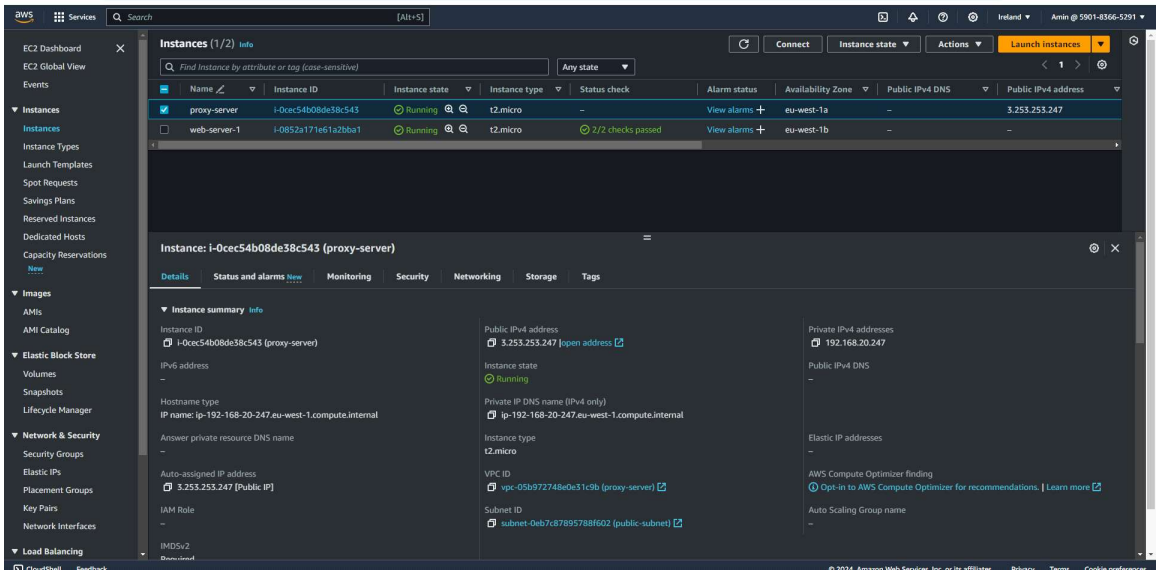
-Create second route table in which private subnet and attached route NAT gateway



Step 2:

- Make ec2-instances, first instance is name of web-server-1 and attched it on private subnet.

-second instance is name of nginx-proxy-server attached it public subnet and choose ubuntu AMI.



Step 3:

- get ssh of the web-server-1 by the nginx-proxy server.

commands : nginx-proxy server

- vi test.pem----------paste the private key.
- chmod 400 test.pem ----------add permission
- ssh -i test.pem ec2-user@<private IP of web-server-1> ----------it connect to private server
- yum update
- yum install httpd -y
- systemctl start httpd

- echo "Hello Amin welcome nginx-proxy" > /var/www/html/index.html
- curl localhost <IP >
- cd /etc/httpd/httpd.conf
- vi httpd.conf----------change the default port 80 to 8080
- systemctl restart httpd
- Now---curl localhost <IP>:8080 ----It will show the web server.
- exit

