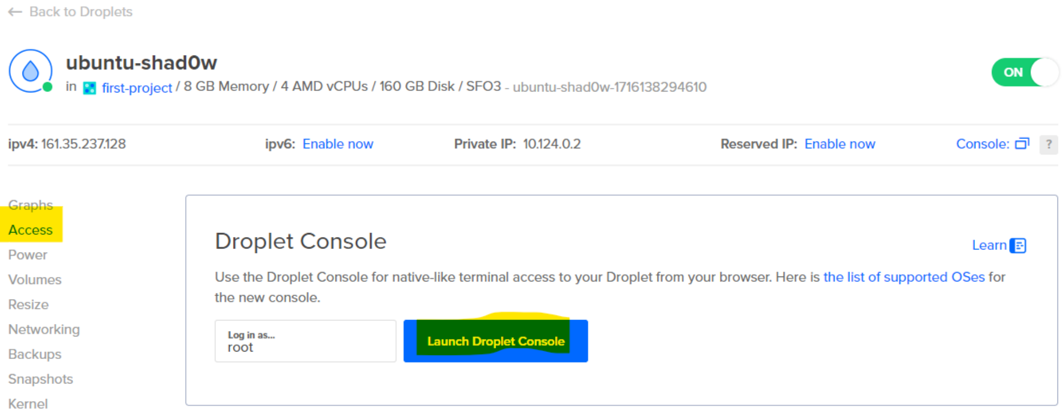
Project Shadow Sentry

A guide to deploying a cowrie honeypot integrated with the Elastic Stack

SSH into virtual machine

After signing up for a digital ocean account and picking a Ubuntu virtual machine (8 GB RAM and 4 CPUs recommended if installing both elastic cloud and honeypot on the same machine), SSH into elastic cloud either through DigitalOcean directly or your own linux machine (ensure that SSH service is up and there is no firewall blocking the connection).

- Digital Ocean method:



ubuntu-shad0w

in first-project / 8 GB Memory / 4 AMD vCPUs / 160 GB Disk / SFO3 - ubuntu-shad0w-1716138294610

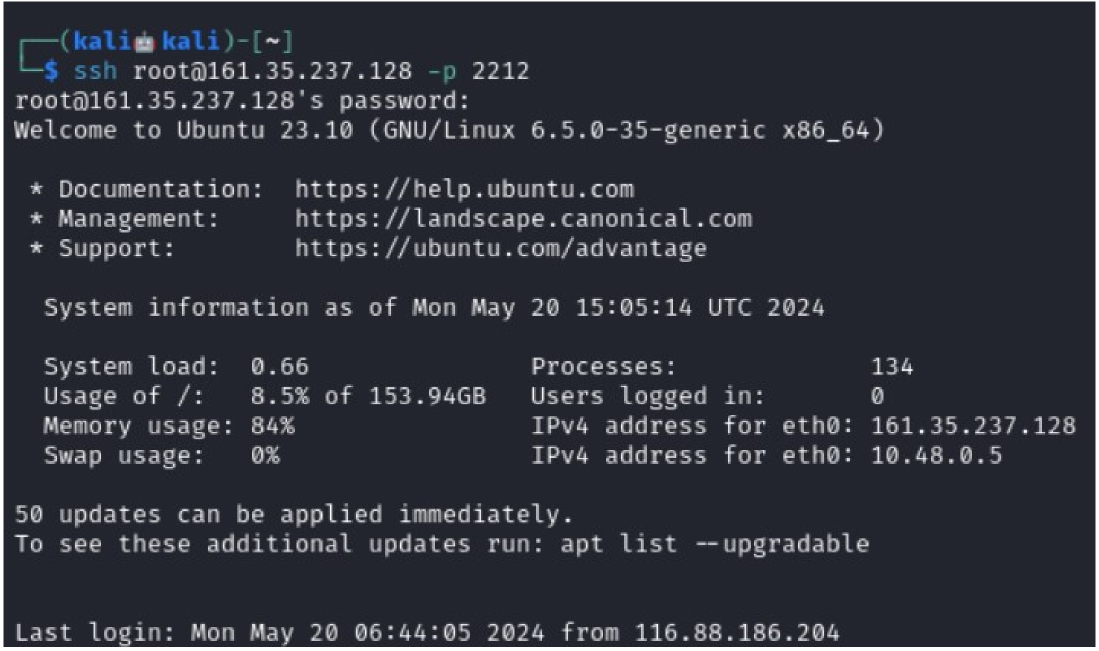
ipv4: 161.35.237.128 ipv6: Enable now Private IP: 10.124.0.2 Reserved IP: Enable now Console:

Droplet Console

Use the Droplet Console for native-like terminal access to your Droplet from your browser. Here is the list of supported OSes for the new console.

Log in as... root

- Linux machine method:
  `ssh <user>@<ip address> -p 22`



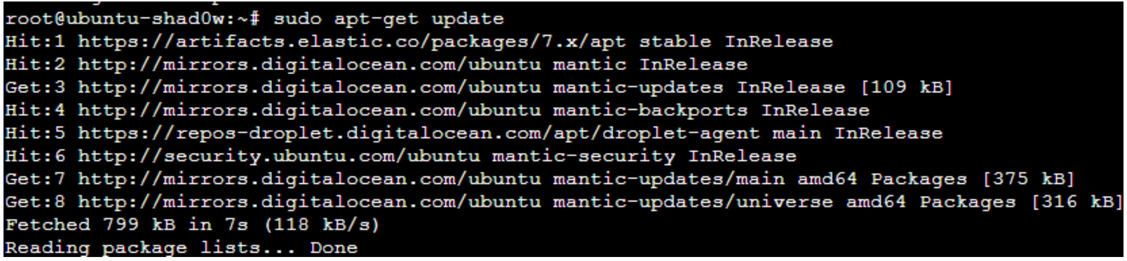
```
┌──(kali㉿kali)-[~]
└─$ ssh root@161.35.237.128 -p 2212
root@161.35.237.128's password:
Welcome to Ubuntu 23.10 (GNU/Linux 6.5.0-35-generic x86_64)

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

  System information as of Mon May 20 15:05:14 UTC 2024

  System load:  0.66               Processes:             134
  Usage of /:   8.5% of 153.94GB   Users logged in:       0
  Memory usage: 84%                IPv4 address for eth0: 161.35.237.128
  Swap usage:   0%                 IPv4 address for eth0: 10.48.0.5

50 updates can be applied immediately.
To see these additional updates run: apt list --upgradable


Last login: Mon May 20 06:44:05 2024 from 116.88.186.204
```

Upon successful login, the terminal presents basic system information.

Creating and Integrating a Cowrie Honeypot

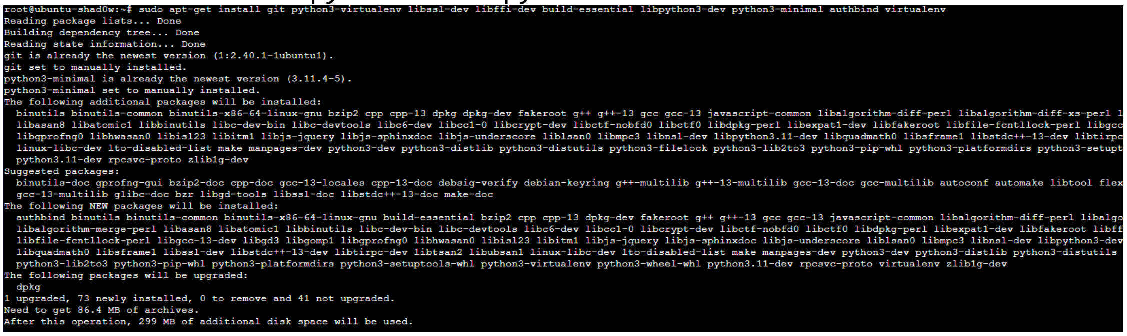
Installing dependencies

- First update your ubuntu package depositories and install system dependencies
  `sudo apt-get update`



```
root@ubuntu-shad0w:~# sudo apt-get update
Hit:1 https://artifacts.elastic.co/packages/7.x/apt stable InRelease
Hit:2 http://mirrors.digitalocean.com/ubuntu mantic InRelease
Get:3 http://mirrors.digitalocean.com/ubuntu mantic-updates InRelease [109 kB]
Hit:4 http://mirrors.digitalocean.com/ubuntu mantic-backports InRelease
Hit:5 https://repos-droplet.digitalocean.com/apt/droplet-agent main InRelease
Hit:6 http://security.ubuntu.com/ubuntu mantic-security InRelease
Get:7 http://mirrors.digitalocean.com/ubuntu mantic-updates/main amd64 Packages [375 kB]
Get:8 http://mirrors.digitalocean.com/ubuntu mantic-updates/universe amd64 Packages [316 kB]
Fetched 799 kB in 7s (118 kB/s)
Reading package lists... Done
```

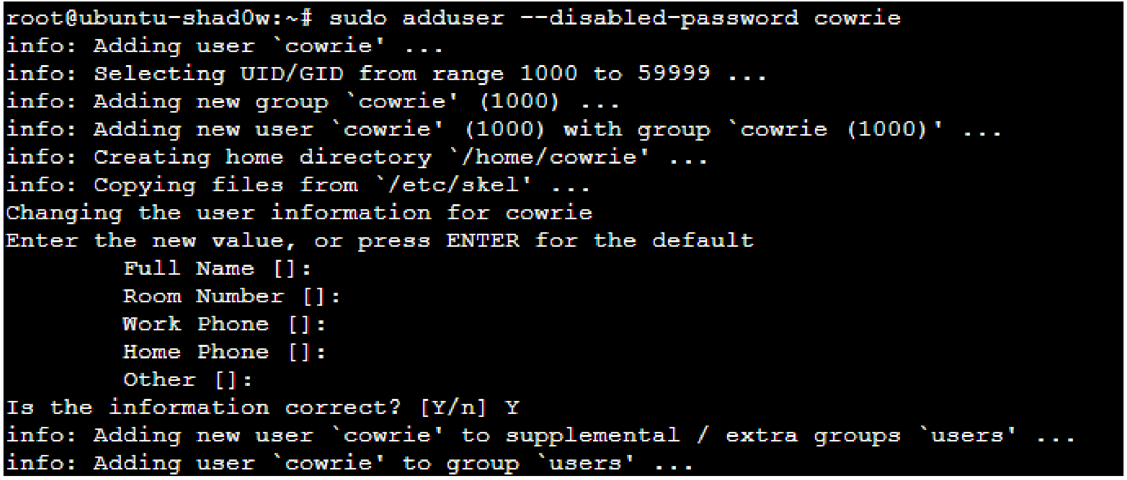
```
sudo apt-get upgrade
sudo apt-get install git python3-virtualenv libssl-dev libffi-dev
build-essential libpython3-dev python3-minimal authbind virtualenv
```



```
root@ubuntu-shad0w:~# sudo apt-get install git python3-virtualenv libssl-dev libffi-dev build-essential libpython3-dev python3-minimal authbind virtualenv
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
git is already the newest version (1:2.40.1-1ubuntu1).
git set to manually installed.
python3-minimal is already the newest version (3.11.4-5).
python3-minimal set to manually installed.
The following additional packages will be installed:
  binutils binutils-common binutils-x86-64-linux-gnu bzip2 cpp cpp-13 dpkg dpkg-dev fakeroot g++ g++-13 gcc gcc-13 javascript-common libalgorithm-diff-perl libalgorithm-diff-xs-perl l
  libasan8 libatomic1 libbinutils libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0 libdpkg-perl libexpat1-dev libfakeroot libfile-fcntllock-perl libgcc
  libgprofng0 libhwasan0 libisl23 libitm1 libjs-jquery libjs-sphinxdoc libjs-underscore liblsan0 libmpc3 libnsl-dev libpython3.11-dev libquadmath0 libsframe1 libstdc++-13-dev libtirpc
  linux-libc-dev lto-disabled-list make manpages-dev python3-dev python3-distlib python3-distutils python3-filelock python3-lib2to3 python3-pip-whl python3-platformdirs python3-setupt
  python3.11-dev rpcsvc-proto zlib1g-dev
Suggested packages:
  binutils-doc gprofng-gui bzip2-doc cpp-doc gcc-13-locales cpp-13-doc debsig-verify debian-keyring g++-multilib g++-13-multilib gcc-13-doc gcc-multilib autoconf automake libtool flex
  gcc-13-multilib glibc-doc bzr libgd-tools libssl-doc libstdc++-13-doc make-doc
The following NEW packages will be installed:
  authbind binutils binutils-common binutils-x86-64-linux-gnu build-essential bzip2 cpp cpp-13 dpkg-dev fakeroot g++ g++-13 gcc gcc-13 javascript-common libalgorithm-diff-perl libalgo
  libalgorithm-merge-perl libasan8 libatomic1 libbinutils libc-dev-bin libc-devtools libc6-dev libcc1-0 libcrypt-dev libctf-nobfd0 libctf0 libdpkg-perl libexpat1-dev libfakeroot libff
  libfile-fcntllock-perl libgcc-13-dev libgd3 libgomp1 libgprofng0 libhwasan0 libisl23 libitm1 libjs-jquery libjs-sphinxdoc libjs-underscore liblsan0 libmpc3 libnsl-dev libpython3-dev
  libquadmath0 libsframe1 libssl-dev libstdc++-13-dev libtirpc-dev libtsan2 libubsan1 linux-libc-dev lto-disabled-list make manpages-dev python3-dev python3-distlib python3-distutils
  python3-lib2to3 python3-pip-whl python3-platformdirs python3-setuptools-whl python3-virtualenv python3-wheel-whl python3.11-dev rpcsvc-proto virtualenv zlib1g-dev
The following packages will be upgraded:
  dpkg
1 upgraded, 73 newly installed, 0 to remove and 41 not upgraded.
Need to get 86.4 MB of archives.
After this operation, 299 MB of additional disk space will be used.
```

