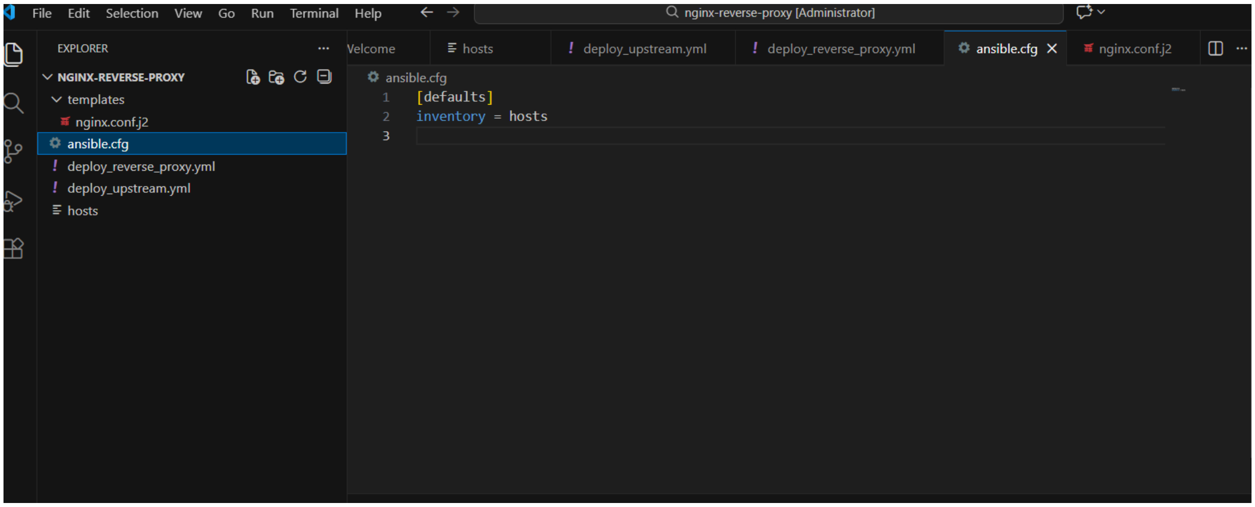
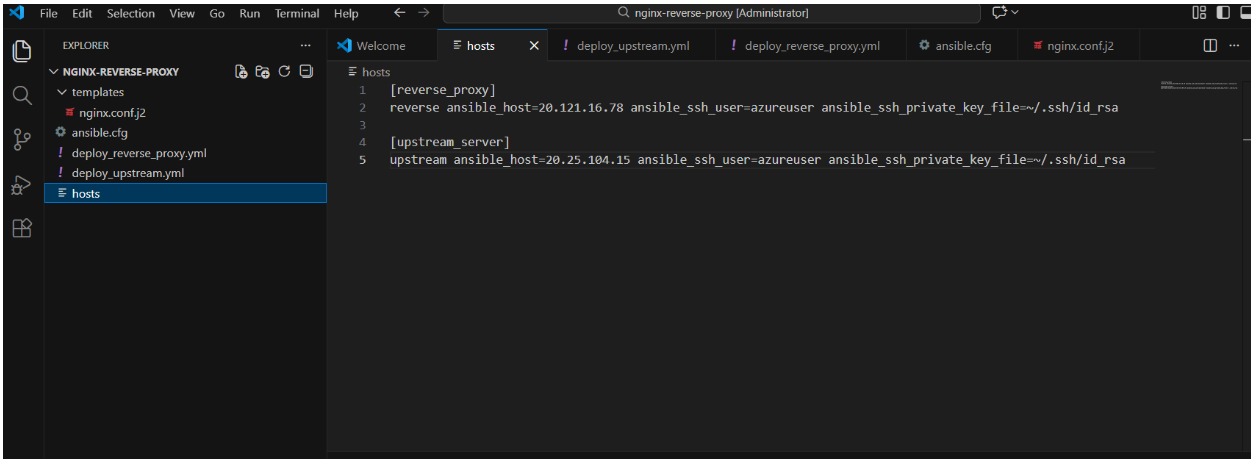
LAB Nginx Reverse Proxy using Jinj2

Configured **Reverse proxy and upstream server**.



