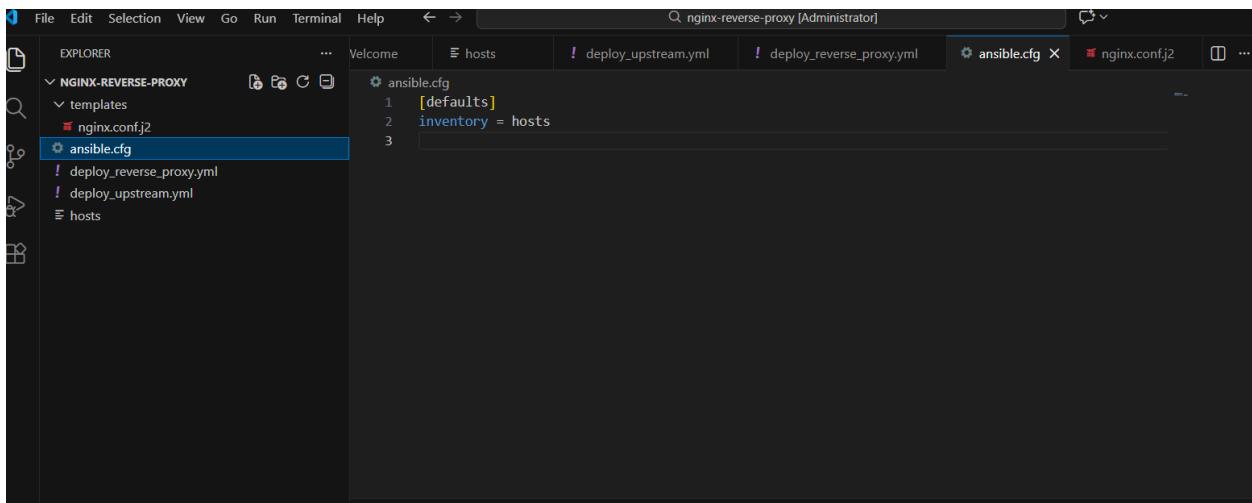


LAB Nginx Reverse Proxy using Jinj2

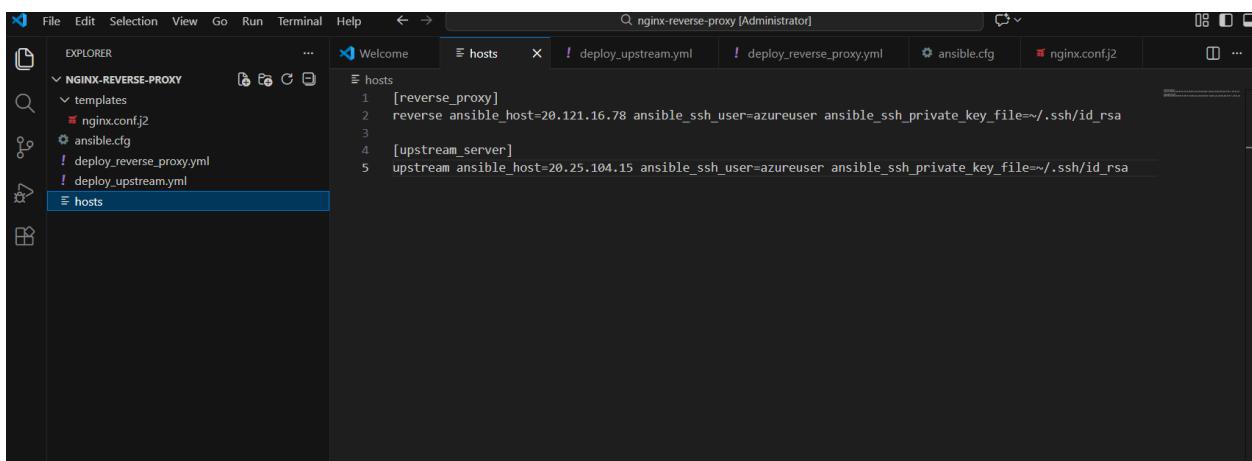
Configured **Reverse proxy and upstream server.**



The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows a folder named "NGINX-REVERSE-PROXY" containing "templates" (with "nginx.conf.j2"), "ansible.cfg", "deploy_reverse_proxy.yml", and "deploy_upstream.yml".
- Terminal:** Shows the command "nginx-reverse-proxy [Administrator]".
- Code Editor:** Displays the content of "ansible.cfg".

```
[defaults]
inventory = hosts
```