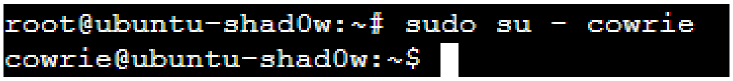
Create a user account

- Create a user called cowrie and switch to the user
  `sudo adduser --disabled-password cowrie`



```
root@ubuntu-shad0w:~# sudo adduser --disabled-password cowrie
info: Adding user `cowrie' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `cowrie' (1000) ...
info: Adding new user `cowrie' (1000) with group `cowrie (1000)' ...
info: Creating home directory `/home/cowrie' ...
info: Copying files from `/etc/skel' ...
Changing the user information for cowrie
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] Y
info: Adding new user `cowrie' to supplemental / extra groups `users' ...
info: Adding user `cowrie' to group `users' ...
```

`sudo su - cowrie`



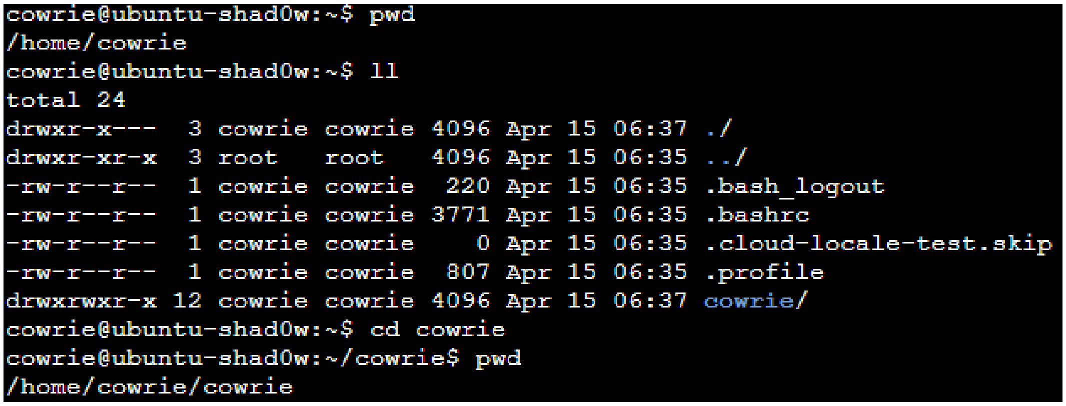
```
root@ubuntu-shad0w:~# sudo su - cowrie
cowrie@ubuntu-shad0w:~$
```

Checkout the code

```
git clone http://github.com/cowrie/cowrie
```

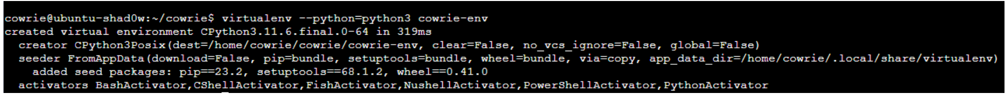
Setup Virtual Environment

- Change directory to the cowrie directory within the user cowrie.
  `cd /home/cowrie/cowrie`



```
cowrie@ubuntu-shad0w:~$ pwd
/home/cowrie
cowrie@ubuntu-shad0w:~$ ll
total 24
drwxr-x---   3 cowrie cowrie 4096 Apr 15 06:37 ./
drwxr-xr-x   3 root   root   4096 Apr 15 06:35 ../
-rw-r--r--   1 cowrie cowrie  220 Apr 15 06:35 .bash_logout
-rw-r--r--   1 cowrie cowrie 3771 Apr 15 06:35 .bashrc
-rw-r--r--   1 cowrie cowrie    0 Apr 15 06:35 .cloud-locale-test.skip
-rw-r--r--   1 cowrie cowrie  807 Apr 15 06:35 .profile
drwxrwxr-x  12 cowrie cowrie 4096 Apr 15 06:37 cowrie/
cowrie@ubuntu-shad0w:~$ cd cowrie
cowrie@ubuntu-shad0w:~/cowrie$ pwd
/home/cowrie/cowrie
```

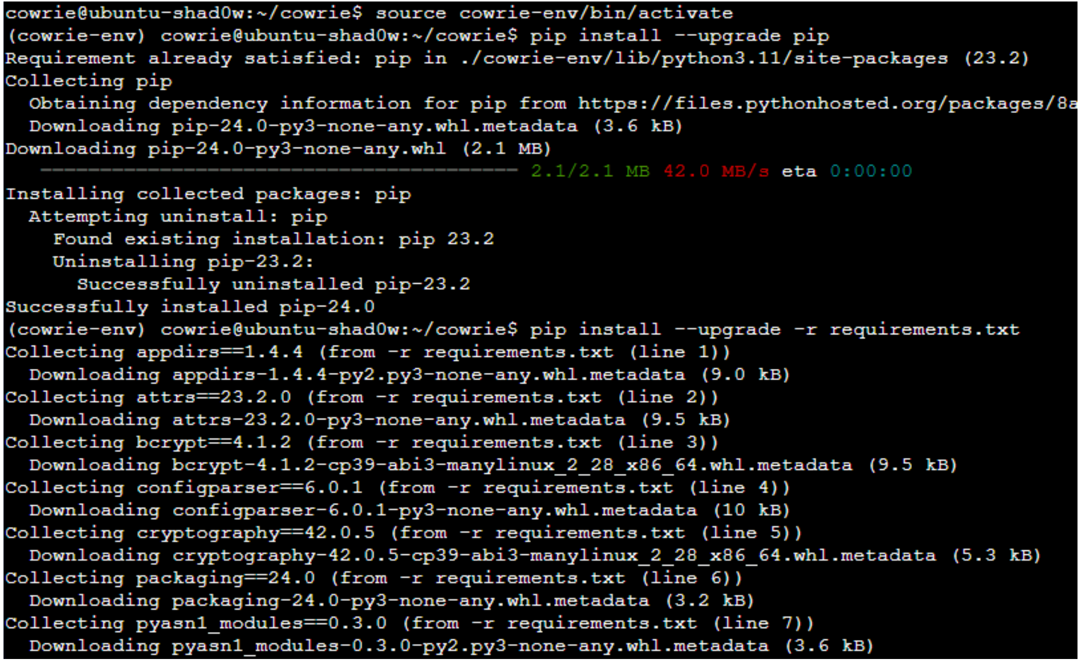
- Creating an new python environment called `cowrie-env` with `python3`
  `virtualenv --python=python3 cowrie-env`



```
cowrie@ubuntu-shad0w:~/cowrie$ virtualenv --python=python3 cowrie-env
created virtual environment CPython3.11.6.final.0-64 in 319ms
  creator CPython3Posix(dest=/home/cowrie/cowrie/cowrie-env, clear=False, no_vcs_ignore=False, global=False)
  seeder FromAppData(download=False, pip=bundle, setuptools=bundle, wheel=bundle, via=copy, app_data_dir=/home/cowrie/.local/share/virtualenv)
    added seed packages: pip==23.2, setuptools==68.1.2, wheel==0.41.0
  activators BashActivator,CShellActivator,FishActivator,NushellActivator,PowerShellActivator,PythonActivator
```

- Setting up the environment to use its isolated Python interpreter and libraries. Upgrades pip and all other dependencies in `requirements.txt`

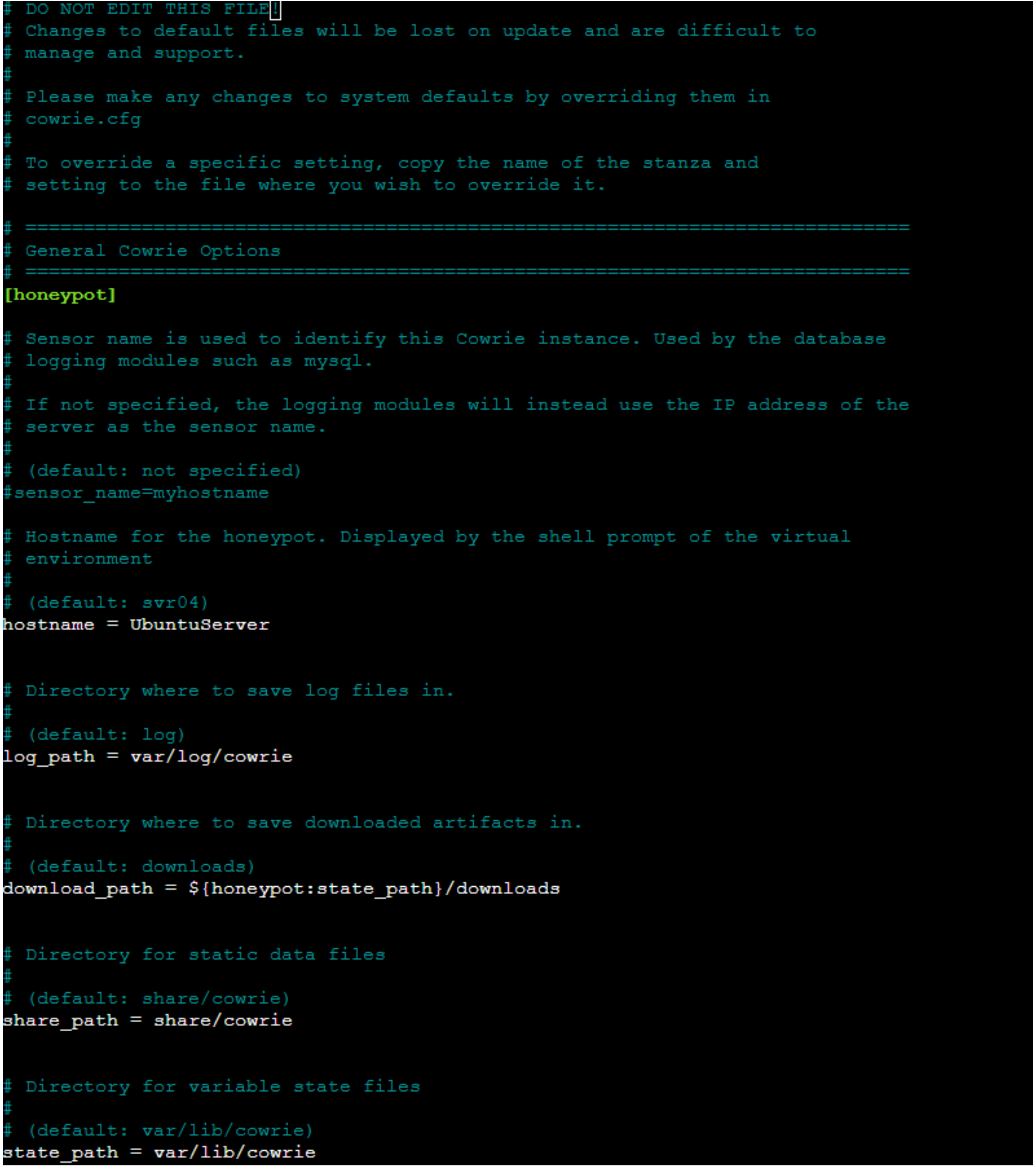
```
source cowrie-env/bin/activate
(cowrie-env) $ pip install --upgrade pip
(cowrie-env) $ pip install --upgrade -r requirements.txt
```