ansible.cfg

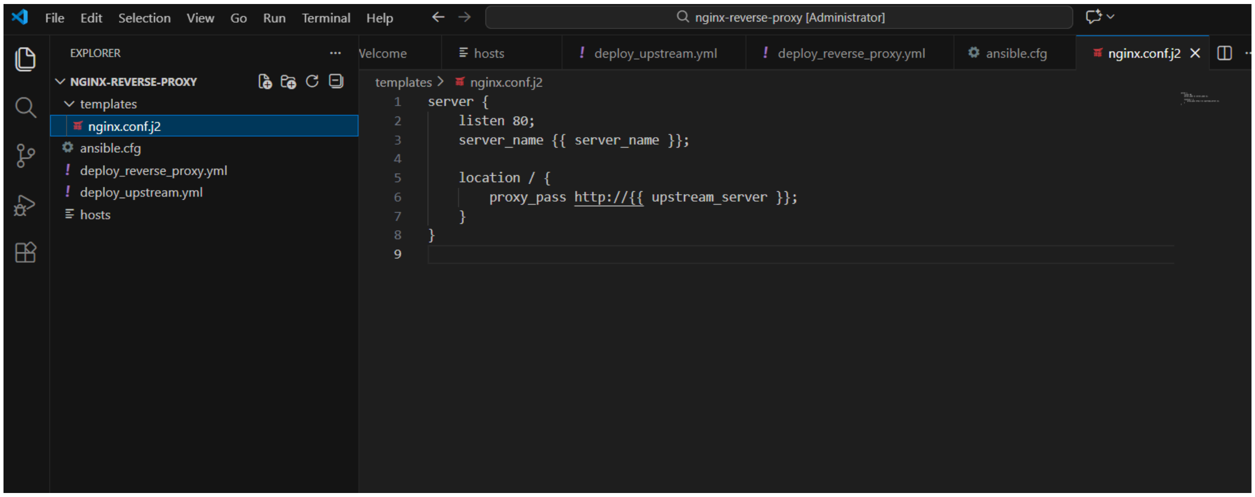
```
[defaults]
inventory = hosts

```



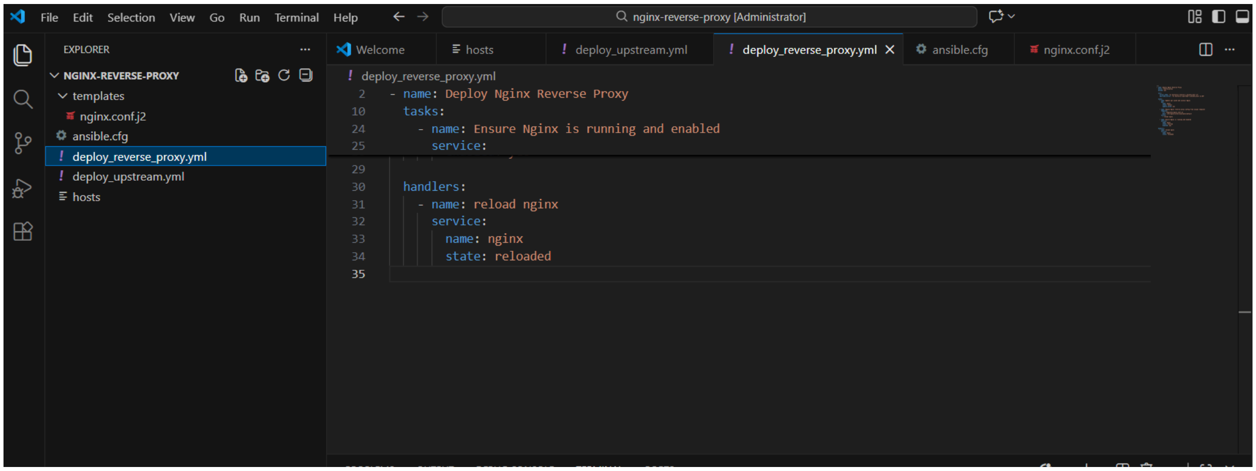
hosts

```
[reverse_proxy]
reverse ansible_host=20.121.16.78 ansible_ssh_user=azureuser ansible_ssh_private_key_file=~/.ssh/id_rsa

[upstream_server]
upstream ansible_host=20.25.104.15 ansible_ssh_user=azureuser ansible_ssh_private_key_file=~/.ssh/id_rsa
```

