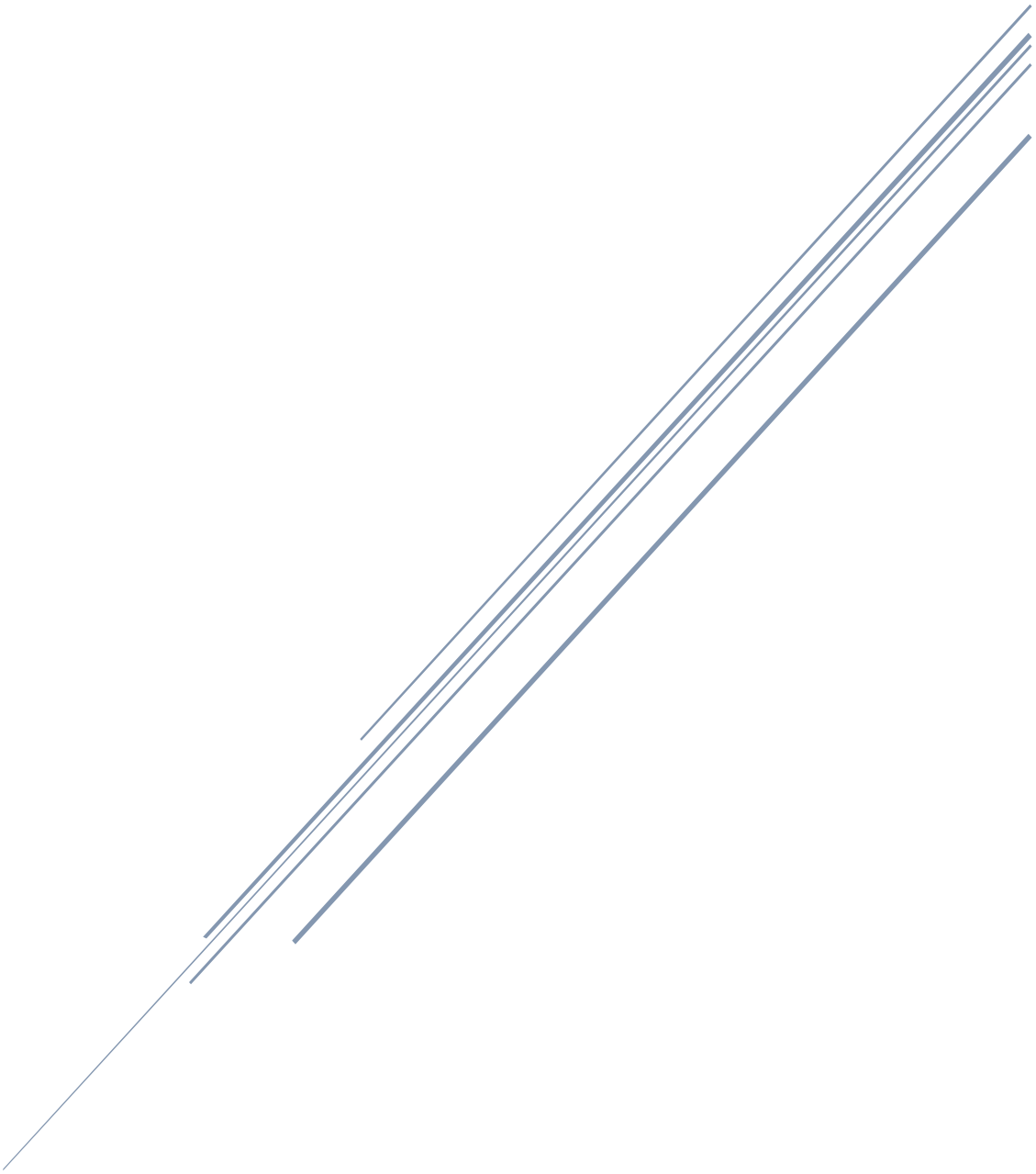
TEST DE SOLUTIONS VPN



ALGO VPN

1. INSTALLATION

a) Passer en root

"sudo su"

b) Mettre à jour le système

"apt update && apt upgrade -y"

c) Installer les dépendances nécessaires

"apt install python3 python3-pip git -y"

d) Installer git si besoin

"apt install git"

e) Cloner le dépôt

"git clone https://github.com/trailofbits/algo"

f) Accéder au répertoire Algo

"cd algo"