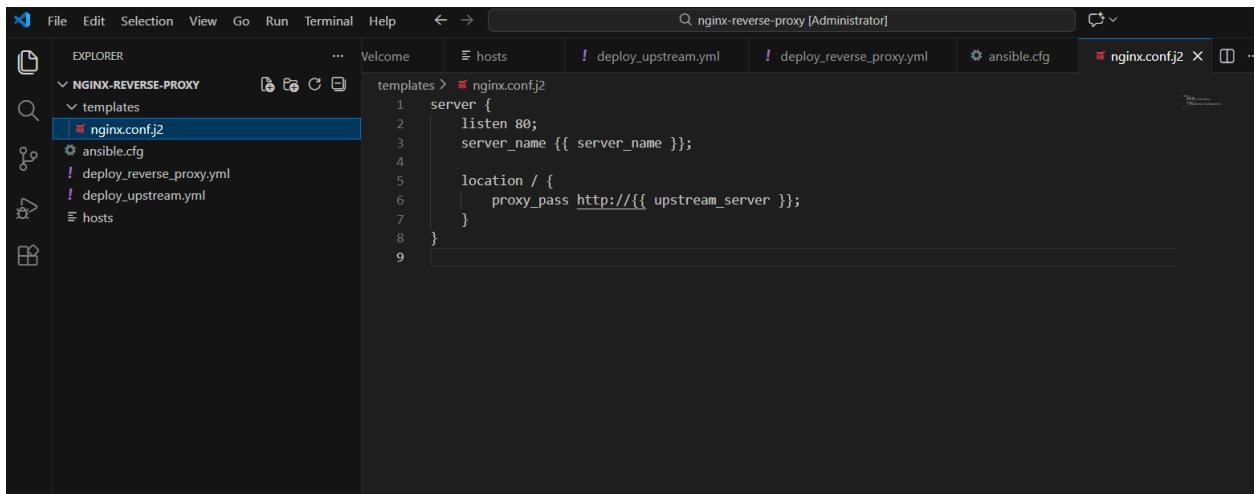
The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows a folder named "NGINX-REVERSE-PROXY" containing "templates" (with "nginx.conf.j2"), "ansible.cfg", "deploy_reverse_proxy.yml", and "deploy_upstream.yml".
- Terminal:** Shows the command "nginx-reverse-proxy [Administrator]".
- Code Editor:** Displays the content of "hosts".

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

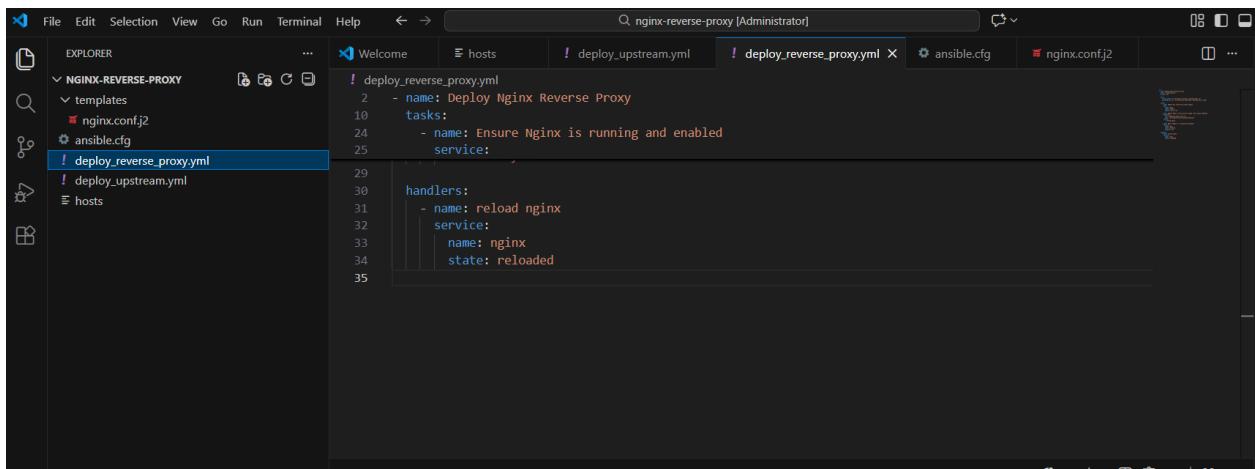
[upstream_server]
upstream ansible_host=20.25.104.15 ansible_ssh_user=azureuser ansible_ssh_private_key_file=/ssh/id_rsa
```

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



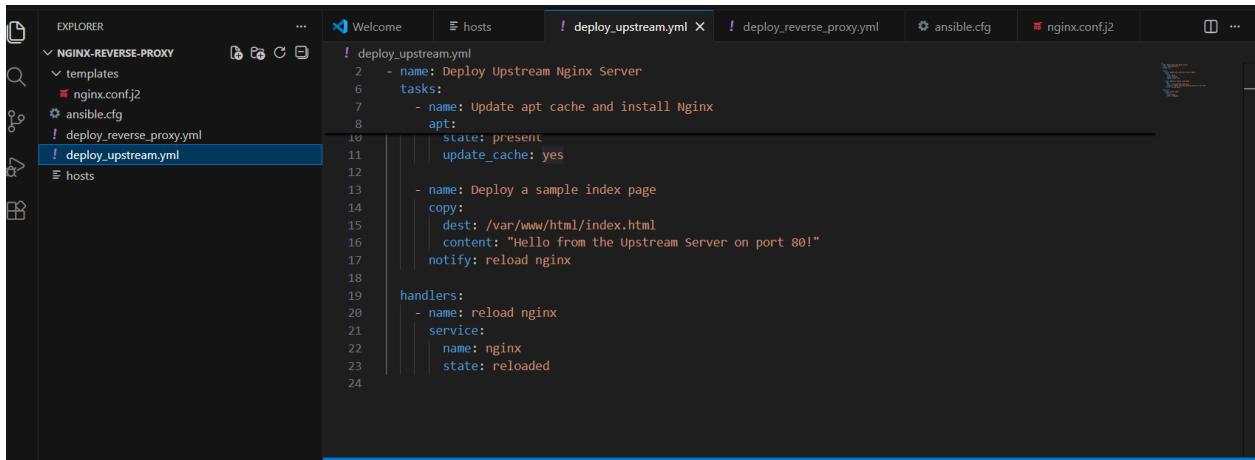
A screenshot of a code editor window titled "nginx-reverse-proxy [Administrator]". The left sidebar shows a project structure under "EXPLORER" with "NGINX-REVERSE-PROXY" expanded, containing "templates", "ansible.cfg", and "hosts". The "templates" folder contains "nginx.conf.j2", "deploy_reverse_proxy.yml", and "deploy_upstream.yml". The main editor area displays the content of "nginx.conf.j2":