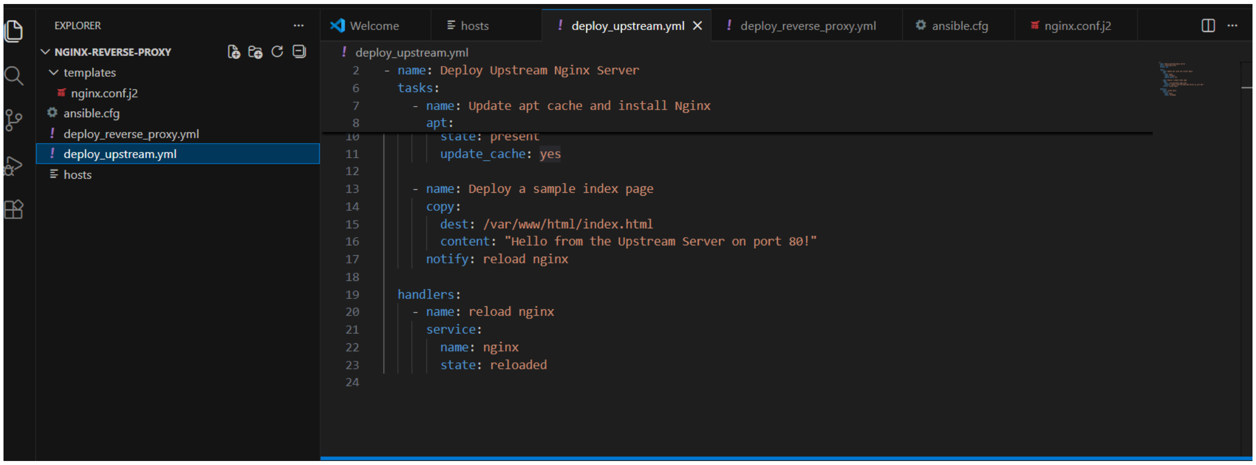
Used a **reverse proxy server** with a Jinja2 templated `nginx.conf` to forward traffic.



```
server {
    listen 80;
    server_name {{ server_name }};

    location / {
        proxy_pass http://{{ upstream_server }};
    }
}

```

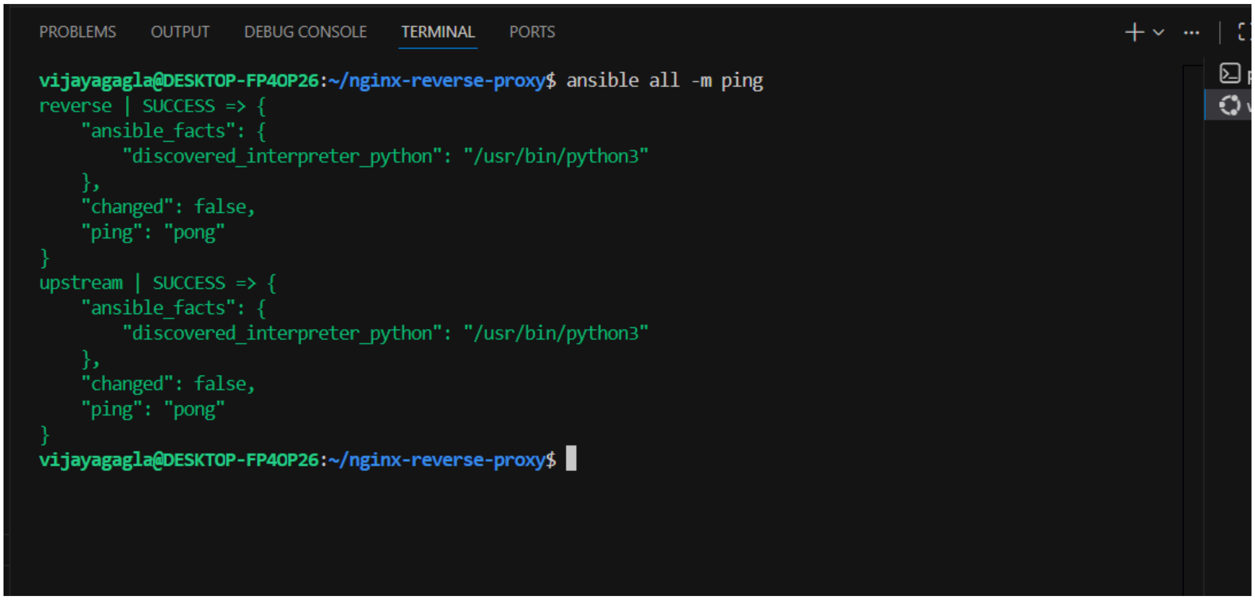


```
- name: Deploy Nginx Reverse Proxy
  tasks:
    - name: Ensure Nginx is running and enabled
      service:

  handlers:
    - name: reload nginx
      service:
        name: nginx
        state: reloaded