g) Installé et configurer l'environnement virtuel

"python3 -m pip install --upgrade pip virtualenv"

```
root@ubuntu-virtual-machine:/home/kaly/algo# python3 -m pip install --upgrade pip virtualenv
Requirement already satisfied: pip in /usr/lib/python3/dist-packages (22.0.2)
Collecting pip
  Downloading pip-24.2-py3-none-any.whl (1.8 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.8/1.8 MB 8.3 MB/s eta 0:00:00
Collecting virtualenv
  Downloading virtualenv-20.26.3-py3-none-any.whl (5.7 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 5.7/5.7 MB 8.4 MB/s eta 0:00:00
Collecting filelock<4,>=3.12.2
  Downloading filelock-3.15.4-py3-none-any.whl (16 kB)
Collecting platformdirs<5,>=3.9.1
  Downloading platformdirs-4.2.2-py3-none-any.whl (18 kB)
Collecting distlib<1,>=0.3.7
  Downloading distlib-0.3.8-py2.py3-none-any.whl (468 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 468.9/468.9 KB 7.6 MB/s eta 0:00:00
Installing collected packages: distlib, platformdirs, pip, filelock, virtualenv
  Attempting uninstall: pip
    Found existing installation: pip 22.0.2
    Not uninstalling pip at /usr/lib/python3/dist-packages, outside environment /usr
    Can't uninstall 'pip'. No files were found to uninstall.
Successfully installed distlib-0.3.8 filelock-3.15.4 pip-24.2 platformdirs-4.2.2 virtualenv-20.26.3
```

“python3 -m virtualenv --python="$(command -v python3)" .env”

```
root@ubuntu-virtual-machine:/home/kaly/algo# python3 -m virtualenv --python="$(command -v python3)" .env
created virtual environment CPython3.10.12.final.0-64 in 447ms
  creator CPython3Posix(dest=/home/kaly/algo/.env, clear=False, no_vcs_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy, app_data_dir=/root/.local/share/virtualenv)
    added seed packages: pip==24.1, setuptools==70.1.0, wheel==0.43.0
  activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator
```

“source .env/bin/activate”

h) Installé les dépendances

“python3 -m pip install -r requirements.txt”

```
(.env) root@ubuntu-virtual-machine:/home/kaly/algo# python3 -m pip install -r requirements.txt
Collecting ansible==9.1.0 (from -r requirements.txt (line 1))
  Downloading ansible-9.1.0-py3-none-any.whl.metadata (7.9 kB)
Collecting jinja2~=3.0.3 (from -r requirements.txt (line 2))
  Downloading Jinja2-3.0.3-py3-none-any.whl.metadata (3.5 kB)
Collecting netaddr (from -r requirements.txt (line 3))
  Downloading netaddr-1.3.0-py3-none-any.whl.metadata (5.0 kB)
Collecting ansible-core~=2.16.1 (from ansible==9.1.0->-r requirements.txt (line 1))
  Downloading ansible_core-2.16.9-py3-none-any.whl.metadata (6.9 kB)
Collecting MarkupSafe>=2.0 (from jinja2~=3.0.3->-r requirements.txt (line 2))
  Downloading MarkupSafe-2.1.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (3.0 kB)
Collecting PyYAML>=5.1 (from ansible-core~=2.16.1->ansible==9.1.0->-r requirements.txt (line 1))
  Downloading PyYAML-6.0.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (2.1 kB)
Collecting cryptography (from ansible-core~=2.16.1->ansible==9.1.0->-r requirements.txt (line 1))
  Downloading cryptography-43.0.0-cp39-abi3-manylinux_2_28_x86_64.whl.metadata (5.4 kB)
Collecting packaging (from ansible-core~=2.16.1->ansible==9.1.0->-r requirements.txt (line 1))
  Downloading packaging-24.1-py3-none-any.whl.metadata (3.2 kB)
Collecting resolvelib<1.1.0,>=0.5.3 (from ansible-core~=2.16.1->ansible==9.1.0->-r requirements.txt (line 1))
  Downloading resolvelib-1.0.1-py2.py3-none-any.whl.metadata (4.0 kB)
Collecting cffi>=1.12 (from cryptography->ansible-core~=2.16.1->ansible==9.1.0->-r requirements.txt (line 1))
  Downloading cffi-1.16.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (1.5 kB)
Collecting pycparser (from cffi>=1.12->cryptography->ansible-core~=2.16.1->ansible==9.1.0->-r requirements.txt (line 1))
  Downloading pycparser-2.22-py3-none-any.whl.metadata (943 bytes)
Downloading ansible-9.1.0-py3-none-any.whl (48.1 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 48.1/48.1 MB 7.7 MB/s eta 0:00:00
Downloading Jinja2-3.0.3-py3-none-any.whl (133 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 133.6/133.6 kB 8.4 MB/s eta 0:00:00
Downloading netaddr-1.3.0-py3-none-any.whl (2.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.3/2.3 MB 715.5 kB/s eta 0:00:00
Downloading ansible_core-2.16.9-py3-none-any.whl (2.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 2.3/2.3 MB 1.6 MB/s eta 0:00:00
Downloading MarkupSafe-2.1.5-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (25 kB)
Downloading PyYAML-6.0.1-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (705 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 705.5/705.5 kB 8.7 MB/s eta 0:00:00
Downloading resolvelib-1.0.1-py2.py3-none-any.whl (17 kB)
Downloading cryptography-43.0.0-cp39-abi3-manylinux_2_28_x86_64.whl (4.0 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 4.0/4.0 MB 8.8 MB/s eta 0:00:00
Downloading packaging-24.1-py3-none-any.whl (53 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 54.0/54.0 kB 5.3 MB/s eta 0:00:00
Downloading cffi-1.16.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (443 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 443.9/443.9 kB 9.7 MB/s eta 0:00:00
Downloading pycparser-2.22-py3-none-any.whl (117 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 117.6/117.6 kB 10.6 MB/s eta 0:00:00
Installing collected packages: resolvelib, PyYAML, pycparser, packaging, netaddr, MarkupSafe, jinja2, cffi, cryptography, ansible-core, ansible
Successfully installed MarkupSafe-2.1.5 PyYAML-6.0.1 ansible-9.1.0 ansible-core-2.16.9 cffi-1.16.0 cryptography-43.0.0 jinja2-3.0.3 netaddr-1.3.0 packaging-24.1 pycparser-2.22 resolvelib-1.0.1
```