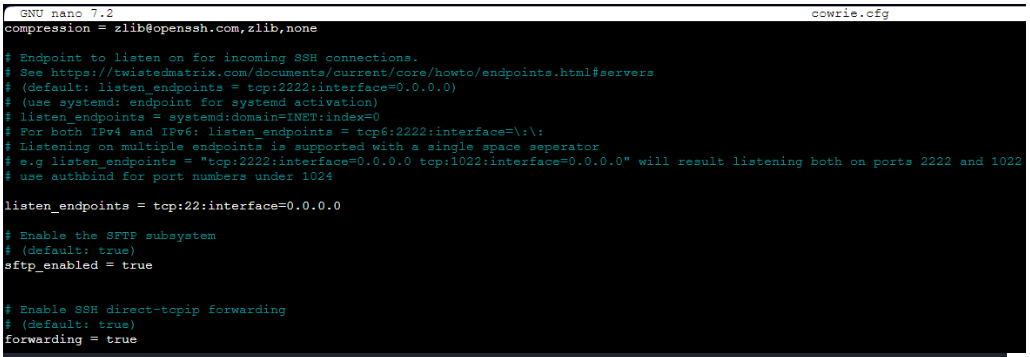
```
cowrie@ubuntu-shad0w:~/cowrie$ source cowrie-env/bin/activate
(cowrie-env) cowrie@ubuntu-shad0w:~/cowrie$ pip install --upgrade pip
Requirement already satisfied: pip in ./cowrie-env/lib/python3.11/site-packages (23.2)
Collecting pip
  Obtaining dependency information for pip from https://files.pythonhosted.org/packages/8a
  Downloading pip-24.0-py3-none-any.whl.metadata (3.6 kB)
Downloading pip-24.0-py3-none-any.whl (2.1 MB)
   ---------------------------------------- 2.1/2.1 MB 42.0 MB/s eta 0:00:00
Installing collected packages: pip
  Attempting uninstall: pip
    Found existing installation: pip 23.2
    Uninstalling pip-23.2:
      Successfully uninstalled pip-23.2
Successfully installed pip-24.0
(cowrie-env) cowrie@ubuntu-shad0w:~/cowrie$ pip install --upgrade -r requirements.txt
Collecting appdirs==1.4.4 (from -r requirements.txt (line 1))
  Downloading appdirs-1.4.4-py2.py3-none-any.whl.metadata (9.0 kB)
Collecting attrs==23.2.0 (from -r requirements.txt (line 2))
  Downloading attrs-23.2.0-py3-none-any.whl.metadata (9.5 kB)
Collecting bcrypt==4.1.2 (from -r requirements.txt (line 3))
  Downloading bcrypt-4.1.2-cp39-abi3-manylinux_2_28_x86_64.whl.metadata (9.5 kB)
Collecting configparser==6.0.1 (from -r requirements.txt (line 4))
  Downloading configparser-6.0.1-py3-none-any.whl.metadata (10 kB)
Collecting cryptography==42.0.5 (from -r requirements.txt (line 5))
  Downloading cryptography-42.0.5-cp39-abi3-manylinux_2_28_x86_64.whl.metadata (5.3 kB)
Collecting packaging==24.0 (from -r requirements.txt (line 6))
  Downloading packaging-24.0-py3-none-any.whl.metadata (3.2 kB)
Collecting pyasn1_modules==0.3.0 (from -r requirements.txt (line 7))
  Downloading pyasn1_modules-0.3.0-py2.py3-none-any.whl.metadata (3.6 kB)
```

Install Configuration File

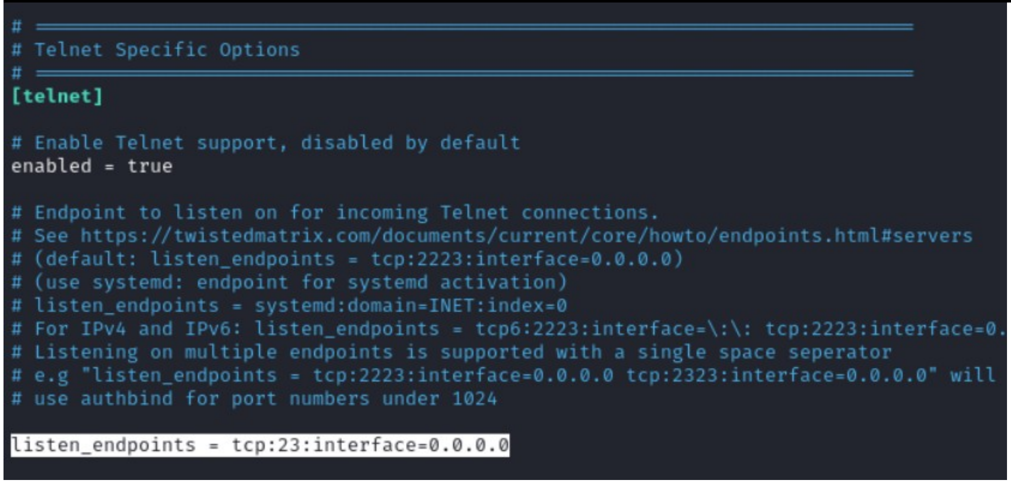
- Copy the file `cowrie.cfg.dist` and rename as `cowrie.cfg`
  `cp /home/cowrie/cowrie/etc/cowrie.cfg.dist /home/cowrie/cowrie/etc/cowrie.cfg`
- Change `enabled = true` for SSH and telnet
- Change the lines at `hostname = <servername>` , `listen_endpoints = tcp:22:interface=0.0.0.0` and `listen_endpoints = tcp:23:interface=0.0.0.0`
  `nano /home/cowrie/cowrie/etc/cowrie.cfg`



```
# DO NOT EDIT THIS FILE!
# Changes to default files will be lost on update and are difficult to
# manage and support.
#
# Please make any changes to system defaults by overriding them in
# cowrie.cfg
#
# To override a specific setting, copy the name of the stanza and
# setting to the file where you wish to override it.

# ============================================================================
# General Cowrie Options
# ============================================================================
[honeypot]

# Sensor name is used to identify this Cowrie instance. Used by the database
# logging modules such as mysql.
#
# If not specified, the logging modules will instead use the IP address of the
# server as the sensor name.
#
# (default: not specified)
#sensor_name=myhostname

# Hostname for the honeypot. Displayed by the shell prompt of the virtual
# environment
#
# (default: svr04)
hostname = UbuntuServer


# Directory where to save log files in.
#
# (default: log)
log_path = var/log/cowrie


# Directory where to save downloaded artifacts in.
#
# (default: downloads)
download_path = ${honeypot:state_path}/downloads


# Directory for static data files
#
# (default: share/cowrie)
share_path = share/cowrie


# Directory for variable state files
#
# (default: var/lib/cowrie)
state_path = var/lib/cowrie
```



```
  GNU nano 7.2                                                                                    cowrie.cfg
compression = zlib@openssh.com,zlib,none

# Endpoint to listen on for incoming SSH connections.
# See https://twistedmatrix.com/documents/current/core/howto/endpoints.html#servers
# (default: listen_endpoints = tcp:2222:interface=0.0.0.0)
# (use systemd: endpoint for systemd activation)
# listen_endpoints = systemd:domain=INET:index=0
# For both IPv4 and IPv6: listen_endpoints = tcp6:2222:interface=\:\:
# Listening on multiple endpoints is supported with a single space seperator
# e.g listen_endpoints = "tcp:2222:interface=0.0.0.0 tcp:1022:interface=0.0.0.0" will result listening both on ports 2222 and 1022
# use authbind for port numbers under 1024

listen_endpoints = tcp:22:interface=0.0.0.0

# Enable the SFTP subsystem
# (default: true)
sftp_enabled = true


# Enable SSH direct-tcpip forwarding
# (default: true)
forwarding = true
```

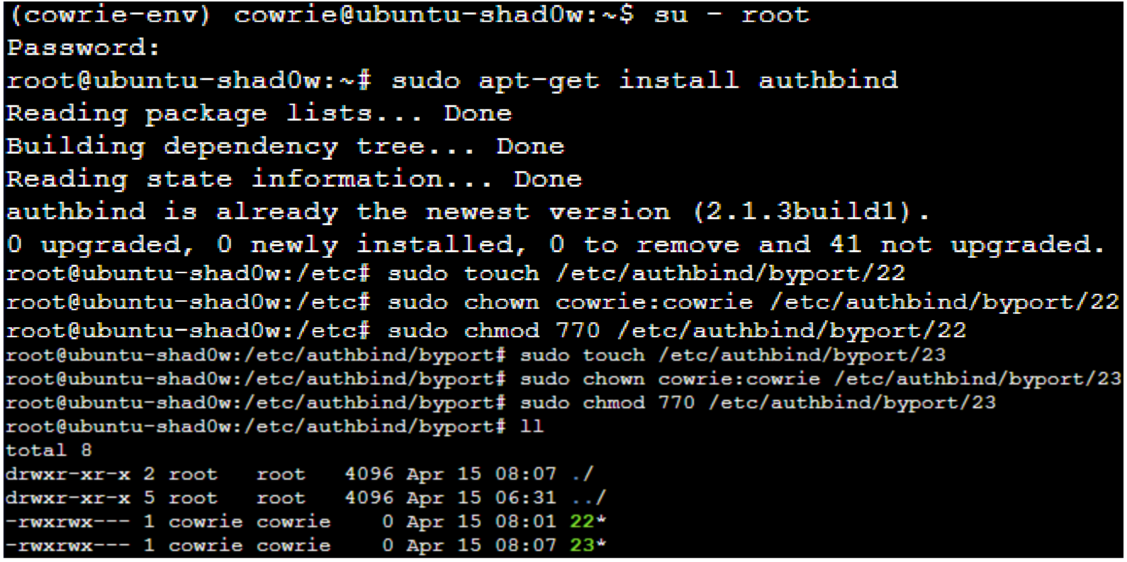


```
# ============================================================================
# Telnet Specific Options
# ============================================================================
[telnet]

# Enable Telnet support, disabled by default
enabled = true

# Endpoint to listen on for incoming Telnet connections.
# See https://twistedmatrix.com/documents/current/core/howto/endpoints.html#servers
# (default: listen_endpoints = tcp:2223:interface=0.0.0.0)
# (use systemd: endpoint for systemd activation)
# listen_endpoints = systemd:domain=INET:index=0
# For IPv4 and IPv6: listen_endpoints = tcp6:2223:interface=\:\: tcp:2223:interface=0.
# Listening on multiple endpoints is supported with a single space seperator
# e.g "listen_endpoints = tcp:2223:interface=0.0.0.0 tcp:2323:interface=0.0.0.0" will
# use authbind for port numbers under 1024

listen_endpoints = tcp:23:interface=0.0.0.0
```

Listening on port 22 & 23 via Authbind

Install Authbind to make Cowrie accessible on the default SSH port 22 and telnet port 23

```
sudo apt-get install authbind
sudo touch /etc/authbind/byport/22
sudo chown cowrie:cowrie /etc/authbind/byport/22
sudo chmod 770 /etc/authbind/byport/22
```

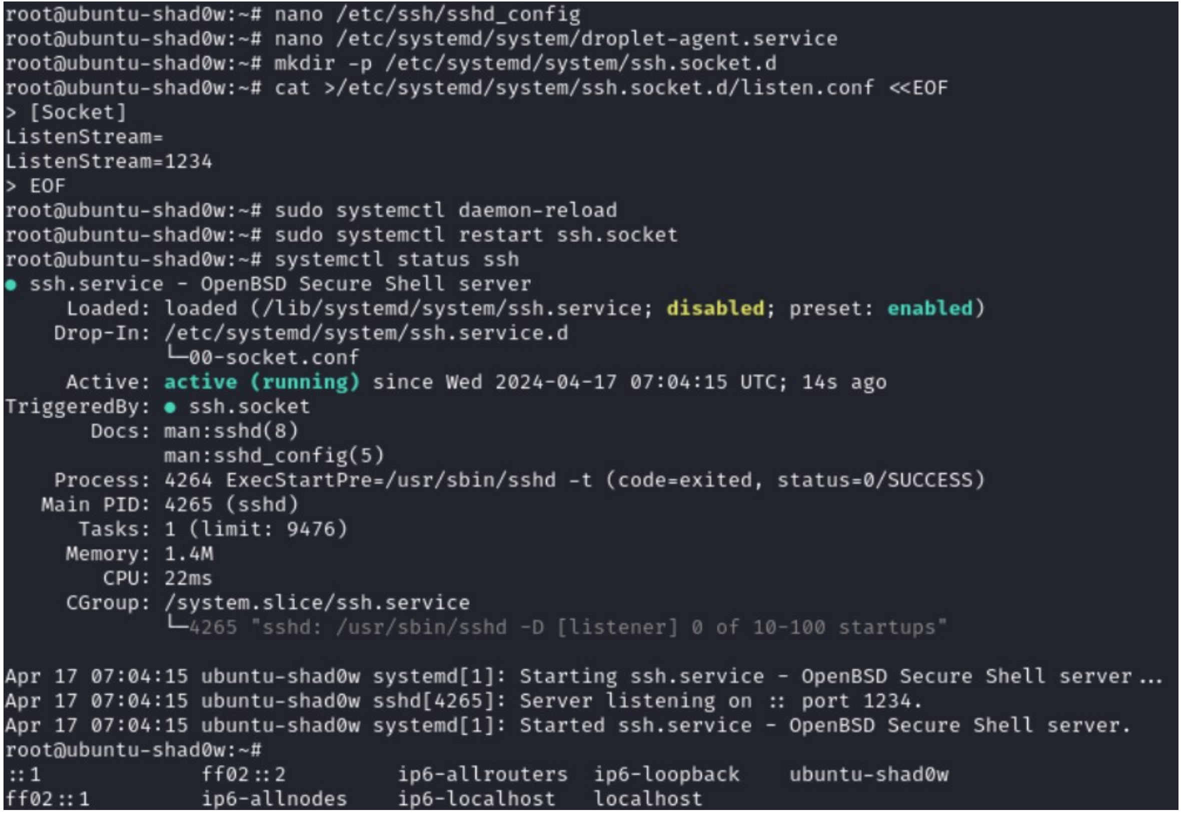


```
(cowrie-env) cowrie@ubuntu-shad0w:~$ su - root
Password:
root@ubuntu-shad0w:~# sudo apt-get install authbind
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
authbind is already the newest version (2.1.3build1).
0 upgraded, 0 newly installed, 0 to remove and 41 not upgraded.
root@ubuntu-shad0w:/etc# sudo touch /etc/authbind/byport/22
root@ubuntu-shad0w:/etc# sudo chown cowrie:cowrie /etc/authbind/byport/22
root@ubuntu-shad0w:/etc# sudo chmod 770 /etc/authbind/byport/22
root@ubuntu-shad0w:/etc/authbind/byport# sudo touch /etc/authbind/byport/23
root@ubuntu-shad0w:/etc/authbind/byport# sudo chown cowrie:cowrie /etc/authbind/byport/23
root@ubuntu-shad0w:/etc/authbind/byport# sudo chmod 770 /etc/authbind/byport/23
root@ubuntu-shad0w:/etc/authbind/byport# ll
total 8
drwxr-xr-x 2 root   root   4096 Apr 15 08:07 ./
drwxr-xr-x 5 root   root   4096 Apr 15 06:31 ../
-rwxrwx--- 1 cowrie cowrie    0 Apr 15 08:01 22*
-rwxrwx--- 1 cowrie cowrie    0 Apr 15 08:07 23*
```