```

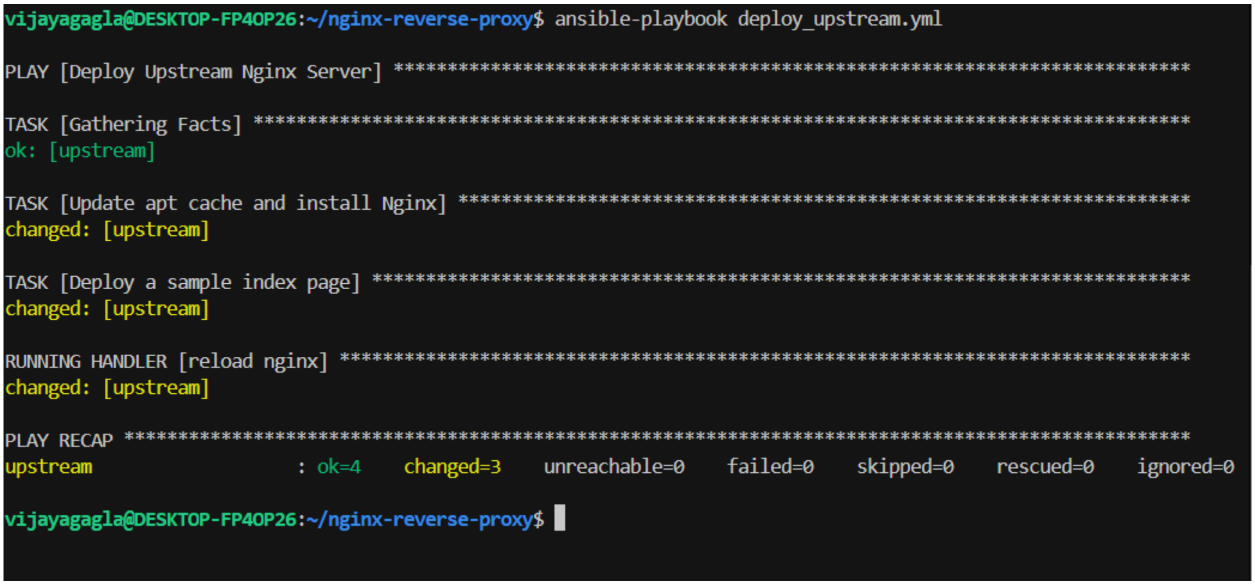


```yaml
- name: Deploy Upstream Nginx Server
  tasks:
    - name: Update apt cache and install Nginx
      apt:
        state: present
        update_cache: yes

    - name: Deploy a sample index page
      copy:
        dest: /var/www/html/index.html
        content: "Hello from the Upstream Server on port 80!"
      notify: reload nginx

  handlers:
    - name: reload nginx
      service:
        name: nginx
        state: reloaded
```