i) Lancé le script d’installation Algo

“./algo”

```
(.env) root@ubuntu-virtual-machine:/home/kaly/algo# ./algo

PLAY [localhost] ***************************************************************************************************************************************************************

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

TASK [Playbook dir stat] *******************************************************************************************************************************************************
ok: [localhost]

TASK [Ensure Ansible is not being run in a world writable directory] ************************************************************************************************************
ok: [localhost] => {
    "changed": false,
    "msg": "All assertions passed"
}
[DEPRECATION WARNING]: Use 'ansible.utils.ipaddr' module instead. This feature will be removed from ansible.netcommon in a release after 2024-01-01. Deprecation warnings can be disabled by setting deprecation_warnings=False in ansible.cfg.
[WARNING]: The value '' is not a valid IP address or network, passing this value to ipaddr filter might result in breaking change in future.

TASK [Ensure the requirements installed] ***************************************************************************************************************************************
ok: [localhost]

TASK [Set required ansible version as a fact] **********************************************************************************************************************************
ok: [localhost] => (item=ansible==9.1.0)

TASK [Just get the list from default pip] **************************************************************************************************************************************
ok: [localhost]

TASK [Verify Python meets Algo VPN requirements] *******************************************************************************************************************************
ok: [localhost] => {
    "changed": false,
    "msg": "All assertions passed"
}

TASK [Verify Ansible meets Algo VPN requirements] ******************************************************************************************************************************
ok: [localhost] => {
    "changed": false,
    "msg": "All assertions passed"
}
[WARNING]: Found variable using reserved name: no_log

PLAY [Ask user for the input] **************************************************************************************************************************************************

TASK [Gathering Facts] *********************************************************************************************************************************************************
ok: [localhost]
[Cloud prompt]
```

j) Vérifiez si Open-SSH est installé sur votre système

"dpkg -s openssh-server"

k) Si SSH n'est pas installé, installez-le

" apt update

apt install openssh-server"

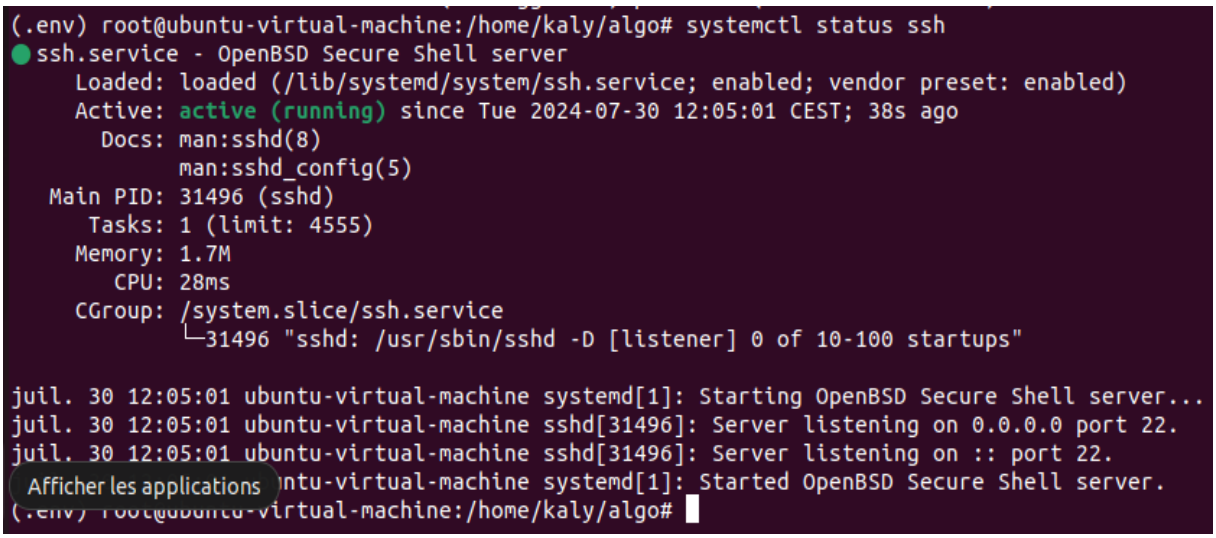
l) Configurer les ports & les autorisation