Changing SSH service to a different port

- Configure the OpenSSH service to a different port, do not pick obvious port numbers like 2222 or any ports already in use by other services. Use `netstat -tpan` to check ports in-use.
  ```
  nano /etc/systemd/system/droplet-agent.service
  mkdir -p /etc/systemd/system/ssh.socket.d
  cat >/etc/systemd/system/ssh.socket.d/listen.conf <<EOF
  ```

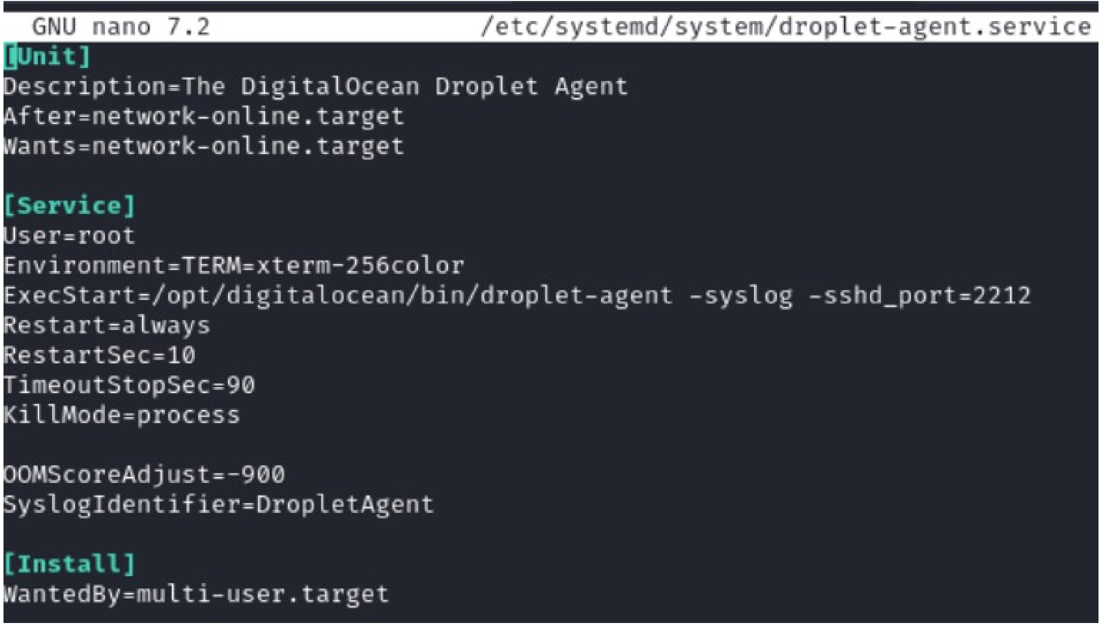
```
> [Socket]
ListenStream=
ListenStream=<port_number>
> EOF
Copy

sudo systemctl daemon-reload
sudo systemctl restart ssh.socket
```



```
root@ubuntu-shad0w:~# nano /etc/ssh/sshd_config
root@ubuntu-shad0w:~# nano /etc/systemd/system/droplet-agent.service
root@ubuntu-shad0w:~# mkdir -p /etc/systemd/system/ssh.socket.d
root@ubuntu-shad0w:~# cat >/etc/systemd/system/ssh.socket.d/listen.conf <<EOF
> [Socket]
ListenStream=
ListenStream=1234
> EOF
root@ubuntu-shad0w:~# sudo systemctl daemon-reload
root@ubuntu-shad0w:~# sudo systemctl restart ssh.socket
root@ubuntu-shad0w:~# systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
     Loaded: loaded (/lib/systemd/system/ssh.service; disabled; preset: enabled)
    Drop-In: /etc/systemd/system/ssh.service.d
             └─00-socket.conf
     Active: active (running) since Wed 2024-04-17 07:04:15 UTC; 14s ago
TriggeredBy: ● ssh.socket
       Docs: man:sshd(8)
             man:sshd_config(5)
    Process: 4264 ExecStartPre=/usr/sbin/sshd -t (code=exited, status=0/SUCCESS)
   Main PID: 4265 (sshd)
      Tasks: 1 (limit: 9476)
     Memory: 1.4M
        CPU: 22ms
     CGroup: /system.slice/ssh.service
             └─4265 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Apr 17 07:04:15 ubuntu-shad0w systemd[1]: Starting ssh.service - OpenBSD Secure Shell server...
Apr 17 07:04:15 ubuntu-shad0w sshd[4265]: Server listening on :: port 1234.
Apr 17 07:04:15 ubuntu-shad0w systemd[1]: Started ssh.service - OpenBSD Secure Shell server.
root@ubuntu-shad0w:~#
::1             ff02::2         ip6-allrouters  ip6-loopback    ubuntu-shad0w
ff02::1         ip6-allnodes    ip6-localhost   localhost
```

- Configure the digital ocean start up service as the same port number as `listen.conf`. Then, change `ExecStart` to `ExecStart=/opt/digitalocean/bin/droplet-agent -syslog -sshd_port=<port_number>`
  ```
  nano /etc/systemd/system/droplet-agent.service
  sudo systemctl daemon-reload
  ```

```
sudo systemctl restart droplet-agent
```

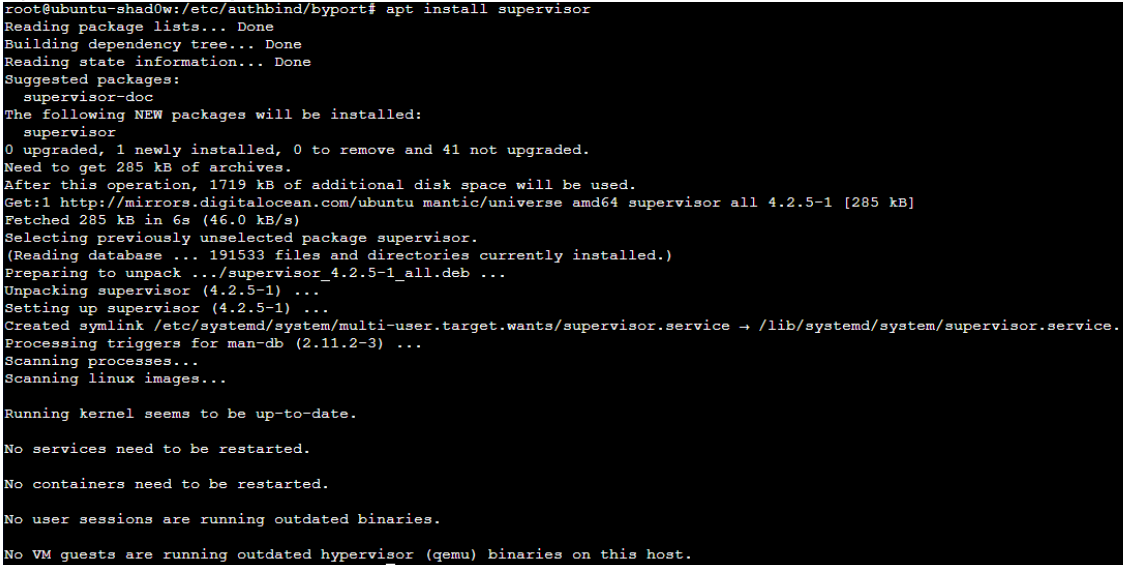


```
  GNU nano 7.2                    /etc/systemd/system/droplet-agent.service
[Unit]
Description=The DigitalOcean Droplet Agent
After=network-online.target
Wants=network-online.target

[Service]
User=root
Environment=TERM=xterm-256color
ExecStart=/opt/digitalocean/bin/droplet-agent -syslog -sshd_port=2212
Restart=always
RestartSec=10
TimeoutStopSec=90
KillMode=process

OOMScoreAdjust=-900
SyslogIdentifier=DropletAgent

[Install]
WantedBy=multi-user.target
```

Running with Supervisord

- Install Supervisord

```
apt install supervisor
```



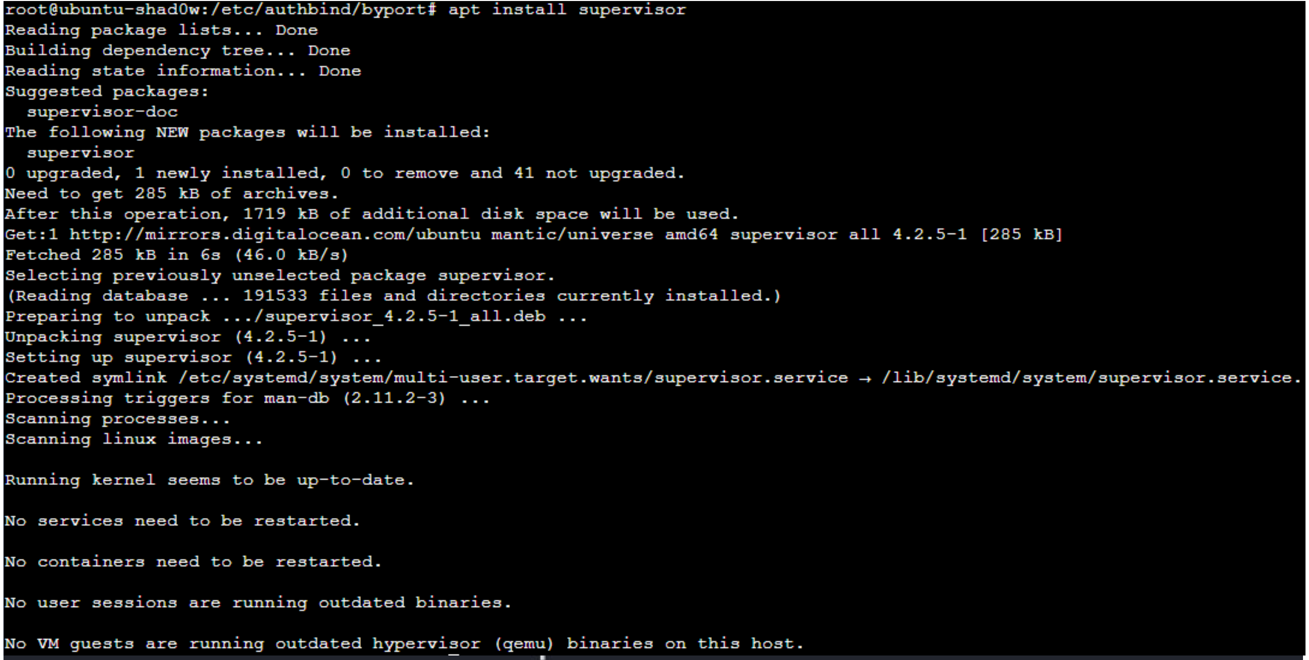
```
root@ubuntu-shad0w:/etc/authbind/byport# apt install supervisor
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  supervisor-doc
The following NEW packages will be installed:
  supervisor
0 upgraded, 1 newly installed, 0 to remove and 41 not upgraded.
Need to get 285 kB of archives.
After this operation, 1719 kB of additional disk space will be used.
Get:1 http://mirrors.digitalocean.com/ubuntu mantic/universe amd64 supervisor all 4.2.5-1 [285 kB]
Fetched 285 kB in 6s (46.0 kB/s)
Selecting previously unselected package supervisor.
(Reading database ... 191533 files and directories currently installed.)
Preparing to unpack .../supervisor_4.2.5-1_all.deb ...
Unpacking supervisor (4.2.5-1) ...
Setting up supervisor (4.2.5-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/supervisor.service → /lib/systemd/system/supervisor.service.
Processing triggers for man-db (2.11.2-3) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
```