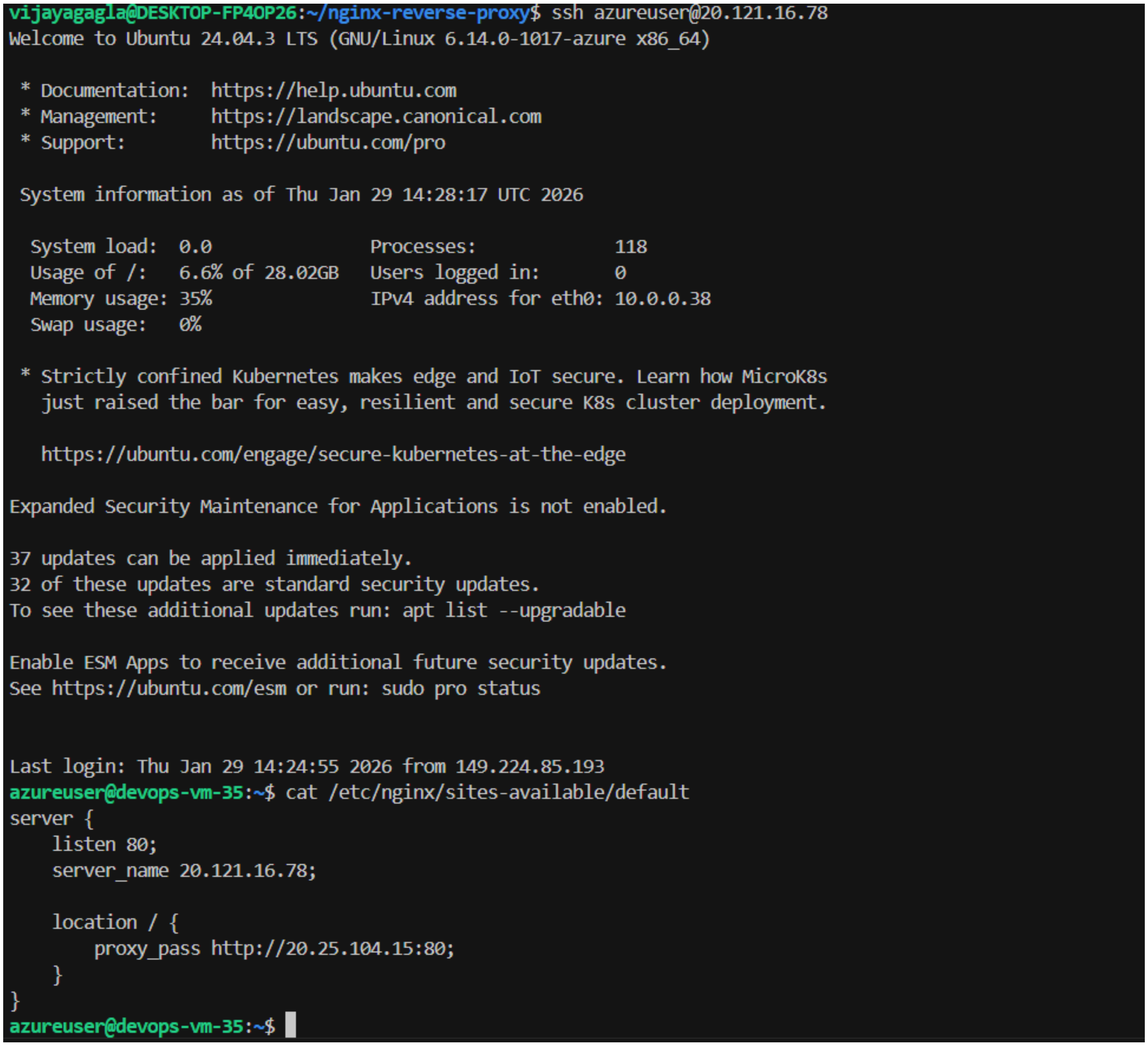
Outputs



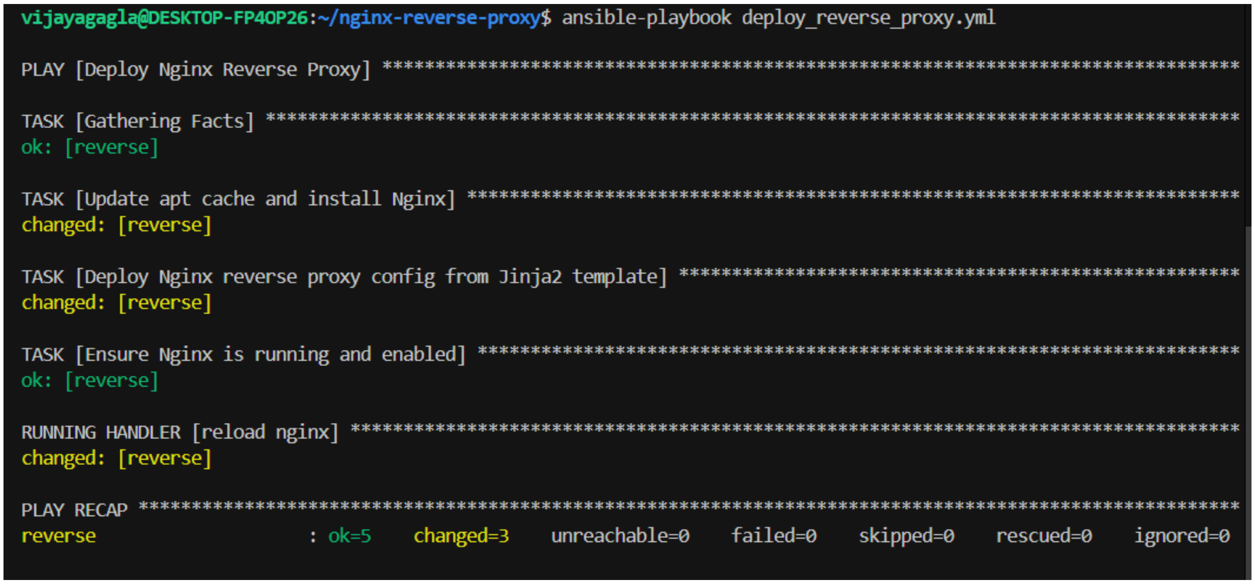
```
vijayagagla@DESKTOP-FP4OP26:~/nginx-reverse-proxy$ ansible all -m ping
reverse | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
upstream | SUCCESS => {
    "ansible_facts": {
        "discovered_interpreter_python": "/usr/bin/python3"
    },
    "changed": false,
    "ping": "pong"
}
vijayagagla@DESKTOP-FP4OP26:~/nginx-reverse-proxy$ 
```