Now come to the nginx-proxy server and install nginx

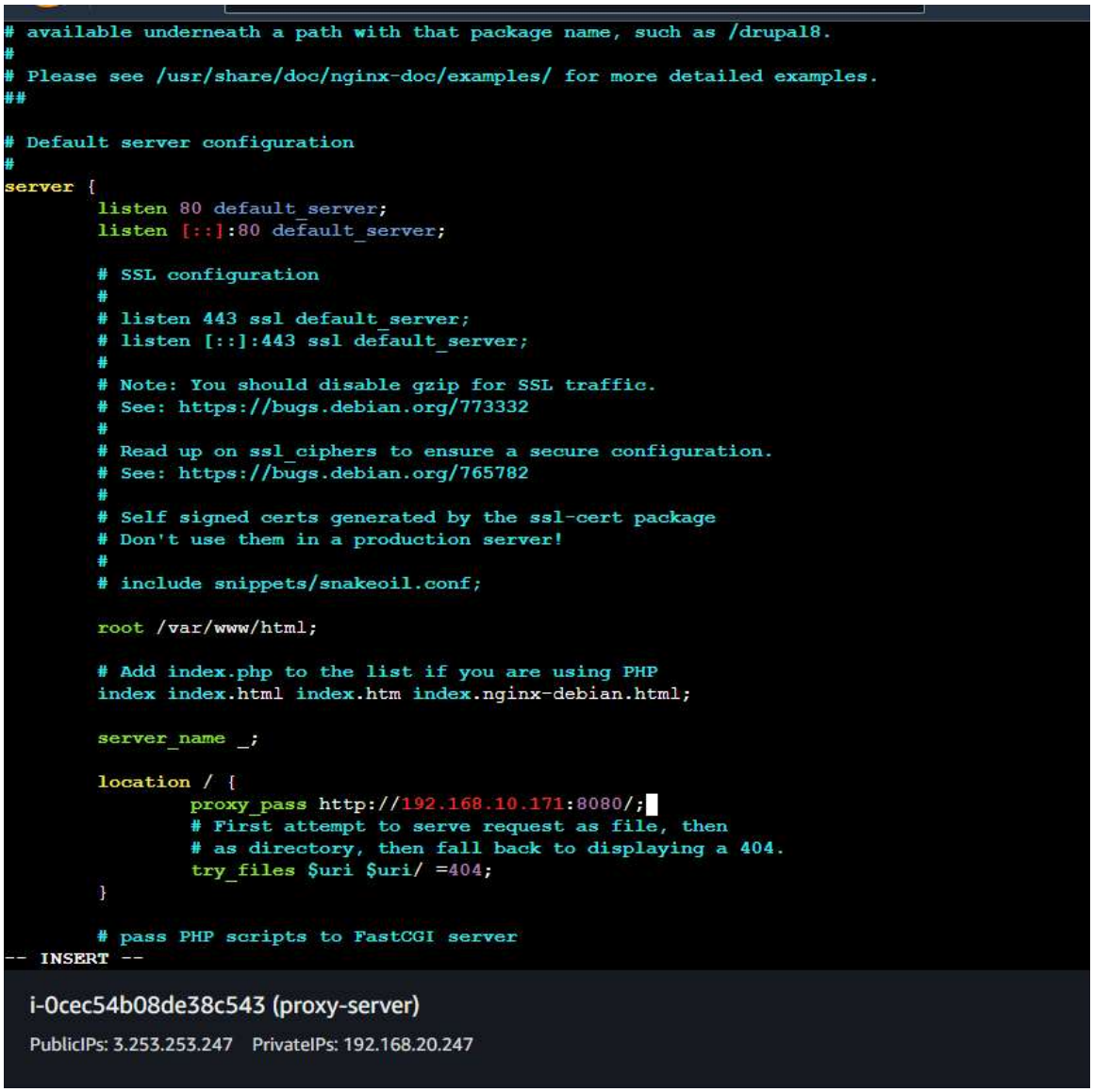
yum install nginx -y

cd /etc/nginx/sites-avilable ----------here you will find the default file.