- Daemonize cowrie under supervisord in `cowrie.conf`, then update supervisorctl

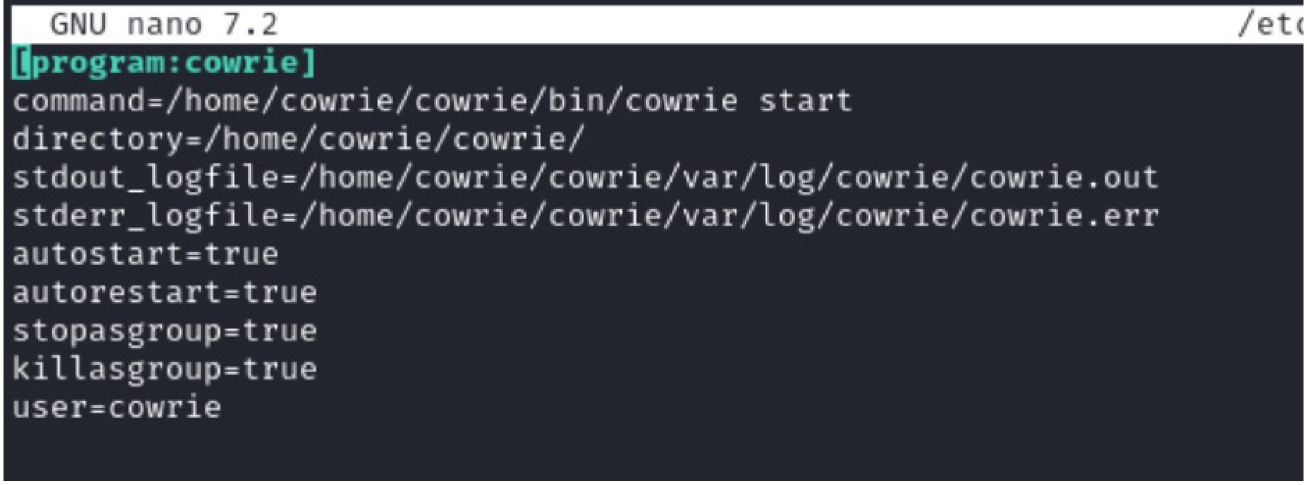
```
cat > /etc/supervisor/conf.d/cowrie.conf <<EOF
```

```
> [program:cowrie]
command=/opt/cowrie/bin/cowrie start
directory=/opt/cowrie
stdout_logfile=/opt/cowrie/var/log/cowrie/cowrie.out
stderr_logfile=/opt/cowrie/var/log/cowrie/cowrie.err
autostart=true
autorestart=true
stopasgroup=true
killasgroup=true
user=cowrie
> EOF
```

```
supervisorctl update
```



```
root@ubuntu-shad0w:/etc/authbind/byport# apt install supervisor
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
Suggested packages:
  supervisor-doc
The following NEW packages will be installed:
  supervisor
0 upgraded, 1 newly installed, 0 to remove and 41 not upgraded.
Need to get 285 kB of archives.
After this operation, 1719 kB of additional disk space will be used.
Get:1 http://mirrors.digitalocean.com/ubuntu mantic/universe amd64 supervisor all 4.2.5-1 [285 kB]
Fetched 285 kB in 6s (46.0 kB/s)
Selecting previously unselected package supervisor.
(Reading database ... 191533 files and directories currently installed.)
Preparing to unpack .../supervisor_4.2.5-1_all.deb ...
Unpacking supervisor (4.2.5-1) ...
Setting up supervisor (4.2.5-1) ...
Created symlink /etc/systemd/system/multi-user.target.wants/supervisor.service → /lib/systemd/system/supervisor.service.
Processing triggers for man-db (2.11.2-3) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
```

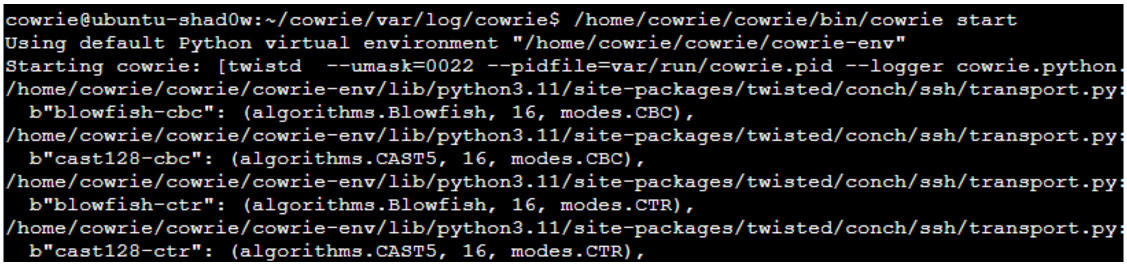


```
  GNU nano 7.2                                      /etc
[program:cowrie]
command=/home/cowrie/cowrie/bin/cowrie start
directory=/home/cowrie/cowrie/
stdout_logfile=/home/cowrie/cowrie/var/log/cowrie/cowrie.out
stderr_logfile=/home/cowrie/cowrie/var/log/cowrie/cowrie.err
autostart=true
autorestart=true
stopasgroup=true
killasgroup=true
user=cowrie
```

Starting Cowrie

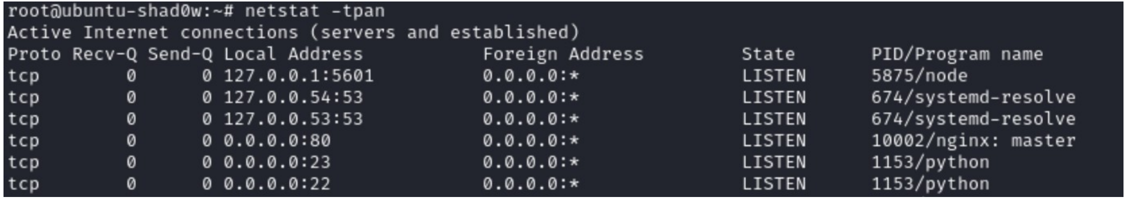
- Once Supervisorctl is installed, you can start cowrie with it
  `supervisorctl`

- Alternatively, if Supervisorctl doesn't work, start cowrie manually
  `/home/cowrie/cowrie/bin/cowrie start`



```
cowrie@ubuntu-shad0w:~/cowrie/var/log/cowrie$ /home/cowrie/cowrie/bin/cowrie start
Using default Python virtual environment "/home/cowrie/cowrie/cowrie-env"
Starting cowrie: [twistd  --umask=0022 --pidfile=var/run/cowrie.pid --logger cowrie.python.
/home/cowrie/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:
  b"blowfish-cbc": (algorithms.Blowfish, 16, modes.CBC),
/home/cowrie/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:
  b"cast128-cbc": (algorithms.CAST5, 16, modes.CBC),
/home/cowrie/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:
  b"blowfish-ctr": (algorithms.Blowfish, 16, modes.CTR),
/home/cowrie/cowrie/cowrie-env/lib/python3.11/site-packages/twisted/conch/ssh/transport.py:
  b"cast128-ctr": (algorithms.CAST5, 16, modes.CTR),
```

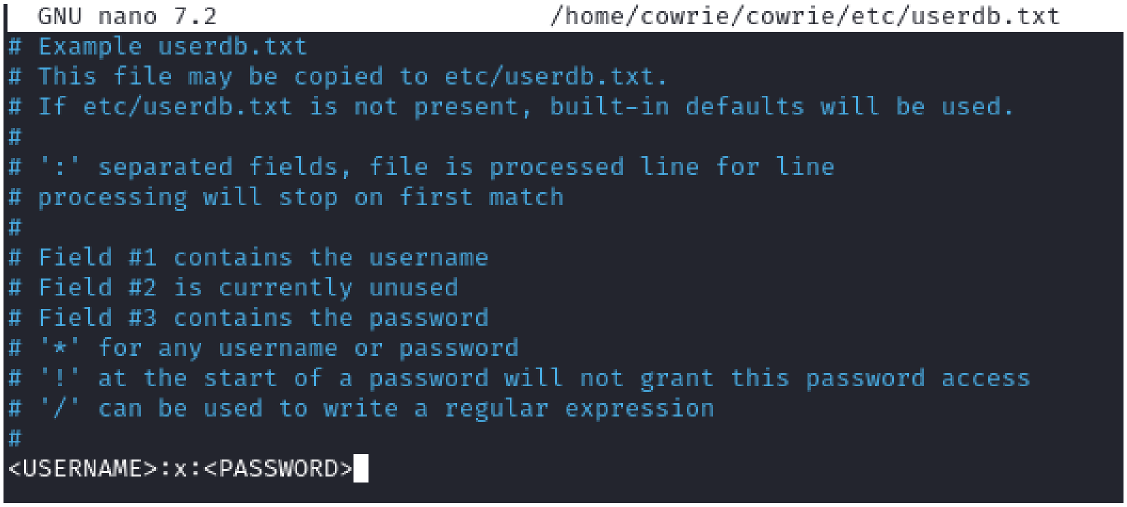
- Check if cowrie has the correct processes listening in every port. Port 22 and 23 should be listening and running on python
  `netstat -tpan`



```
root@ubuntu-shad0w:~# netstat -tpan
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:5601          0.0.0.0:*               LISTEN      5875/node
tcp        0      0 127.0.0.54:53           0.0.0.0:*               LISTEN      674/systemd-resolve
tcp        0      0 127.0.0.53:53           0.0.0.0:*               LISTEN      674/systemd-resolve
tcp        0      0 0.0.0.0:80              0.0.0.0:*               LISTEN      10002/nginx: master
tcp        0      0 0.0.0.0:23              0.0.0.0:*               LISTEN      1153/python
tcp        0      0 0.0.0.0:22              0.0.0.0:*               LISTEN      1153/python
```

Configuring Honeypot Access

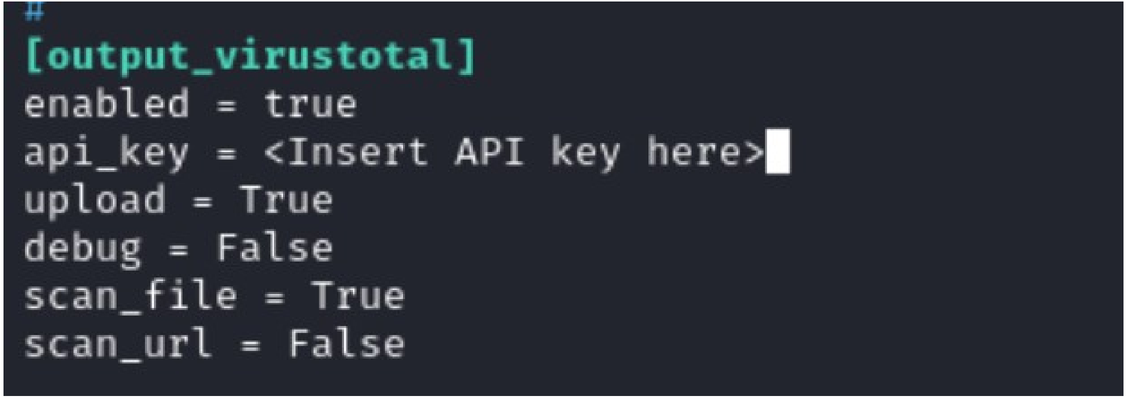
- Depending on what information you wish to collect on the attackers, you can set the difficulty by editing the users that can access the honeypot.
  `cp /home/cowrie/cowrie/etc/userdb.example /home/cowrie/cowrie/etc/userdb.txt`
  `nano /home/cowrie/cowrie/etc/userdb.txt`



```
  GNU nano 7.2                       /home/cowrie/cowrie/etc/userdb.txt
# Example userdb.txt
# This file may be copied to etc/userdb.txt.
# If etc/userdb.txt is not present, built-in defaults will be used.
#
# ':' separated fields, file is processed line for line
# processing will stop on first match
#
# Field #1 contains the username
# Field #2 is currently unused
# Field #3 contains the password
# '*' for any username or password
# '!' at the start of a password will not grant this password access
# '/' can be used to write a regular expression
#
<USERNAME>:x:<PASSWORD>
```

Integration with VirusTotal

- Integration allows you to view payloads/malware uploaded on the honeypot for analysis on VirusTotal.
- First, create a VirusTotal account and copy the api key
- Edit the `cowrie.cfg` file and insert the api key
  `nano /home/cowrie/cowrie/etc/cowrie.cfg`