"sudo nano /etc/ssh/sshd_config"

- Port 22
- Port 2222
- PermitRootLogin yes
- PubkeyAuthentication yes
- PasswordAuthentication no

m) Assurez-vous que le service est en cours

"systemctl status ssh"



n) Restart SHH

" sudo systemctl restart ssh"

o) Installez Ansible

“sudo apt install ansible”

p) **Installez Curl & Noter l’IP**

“sudo apt install curl”

```
kaly@ubuntu-virtual-machine:~$ curl ifconfig.me
176.165.51.97kaly@ubuntu-virtual-machine:~$ ssh root@176.165.51.97
```

2. CONFIGURATION

a) Installer sur un serveur Ubuntu existant

“12”

```
[Cloud prompt]
What provider would you like to use?
    1. DigitalOcean
    2. Amazon Lightsail
    3. Amazon EC2
    4. Microsoft Azure
    5. Google Compute Engine
    6. Hetzner Cloud
    7. Vultr
    8. Scaleway
    9. OpenStack (DreamCompute optimised)
    10. CloudStack (Exoscale optimised)
    11. Linode
    12. Install to existing Ubuntu latest LTS server (for more advanced users)

Enter the number of your desired provider
:
12
```

b) Sélectionnez ‘Y’ pour les 2 prochaines questions

```
Do you want macOS/iOS clients to enable "Connect On Demand" when connected to cellular networks?
[y/N]
:

TASK [Cellular On Demand prompt] ***********************************************************
ok: [localhost]
[Wi-Fi On Demand prompt]
Do you want macOS/iOS clients to enable "Connect On Demand" when connected to Wi-Fi?
[y/N]
:
```

c) Sélectionnez

“HomeNet”

```
TASK [Wi-Fi On Demand prompt] ***************************************************************
ok: [localhost]
[Trusted Wi-Fi networks prompt]
List the names of any trusted Wi-Fi networks where macOS/iOS clients should not use "Connect On Demand"
(e.g., your home network. Comma-separated value, e.g., HomeNet,OfficeWifi,AlgoWiFi)
:
```

d) Sélectionnez ‘Y’ pour les 3 prochaines questions

```
ok: [localhost]
[Retain the PKI prompt]
Do you want to retain the keys (PKI)? (required to add users in the future, but less secure)
[y/N]
:

TASK [Retain the PKI prompt] ***************************************************************
ok: [localhost]
[DNS adblocking prompt]
Do you want to enable DNS ad blocking on this VPN server?
[y/N]
:

TASK [DNS adblocking prompt] ***************************************************************
ok: [localhost]
[SSH tunneling prompt]
Do you want each user to have their own account for SSH tunneling?
[y/N]
:
```

e) Entrez l’IP récupérer avec Curl

```
Enter the public IP address or domain name of your server: (IMPORTANT! This is used to verify the certificate)
[192.168.1.20]
:
```

f) Fin

```
ok: [192.168.1.20] => {
    "msg": [
        [
            "\"#                          Congratulations!                           #\"",
            "\"#                     Your Algo server is running.                    #\"",
            "\"#    Config files and certificates are in the ./configs/ directory.   #\"",
            "\"#              Go to https://whoer.net/ after connecting             #\"",
            "\"#        and ensure that all your traffic passes through the VPN.     #\"",
            "\"#                     Local DNS resolver 172.22.212.26                  #\"",
            ""
        ],
        "    \"#        The p12 and SSH keys password for new users is 3t5BIQsbF       #\"\n",
        "    \"#        The CA key password is JT8KUE4le@xNoHt@       #\"\n",
        "    "
    ]
}
```