```
templates > nginx.conf.j2
1 server {
2     listen 80;
3     server_name {{ server_name }};
4
5     location / {
6         proxy_pass http://{{ upstream_server }};
7     }
8 }
```



A screenshot of a code editor window titled "nginx-reverse-proxy [Administrator]". The left sidebar shows a project structure under "EXPLORER" with "NGINX-REVERSE-PROXY" expanded, containing "templates", "ansible.cfg", and "hosts". The "templates" folder contains "nginx.conf.j2", "deploy_reverse_proxy.yml", and "deploy_upstream.yml". The main editor area displays the content of "deploy_reverse_proxy.yml":

```
! deploy_reverse_proxy.yml
2 - name: Deploy Nginx Reverse Proxy
10   tasks:
24     - name: Ensure Nginx is running and enabled
25       service:
29
30       handlers:
31         - name: reload nginx
32           service:
33             name: nginx
34             state: reloaded
35
```

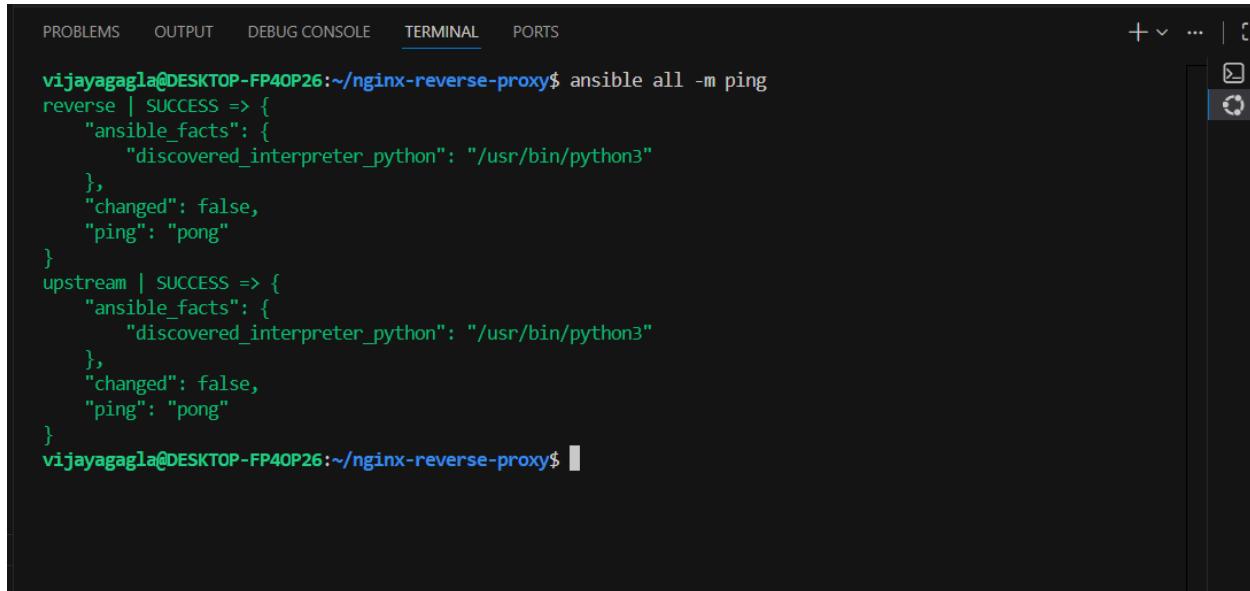


The screenshot shows the Visual Studio Code interface with the following details:

- EXPLORER** sidebar: Shows a project structure for "NGINX-REVERSE-PROXY". It includes a "templates" folder containing "nginx.conf.j2", an "ansible.cfg" file, and two Ansible playbooks: "deploy_reverse_proxy.yml" and "deploy_upstream.yml". The "deploy_upstream.yml" file is currently selected.