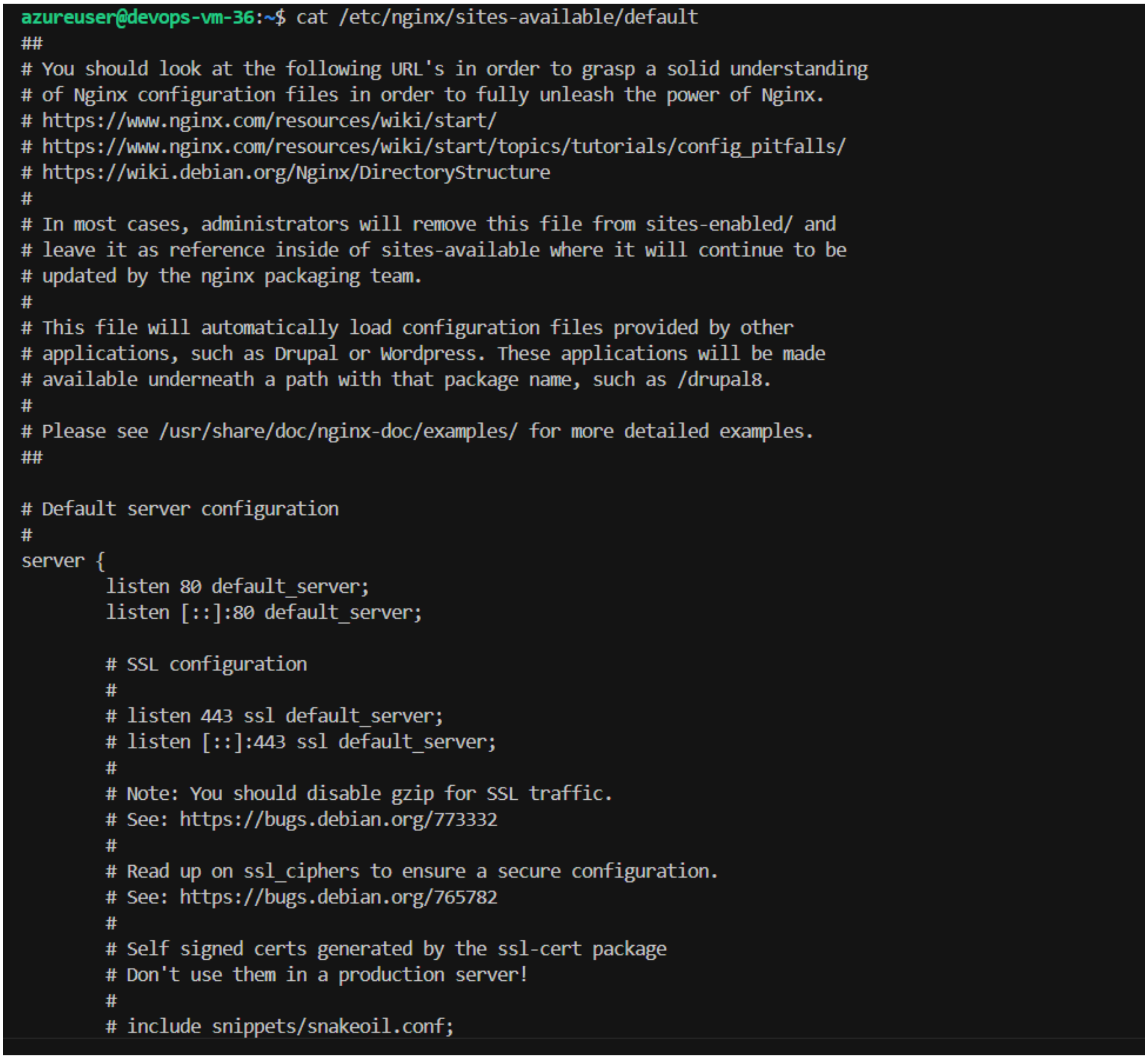
```
vijayagagla@DESKTOP-FP4OP26:~/nginx-reverse-proxy$ ansible-playbook deploy_upstream.yml

PLAY [Deploy Upstream Nginx Server] **************************************************************************

TASK [Gathering Facts] *********************************************************************************************
ok: [upstream]

TASK [Update apt cache and install Nginx] ***************************************************************
changed: [upstream]

TASK [Deploy a sample index page] **************************************************************************
changed: [upstream]

RUNNING HANDLER [reload nginx] ******************************************************************************
changed: [upstream]

PLAY RECAP ************************************************************************************************************
upstream                   : ok=4    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0

vijayagagla@DESKTOP-FP4OP26:~/nginx-reverse-proxy$ 
```