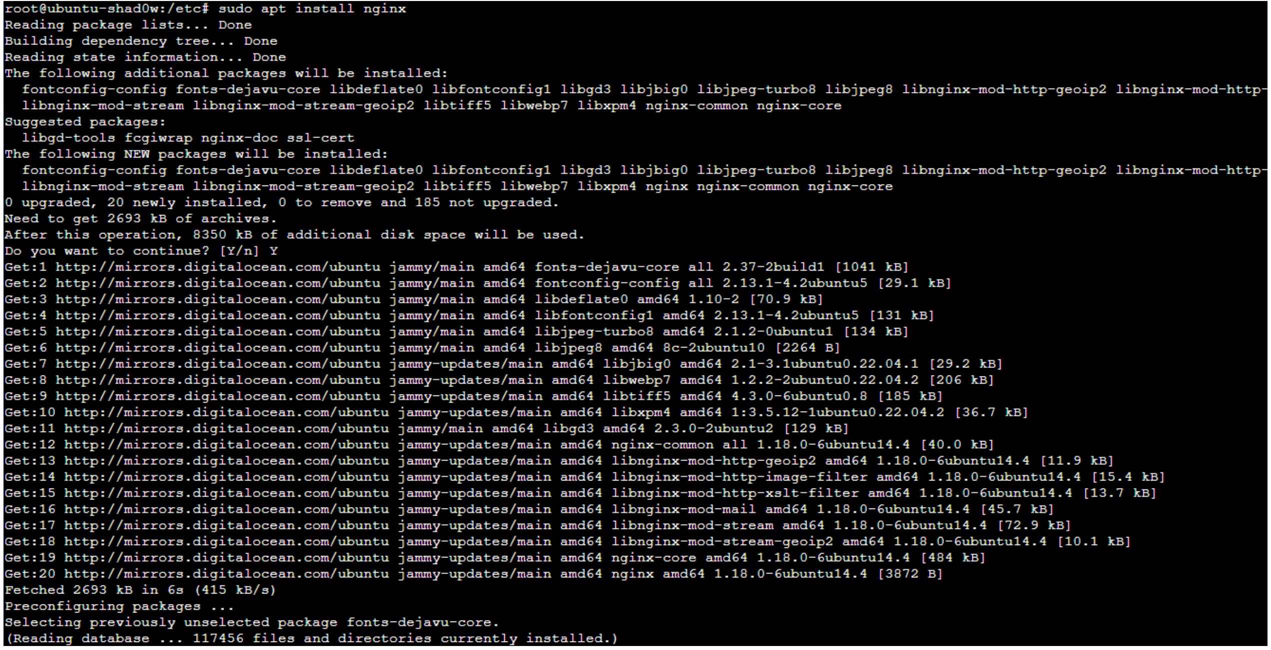
```
#
[output_virustotal]
enabled = true
api_key = <Insert API key here>
upload = True
debug = False
scan_file = True
scan_url = False
```

Installing Elastic Cloud on DigitalOcean

Installing Nginx and Server Hardening

Installing Nginx

```
sudo apt install nginx
```

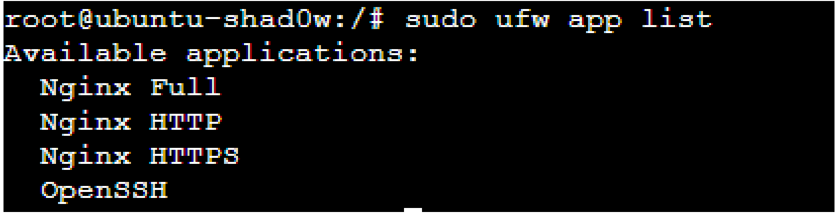


```
root@ubuntu-shad0w:/etc# sudo apt install nginx
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2 libnginx-mod-http-
  libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5 libwebp7 libxpm4 nginx-common nginx-core
Suggested packages:
  libgd-tools fcgiwrap nginx-doc ssl-cert
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core libdeflate0 libfontconfig1 libgd3 libjbig0 libjpeg-turbo8 libjpeg8 libnginx-mod-http-geoip2 libnginx-mod-http-
  libnginx-mod-stream libnginx-mod-stream-geoip2 libtiff5 libwebp7 libxpm4 nginx nginx-common nginx-core
0 upgraded, 20 newly installed, 0 to remove and 185 not upgraded.
Need to get 2693 kB of archives.
After this operation, 8350 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://mirrors.digitalocean.com/ubuntu jammy/main amd64 fonts-dejavu-core all 2.37-2build1 [1041 kB]
Get:2 http://mirrors.digitalocean.com/ubuntu jammy/main amd64 fontconfig-config all 2.13.1-4.2ubuntu5 [29.1 kB]
Get:3 http://mirrors.digitalocean.com/ubuntu jammy/main amd64 libdeflate0 amd64 1.10-2 [70.9 kB]
Get:4 http://mirrors.digitalocean.com/ubuntu jammy/main amd64 libfontconfig1 amd64 2.13.1-4.2ubuntu5 [131 kB]
Get:5 http://mirrors.digitalocean.com/ubuntu jammy/main amd64 libjpeg-turbo8 amd64 2.1.2-0ubuntu1 [134 kB]
Get:6 http://mirrors.digitalocean.com/ubuntu jammy/main amd64 libjpeg8 amd64 8c-2ubuntu10 [2264 B]
Get:7 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libjbig0 amd64 2.1-3.1ubuntu0.22.04.1 [29.2 kB]
Get:8 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libwebp7 amd64 1.2.2-2ubuntu0.22.04.2 [206 kB]
Get:9 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libtiff5 amd64 4.3.0-6ubuntu0.8 [185 kB]
Get:10 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libxpm4 amd64 1:3.5.12-1ubuntu0.22.04.2 [36.7 kB]
Get:11 http://mirrors.digitalocean.com/ubuntu jammy/main amd64 libgd3 amd64 2.3.0-2ubuntu2 [129 kB]
Get:12 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 nginx-common all 1.18.0-6ubuntu14.4 [40.0 kB]
Get:13 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libnginx-mod-http-geoip2 amd64 1.18.0-6ubuntu14.4 [11.9 kB]
Get:14 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libnginx-mod-http-image-filter amd64 1.18.0-6ubuntu14.4 [15.4 kB]
Get:15 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libnginx-mod-http-xslt-filter amd64 1.18.0-6ubuntu14.4 [13.7 kB]
Get:16 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libnginx-mod-mail amd64 1.18.0-6ubuntu14.4 [45.7 kB]
Get:17 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libnginx-mod-stream amd64 1.18.0-6ubuntu14.4 [72.9 kB]
Get:18 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 libnginx-mod-stream-geoip2 amd64 1.18.0-6ubuntu14.4 [10.1 kB]
Get:19 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 nginx-core amd64 1.18.0-6ubuntu14.4 [484 kB]
Get:20 http://mirrors.digitalocean.com/ubuntu jammy-updates/main amd64 nginx amd64 1.18.0-6ubuntu14.4 [3872 B]
Fetched 2693 kB in 6s (415 kB/s)
Preconfiguring packages ...
Selecting previously unselected package fonts-dejavu-core.
(Reading database ... 117456 files and directories currently installed.)
```

Server hardening

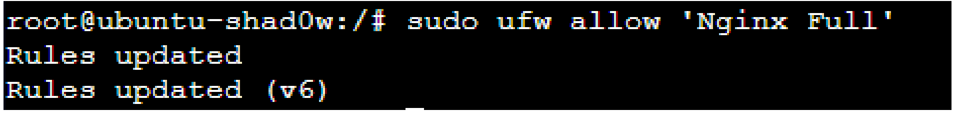
- Enable ufw firewall
  `sudo ufw enable`

- View available applications that work with ufw
  `sudo ufw app list`



```
root@ubuntu-shad0w:/# sudo ufw app list
Available applications:
  Nginx Full
  Nginx HTTP
  Nginx HTTPS
  OpenSSH
```

- Select nginx allow full to enable both port 80 (HTTP) and port 443 (HTTPS)
  `sudo ufw allow 'Nginx Full'`

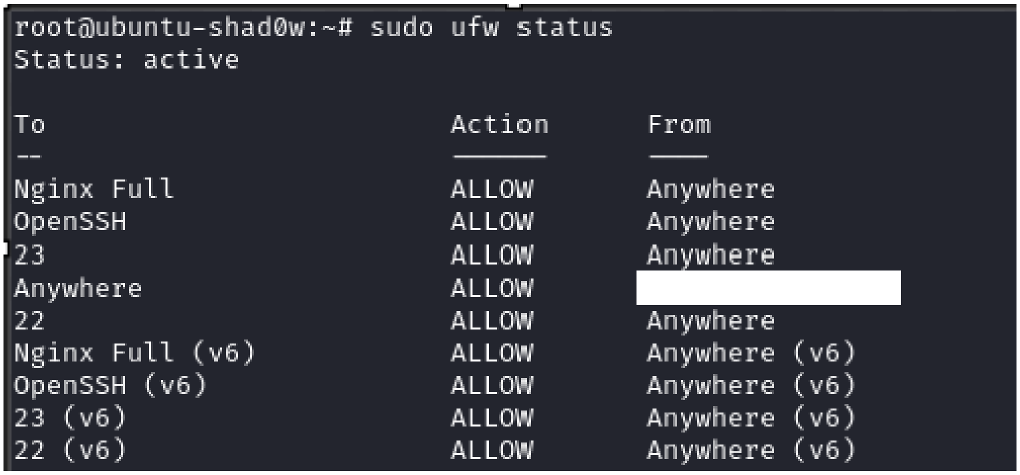


```
root@ubuntu-shad0w:/# sudo ufw allow 'Nginx Full'
Rules updated
Rules updated (v6)
```

- Allow OpenSSH and telnet services for cowrie
  `sudo ufw allow 22`
  `sudo ufw allow 23`

- Allow connection for yourself if you are connecting through a local linux machine
  `sudo ufw allow from <ip address>`

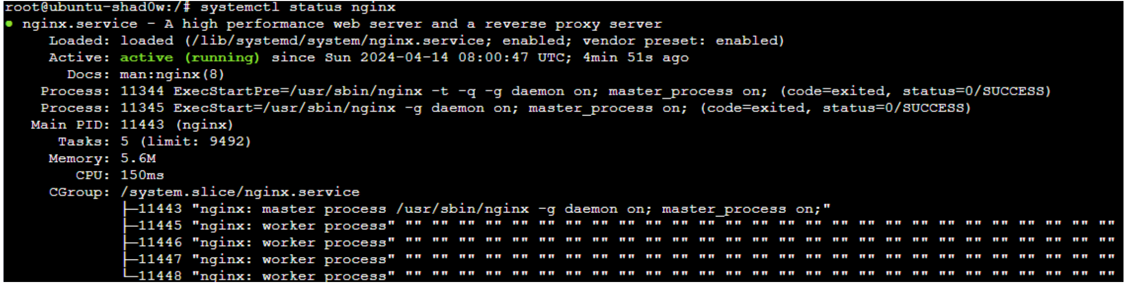
- Checking the firewall rules
  `sudo ufw status`



```
root@ubuntu-shad0w:~# sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
Nginx Full                 ALLOW       Anywhere
OpenSSH                    ALLOW       Anywhere
23                         ALLOW       Anywhere
Anywhere                   ALLOW       [redacted]
22                         ALLOW       Anywhere
Nginx Full (v6)            ALLOW       Anywhere (v6)
OpenSSH (v6)               ALLOW       Anywhere (v6)
23 (v6)                    ALLOW       Anywhere (v6)
22 (v6)                    ALLOW       Anywhere (v6)
```

Checking nginx service status

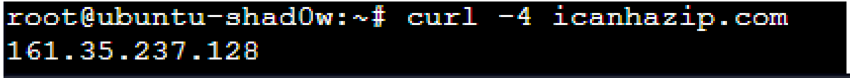
```
systemctl status nginx
```



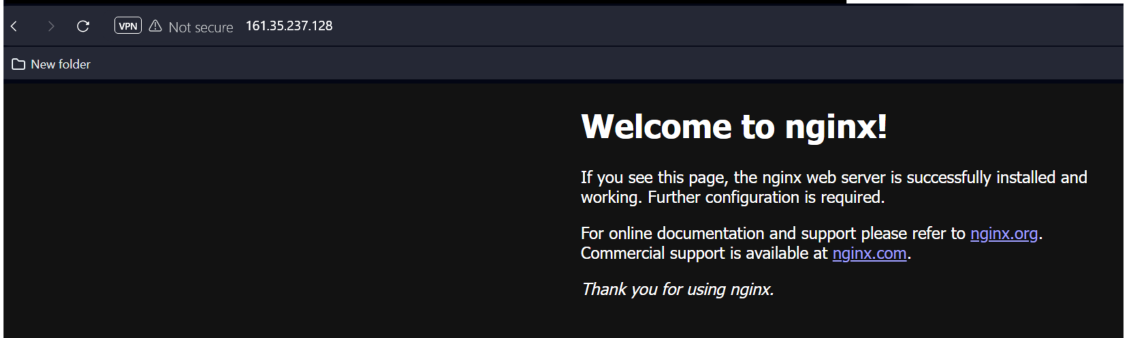
```
root@ubuntu-shad0w:/# systemctl status nginx
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2024-04-14 08:00:47 UTC; 4min 51s ago
       Docs: man:nginx(8)
    Process: 11344 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 11345 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 11443 (nginx)
      Tasks: 5 (limit: 9492)
     Memory: 5.6M
        CPU: 150ms
     CGroup: /system.slice/nginx.service
             ├─11443 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             ├─11445 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
             ├─11446 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
             ├─11447 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
             └─11448 "nginx: worker process" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" "" ""
```