- CODE** tab bar: Includes tabs for "Welcome", "hosts", "deploy_upstream.yml" (which is active), "deploy_reverse_proxy.yml", "ansible.cfg", "nginx.conf.j2", and "...".
- Content Area**: Displays the YAML code for the "deploy_upstream.yml" playbook. The code defines a task to update apt cache and install Nginx, copy an index.html file, and reload the Nginx service.

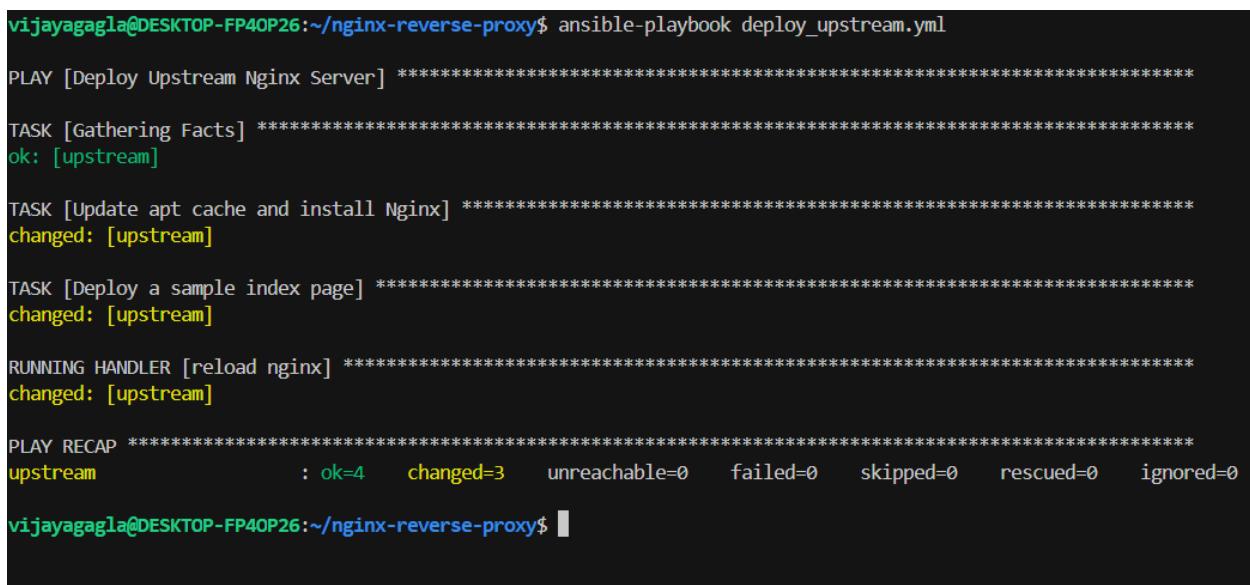
```
deploy_upstream.yml
1 deploy: &name: Deploy Upstream Nginx Server
2   tasks:
3     - name: Update apt cache and install Nginx
4       apt:
5         state: present
6         update_cache: yes
7
8     - name: Deploy a sample index page
9       copy:
10      dest: /var/www/html/index.html
11      content: "Hello from the Upstream Server on port 80!"
12      notify: reload nginx
13
14 handlers:
15   - name: reload nginx
16     service:
17       name: nginx
18       state: reloaded
19
20
```

Outputs



PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS + ⌂ ⌄ ⌁

```
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$
```