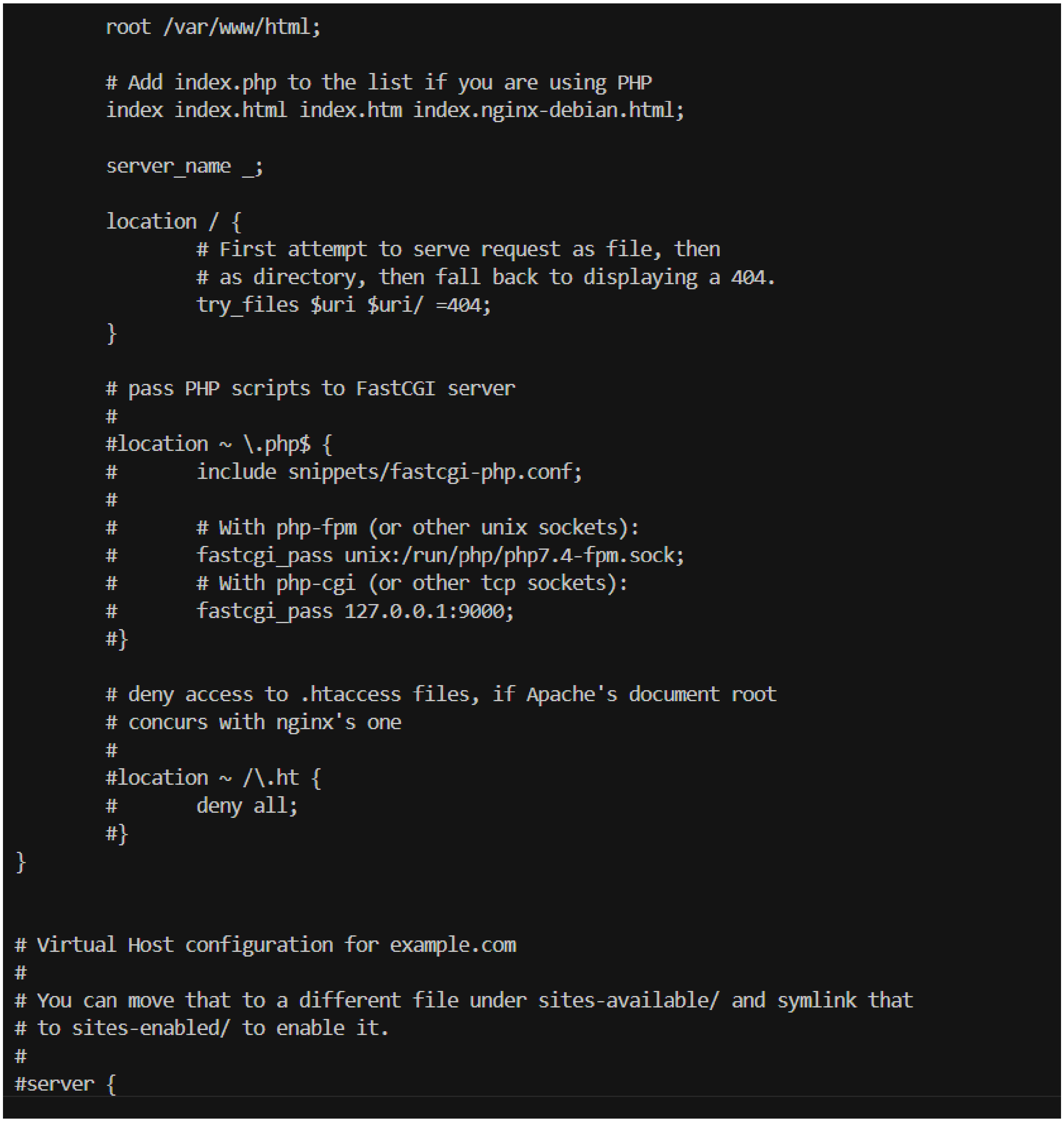
```
vijayagagla@DESKTOP-FP4OP26:~/nginx-reverse-proxy$ ssh azureuser@20.121.16.78
Welcome to Ubuntu 24.04.3 LTS (GNU/Linux 6.14.0-1017-azure x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/pro

 System information as of Thu Jan 29 14:28:17 UTC 2026

  System load:  0.0               Processes:             118
  Usage of /:   6.6% of 28.02GB   Users logged in:       0
  Memory usage: 35%               IPv4 address for eth0: 10.0.0.38
  Swap usage:   0%

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

Expanded Security Maintenance for Applications is not enabled.

37 updates can be applied immediately.
32 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status


Last login: Thu Jan 29 14:24:55 2026 from 149.224.85.193
azureuser@devops-vm-35:~$ cat /etc/nginx/sites-available/default
server {
    listen 80;
    server_name 20.121.16.78;

    location / {
        proxy_pass http://20.25.104.15:80;
    }
}
azureuser@devops-vm-35:~$ 
```