Checking your web server

- Checking if the web server is up. Type the ip address into browser after `curl`
  `curl -4 icanhazip.com`



```
root@ubuntu-shad0w:~# curl -4 icanhazip.com
161.35.237.128
```



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

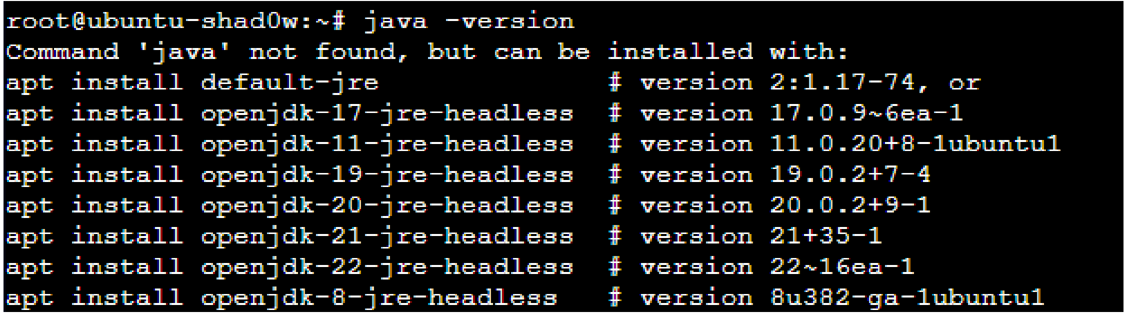
For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

Installing OpenJDK/JRE

Installing Java

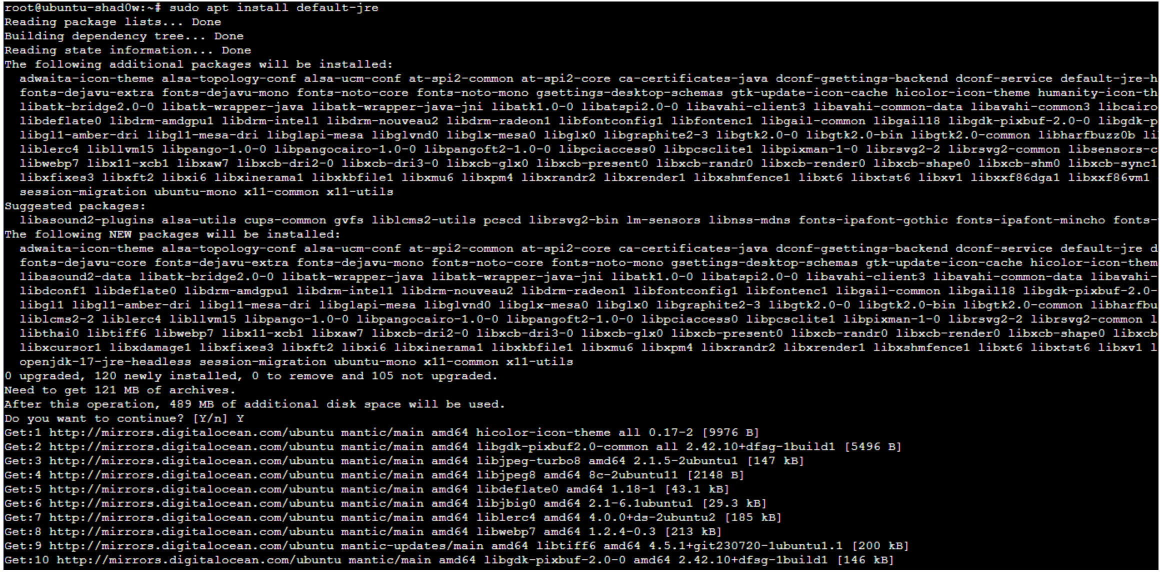
- Check if java is already installed
  `java -version`



```
root@ubuntu-shad0w:~# java -version
Command 'java' not found, but can be installed with:
apt install default-jre              # version 2:1.17-74, or
apt install openjdk-17-jre-headless  # version 17.0.9~6ea-1
apt install openjdk-11-jre-headless  # version 11.0.20+8-1ubuntu1
apt install openjdk-19-jre-headless  # version 19.0.2+7-4
apt install openjdk-20-jre-headless  # version 20.0.2+9-1
apt install openjdk-21-jre-headless  # version 21+35-1
apt install openjdk-22-jre-headless  # version 22~16ea-1
apt install openjdk-8-jre-headless   # version 8u382-ga-1ubuntu1
```

- Install the JRE

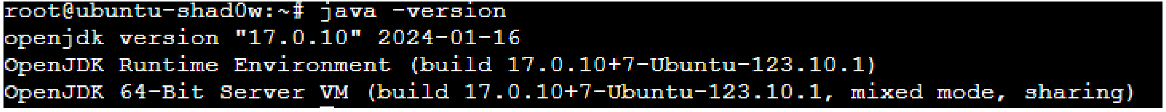
```
sudo apt install default-jre
```



```
root@ubuntu-shad0w:~# sudo apt install default-jre
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  adwaita-icon-theme alsa-topology-conf alsa-ucm-conf at-spi2-common at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-service default-jre-h
  fonts-dejavu-extra fonts-dejavu-mono fonts-noto-core fonts-noto-mono gsettings-desktop-schemas gtk-update-icon-cache hicolor-icon-theme humanity-icon-th
  libatk-bridge2.0-0 libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0 libatspi2.0-0 libavahi-client3 libavahi-common-data libavahi-common3 libcairo
  libdeflate0 libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libfontconfig1 libfontenc1 libgail-common libgail18 libgdk-pixbuf-2.0-0 libgdk-p
  libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3 libgtk2.0-0 libgtk2.0-bin libgtk2.0-common libharfbuzz0b li
  liblerc4 libllvm15 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpciaccess0 libpcsclite1 libpixman-1-0 librsvg2-2 librsvg2-common libsensors-c
  libwebp7 libx11-xcb1 libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0 libxcb-shm0 libxcb-sync1
  libxfixes3 libxft2 libxi6 libxinerama1 libxkbfile1 libxmu6 libxpm4 libxrandr2 libxrender1 libxshmfence1 libxt6 libxtst6 libxv1 libxxf86dga1 libxxf86vm1
  session-migration ubuntu-mono x11-common x11-utils
Suggested packages:
  libasound2-plugins alsa-utils cups-common gvfs liblcms2-utils pcscd librsvg2-bin lm-sensors libnss-mdns fonts-ipafont-gothic fonts-ipafont-mincho fonts-
The following NEW packages will be installed:
  adwaita-icon-theme alsa-topology-conf alsa-ucm-conf at-spi2-common at-spi2-core ca-certificates-java dconf-gsettings-backend dconf-service default-jre d
  fonts-dejavu-core fonts-dejavu-extra fonts-dejavu-mono fonts-noto-core fonts-noto-mono gsettings-desktop-schemas gtk-update-icon-cache hicolor-icon-them
  libasound2-data libatk-bridge2.0-0 libatk-wrapper-java libatk-wrapper-java-jni libatk1.0-0 libatspi2.0-0 libavahi-client3 libavahi-common-data libavahi-
  libdconf1 libdeflate0 libdrm-amdgpu1 libdrm-intel1 libdrm-nouveau2 libdrm-radeon1 libfontconfig1 libfontenc1 libgail-common libgail18 libgdk-pixbuf-2.0-
  libgl1 libgl1-amber-dri libgl1-mesa-dri libglapi-mesa libglvnd0 libglx-mesa0 libglx0 libgraphite2-3 libgtk2.0-0 libgtk2.0-bin libgtk2.0-common libharfbu
  liblcms2-2 liblerc4 libllvm15 libpango-1.0-0 libpangocairo-1.0-0 libpangoft2-1.0-0 libpciaccess0 libpcsclite1 libpixman-1-0 librsvg2-2 librsvg2-common l
  libthai0 libtiff6 libwebp7 libx11-xcb1 libxaw7 libxcb-dri2-0 libxcb-dri3-0 libxcb-glx0 libxcb-present0 libxcb-randr0 libxcb-render0 libxcb-shape0 libxcb
  libxcursor1 libxdamage1 libxfixes3 libxft2 libxi6 libxinerama1 libxkbfile1 libxmu6 libxpm4 libxrandr2 libxrender1 libxshmfence1 libxt6 libxtst6 libxv1 l
  openjdk-17-jre-headless session-migration ubuntu-mono x11-common x11-utils
0 upgraded, 120 newly installed, 0 to remove and 105 not upgraded.
Need to get 121 MB of archives.
After this operation, 489 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 hicolor-icon-theme all 0.17-2 [9976 B]
Get:2 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 libgdk-pixbuf2.0-common all 2.42.10+dfsg-1build1 [5496 B]
Get:3 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 libjpeg-turbo8 amd64 2.1.5-2ubuntu1 [147 kB]
Get:4 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 libjpeg8 amd64 8c-2ubuntu11 [2148 B]
Get:5 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 libdeflate0 amd64 1.18-1 [43.1 kB]
Get:6 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 libjbig0 amd64 2.1-6.1ubuntu1 [29.3 kB]
Get:7 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 liblerc4 amd64 4.0.0+ds-2ubuntu2 [185 kB]
Get:8 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 libwebp7 amd64 1.2.4-0.3 [213 kB]
Get:9 http://mirrors.digitalocean.com/ubuntu mantic-updates/main amd64 libtiff6 amd64 4.5.1+git230720-1ubuntu1.1 [200 kB]
Get:10 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 libgdk-pixbuf-2.0-0 amd64 2.42.10+dfsg-1build1 [146 kB]
```

- Verify that it has been installed

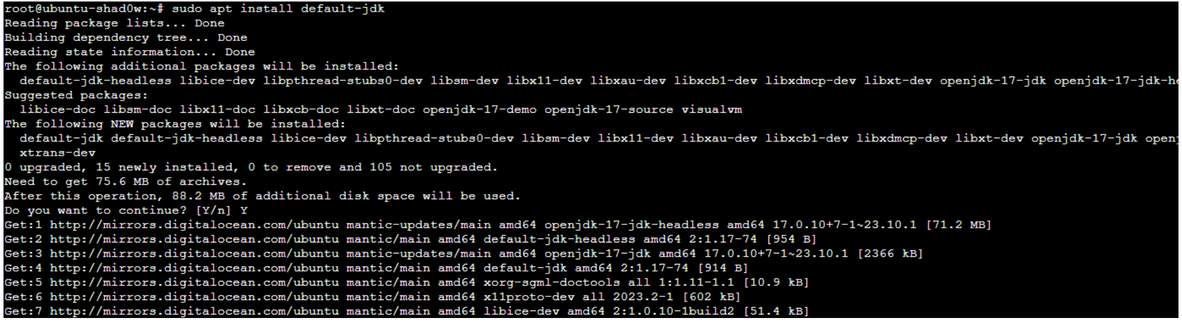
```
java -version
```



```
root@ubuntu-shad0w:~# java -version
openjdk version "17.0.10" 2024-01-16
OpenJDK Runtime Environment (build 17.0.10+7-Ubuntu-123.10.1)
OpenJDK 64-Bit Server VM (build 17.0.10+7-Ubuntu-123.10.1, mixed mode, sharing)
```

- Install JDK in addition to the JRE in order to compile and run some specific Java-based software

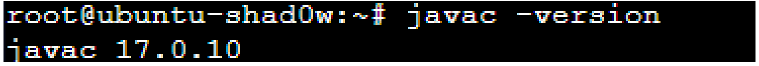
```
sudo apt install default-jdk
```



```
root@ubuntu-shad0w:~# sudo apt install default-jdk
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  default-jdk-headless libice-dev libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-17-jdk openjdk-17-jdk-he
Suggested packages:
  libice-doc libsm-doc libx11-doc libxcb-doc libxt-doc openjdk-17-demo openjdk-17-source visualvm
The following NEW packages will be installed:
  default-jdk default-jdk-headless libice-dev libpthread-stubs0-dev libsm-dev libx11-dev libxau-dev libxcb1-dev libxdmcp-dev libxt-dev openjdk-17-jdk openj
  xtrans-dev
0 upgraded, 15 newly installed, 0 to remove and 105 not upgraded.
Need to get 75.6 MB of archives.
After this operation, 88.2 MB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://mirrors.digitalocean.com/ubuntu mantic-updates/main amd64 openjdk-17-jdk-headless amd64 17.0.10+7-1~23.10.1 [71.2 MB]
Get:2 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 default-jdk-headless amd64 2:1.17-74 [954 B]
Get:3 http://mirrors.digitalocean.com/ubuntu mantic-updates/main amd64 openjdk-17-jdk amd64 17.0.10+7-1~23.10.1 [2366 kB]
Get:4 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 default-jdk amd64 2:1.17-74 [914 B]
Get:5 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 xorg-sgml-doctools all 1:1.11-1.1 [10.9 kB]
Get:6 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 x11proto-dev all 2023.2-1 [602 kB]
Get:7 http://mirrors.digitalocean.com/ubuntu mantic/main amd64 libice-dev amd64 2:1.0.10-1build2 [51.4 kB]
```