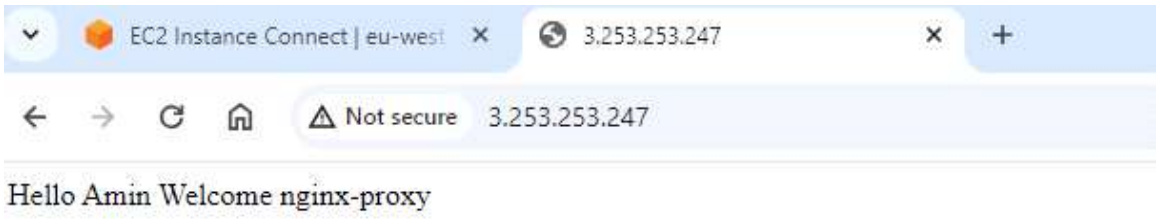
vi default ----------edit the default file change here

```
root@ip-192-168-20-247:/etc/nginx# ls -l
total 64
drwxr-xr-x 2 root root 4096 May 30  2023 conf.d
-rw-r--r-- 1 root root 1125 May 30  2023 fastcgi.conf
-rw-r--r-- 1 root root 1055 May 30  2023 fastcgi_params
-rw-r--r-- 1 root root 2837 May 30  2023 koi-utf
-rw-r--r-- 1 root root 2223 May 30  2023 koi-win
-rw-r--r-- 1 root root 3957 May 30  2023 mime.types
drwxr-xr-x 2 root root 4096 May 30  2023 modules-available
drwxr-xr-x 2 root root 4096 Feb 23 08:54 modules-enabled
-rw-r--r-- 1 root root 1447 May 30  2023 nginx.conf
-rw-r--r-- 1 root root  180 May 30  2023 proxy_params
-rw-r--r-- 1 root root  636 May 30  2023 scgi_params
drwxr-xr-x 2 root root 4096 Feb 23 08:54 sites-available
drwxr-xr-x 2 root root 4096 Feb 23 08:54 sites-enabled
drwxr-xr-x 2 root root 4096 Feb 23 08:54 snippets
-rw-r--r-- 1 root root  664 May 30  2023 uwsgi_params
-rw-r--r-- 1 root root 3071 May 30  2023 win-utf
root@ip-192-168-20-247:/etc/nginx# cd sites-available/
root@ip-192-168-20-247:/etc/nginx/sites-available# ls
default
root@ip-192-168-20-247:/etc/nginx/sites-available# vi default
```

in location write the: proxy_pass http://<IP of web-server 1>:8080/;



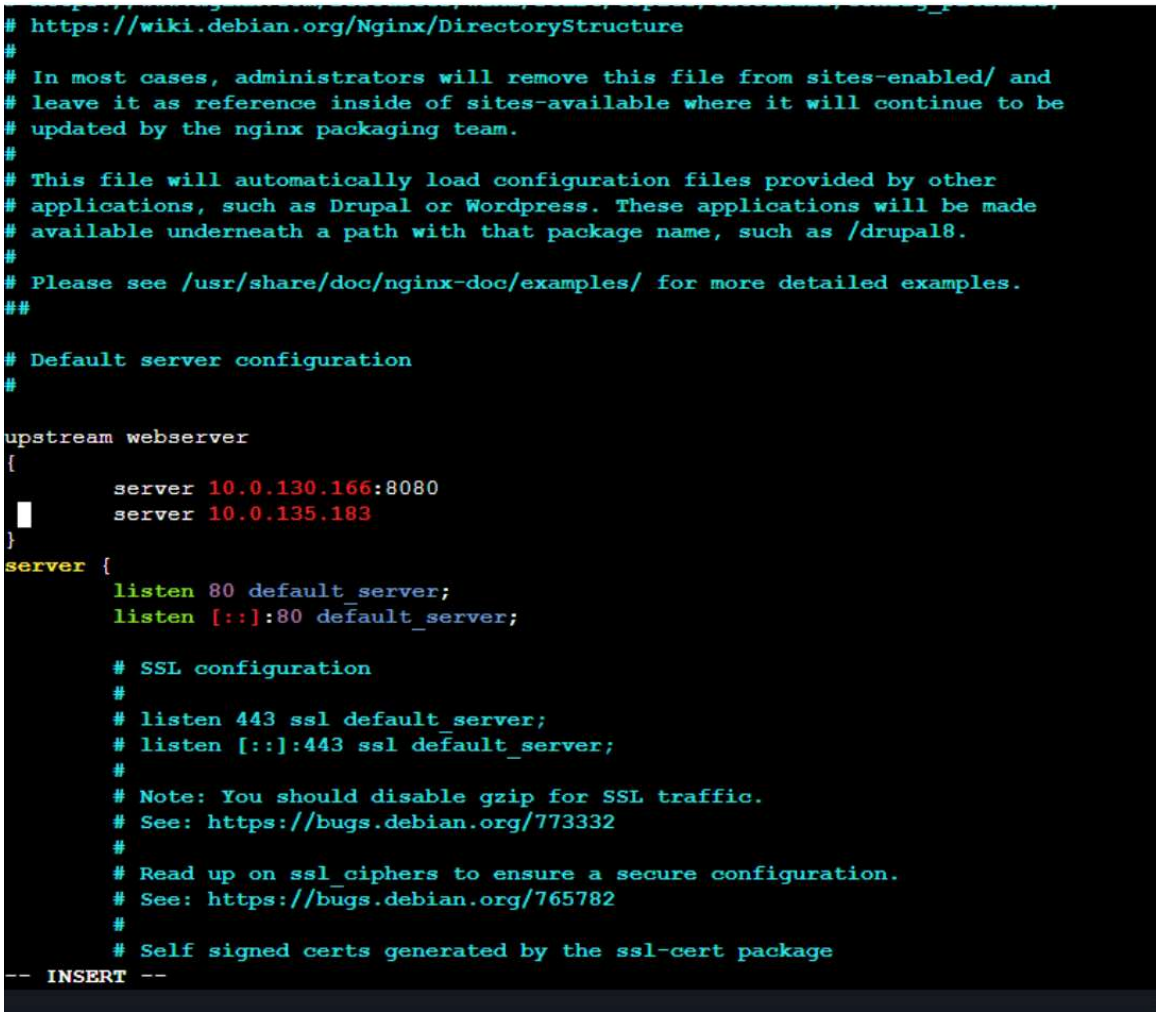
- Now Hit the IP of nginx-proxy(public instance ip) now it automaticaly redirect to the <IP of webser>:8080