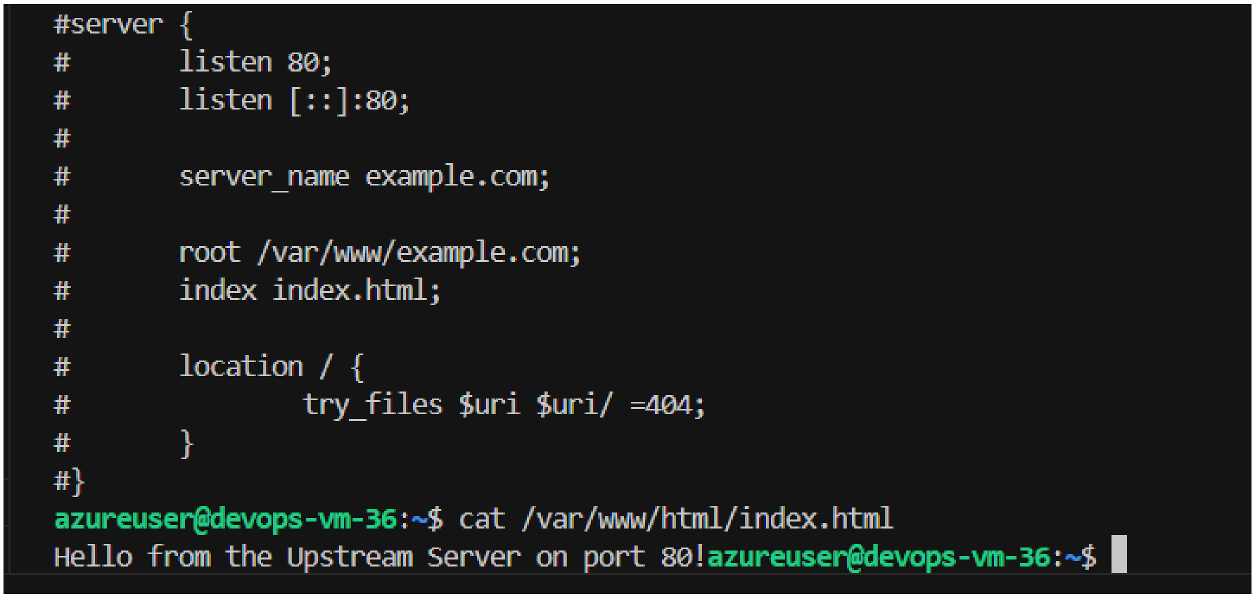
```
vijayagagla@DESKTOP-FP4OP26:~/nginx-reverse-proxy$ ansible-playbook deploy_reverse_proxy.yml

PLAY [Deploy Nginx Reverse Proxy] ****************************************************************************

TASK [Gathering Facts] ***************************************************************************************
ok: [reverse]

TASK [Update apt cache and install Nginx] ********************************************************************
changed: [reverse]

TASK [Deploy Nginx reverse proxy config from Jinja2 template] ************************************************
changed: [reverse]

TASK [Ensure Nginx is running and enabled] *******************************************************************
ok: [reverse]

RUNNING HANDLER [reload nginx] *******************************************************************************
changed: [reverse]

PLAY RECAP ***************************************************************************************************
reverse                    : ok=5    changed=3    unreachable=0    failed=0    skipped=0    rescued=0    ignored=0
```



```
azureuser@devops-vm-36:~$ cat /etc/nginx/sites-available/default
##
# You should look at the following URL's in order to grasp a solid understanding
# of Nginx configuration files in order to fully unleash the power of Nginx.
# https://www.nginx.com/resources/wiki/start/
# https://www.nginx.com/resources/wiki/start/topics/tutorials/config_pitfalls/
# 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
#
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
        # Don't use them in a production server!
        #
        # include snippets/snakeoil.conf;
```

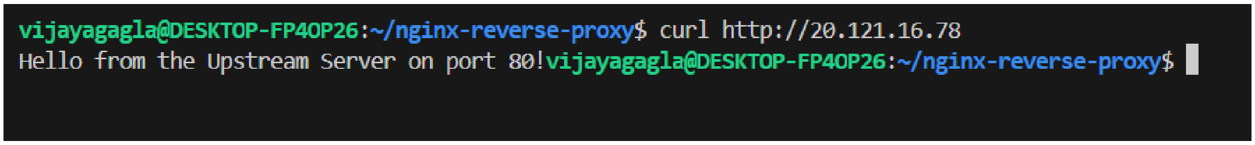


```
        root /var/www/html;

        # Add index.php to the list if you are using PHP
        index index.html index.htm index.nginx-debian.html;

        server_name _;

        location / {
                # 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.
#
#server {
```



```
#server {
#       listen 80;
#       listen [::]:80;
#
#       server_name example.com;
#
#       root /var/www/example.com;
#       index index.html;
#
#       location / {
#               try_files $uri $uri/ =404;
#       }
#}
azureuser@devops-vm-36:~$ cat /var/www/html/index.html
Hello from the Upstream Server on port 80!azureuser@devops-vm-36:~$
```

Verified successful request routing.



```
vijayagagla@DESKTOP-FP4OP26:~/nginx-reverse-proxy$ curl http://20.121.16.78
Hello from the Upstream Server on port 80!vijayagagla@DESKTOP-FP4OP26:~/nginx-reverse-proxy$
```