==========

Nginx as Loadbalancer

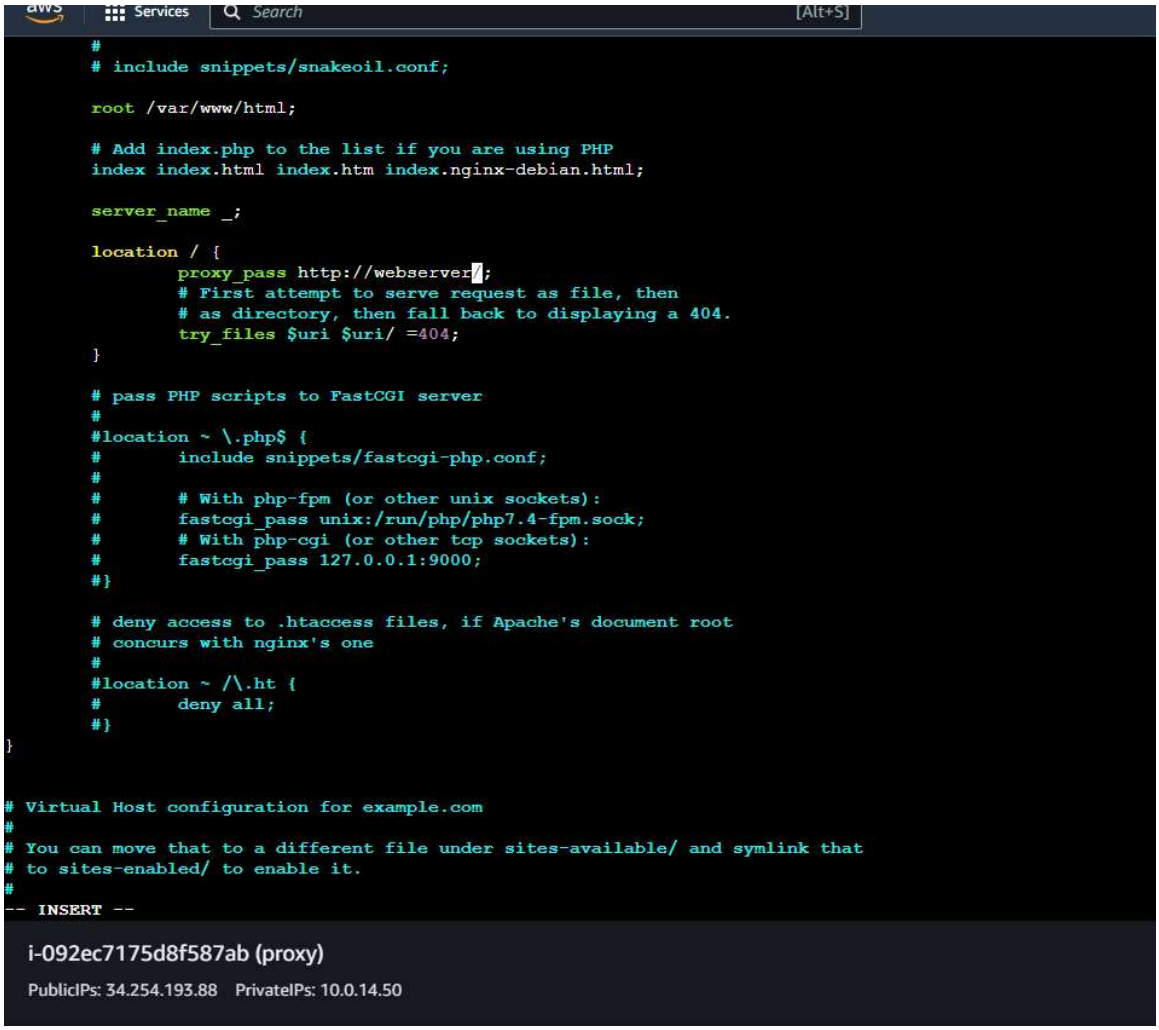
Configure in cd /etc/nginx/sites-available/default-------change their


https://wiki.debian.org/Nginx/DirectoryStructure
#
In most cases, administrators will remove this file from sites-enabled/ and
leave it as reference inside of sites-available where it will continue to be
updated by the nginx packaging team.
#
This file will automatically load configuration files provided by other
applications, such as Drupal or Wordpress. These applications will be made
available underneath a path with that package name, such as /drupal8.
#
Please see /usr/share/doc/nginx-doc/examples/ for more detailed examples.
##

Default server configuration
#

upstream webserver
{
server 10.0.130.166:8080
server 10.0.135.183
}
server {
listen 80 default_server;
listen [::]:80 default_server;

SSL configuration
#
listen 443 ssl default_server;
listen [::]:443 ssl default_server;
#
Note: You should disable gzip for SSL traffic.
See: https://bugs.debian.org/773332
#
Read up on ssl_ciphers to ensure a secure configuration.
See: https://bugs.debian.org/765782
#
Self signed certs generated by the ssl-cert package
-- INSERT --


Services
Search
[Alt+S]
#
include snippets/snakeoil.conf;
root /var/www/html;
Add index.php to the list if you are using PHP
index index.html index.htm index.nginx-debian.html;
server_name _;
location / {
proxy_pass http://webserver/;
# First attempt to serve request as file, then
# as directory, then fall back to displaying a 404.
try_files $uri $uri/ =404;
}
# pass PHP scripts to FastCGI server
#
#location ~ \.php$ {
include snippets/fastcgi-php.conf;
#
With php-fpm (or other unix sockets):
fastcgi_pass unix:/run/php/php7.4-fpm.sock;
With php-cgi (or other tcp sockets):
fastcgi_pass 127.0.0.1:9000;
#}
deny access to .htaccess files, if Apache's document root
concurs with nginx's one
#
#location ~ /\.ht {
deny all;
#}
}
# Virtual Host configuration for example.com
#
You can move that to a different file under sites-available/ and symlink that
to sites-enabled/ to enable it.
#
-- INSERT --
i-092ec7175d8f587ab (proxy)
PublicIPs: 34.254.193.88 PrivateIPs: 10.0.14.50