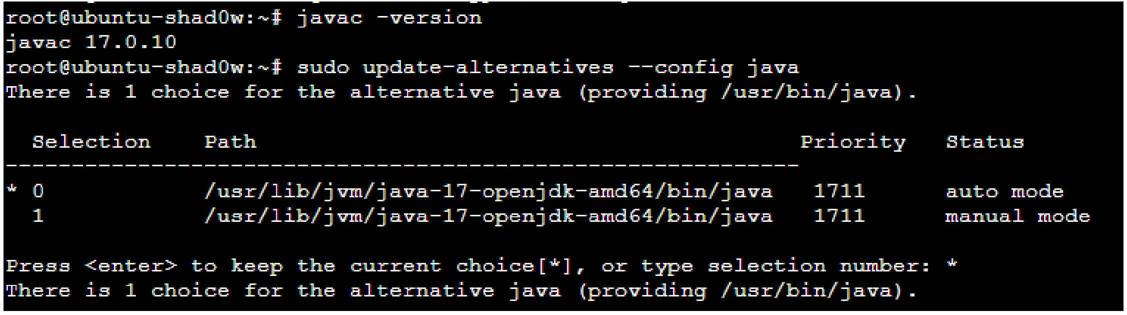
- Verify that the JDK is installed by checking the version of `javac`, the Java compiler

```
javac -version
```



```
root@ubuntu-shad0w:~# javac -version
javac 17.0.10
```

Managing Java

- There are multiple Java installations on one server. You can configure which version is the default for use on the command line by using the `update-alternatives` command.
  ```
  sudo update-alternatives --config java
  ```
  
  ```
  root@ubuntu-shad0w:~# javac -version
  javac 17.0.10
  root@ubuntu-shad0w:~# sudo update-alternatives --config java
  There is 1 choice for the alternative java (providing /usr/bin/java).

    Selection    Path                                         Priority   Status
  ------------------------------------------------------------
  * 0            /usr/lib/jvm/java-17-openjdk-amd64/bin/java   1711      auto mode
    1            /usr/lib/jvm/java-17-openjdk-amd64/bin/java   1711      manual mode

  Press <enter> to keep the current choice[*], or type selection number: *
  There is 1 choice for the alternative java (providing /usr/bin/java).
  ```

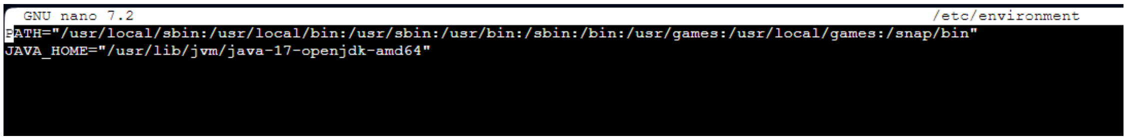
You can do this for other Java commands, such as the compiler (`javac`)

```
sudo update-alternatives --config javac
```

Setting the `JAVA_HOME` Environment Variable

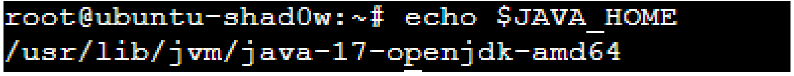
- Add the `JAVA_HOME`to determine the Java installation location. The previous command would show the file path where Java is installed.
- Edit the environment file and add the following text at the end:
  ```
  sudo nano /etc/environment
  ```

```
JAVA_HOME="/usr/lib/jvm/java-11-openjdk-amd64"
Copy
```



```
  GNU nano 7.2                                                          /etc/environment
PATH="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin"
JAVA_HOME="/usr/lib/jvm/java-17-openjdk-amd64"
```

- Reload the file and apply the changes
  ```
  source /etc/environment
  ```

- Verify that the environment variable is set
  ```
  echo $JAVA_HOME
  ```
  
  ```
  root@ubuntu-shad0w:~# echo $JAVA_HOME
  /usr/lib/jvm/java-17-openjdk-amd64
  ```

Installing & Configuring ELK Stack

- Import the Elasticsearch public GPG key and add the Elastic package source list in order to install Elasticsearch
  ```
  curl -fsSL https://artifacts.elastic.co/GPG-KEY-elasticsearch |sudo gpg
  --dearmor -o /usr/share/keyrings/elastic.gpg
  ```

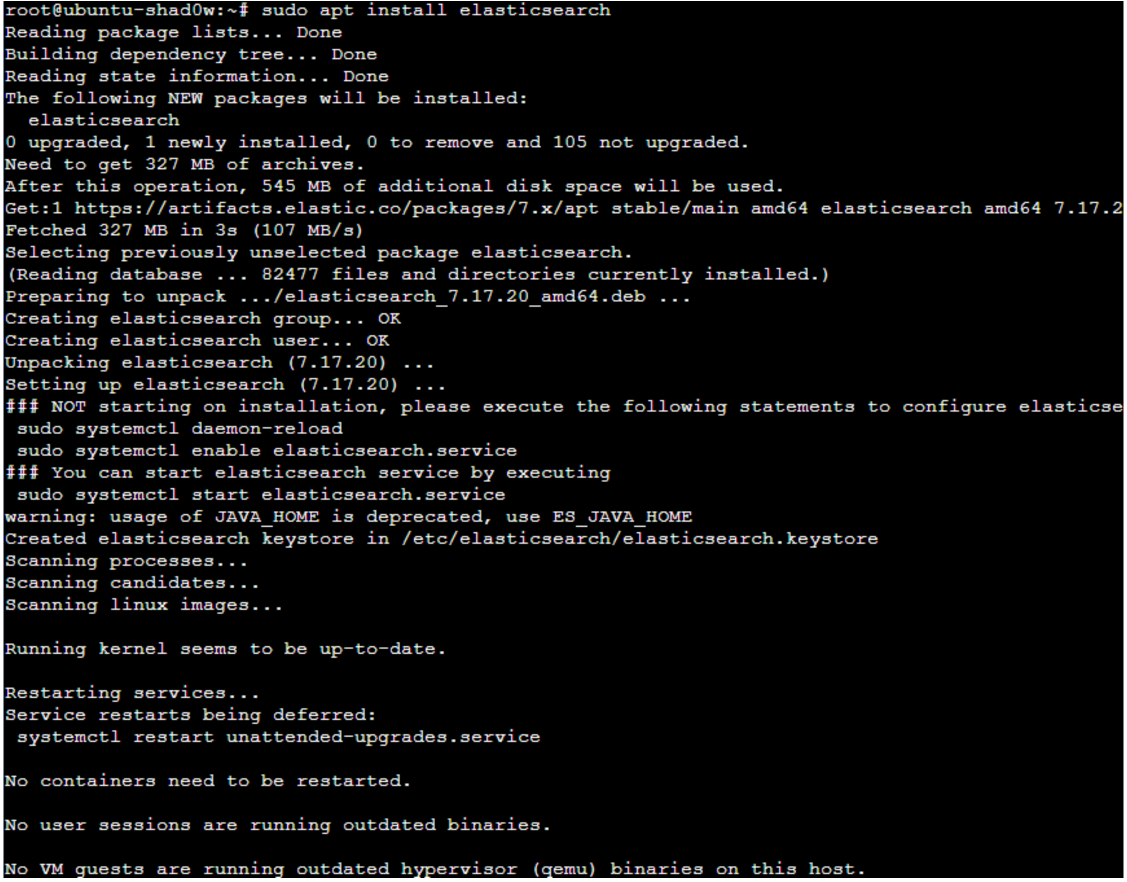
- Add the Elastic source list to the `sources.list.d` directory, where APT will search for new sources
  ```
  echo "deb [signed-by=/usr/share/keyrings/elastic.gpg]
  ```

```
https://artifacts.elastic.co/packages/7.x/apt stable main" | sudo tee -
a /etc/apt/sources.list.d/elastic-7.x.list
```

- Update your package lists so APT will read the new Elastic source
  `sudo apt update`

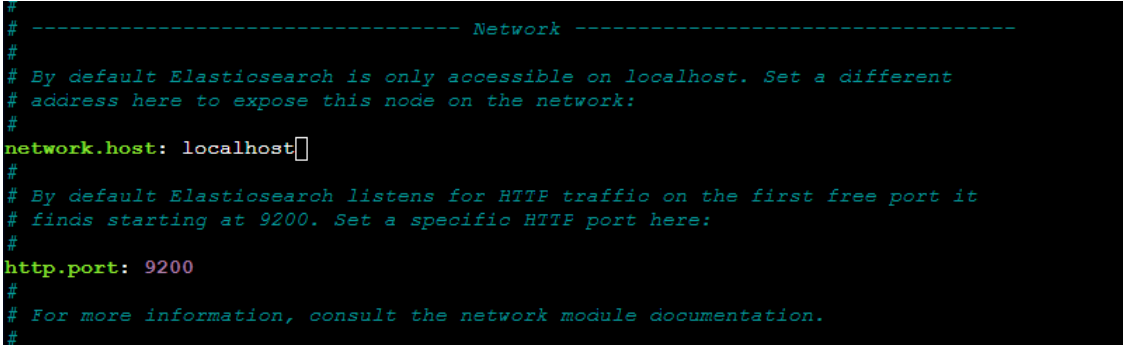
Installing & Configuring Elasticsearch

- Install Elasticsearch
  `sudo apt install elasticsearch`

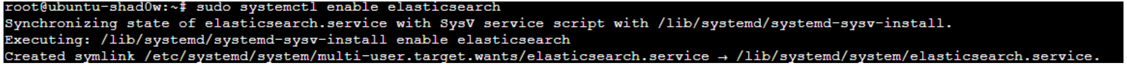


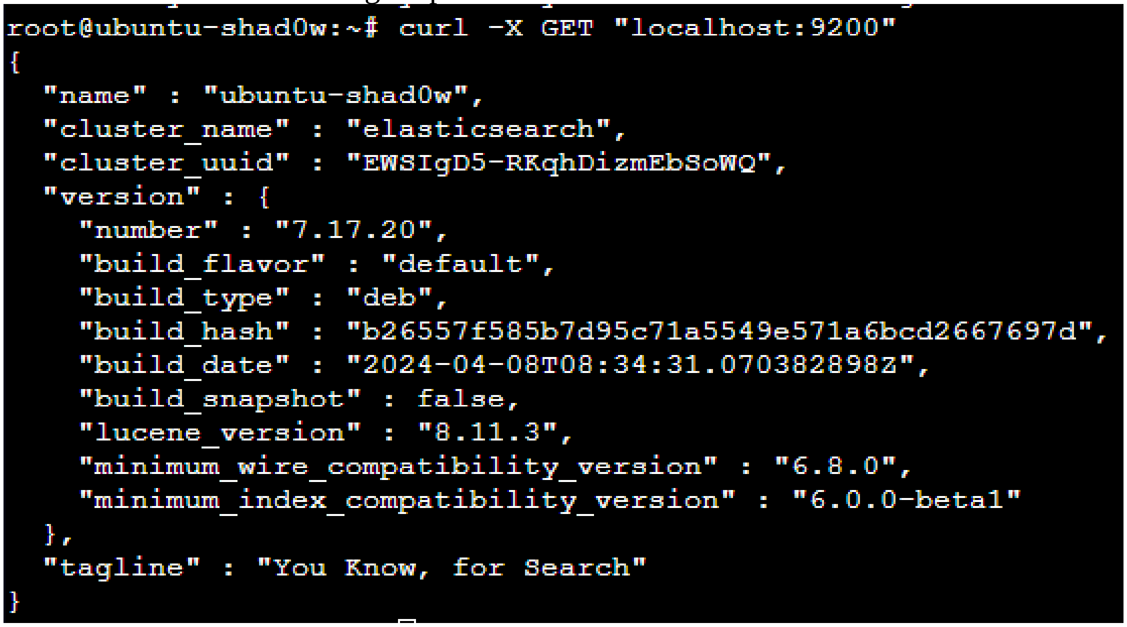
```
root@ubuntu-shad0w:~# sudo apt install elasticsearch
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  elasticsearch
0 upgraded, 1 newly installed, 0 to remove and 105 not upgraded.
Need to get 327 MB of archives.
After this operation, 545 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 elasticsearch amd64 7.17.2
Fetched 327 MB in 3s (107 MB/s)
Selecting previously unselected package elasticsearch.
(Reading database ... 82477 files and directories currently installed.)
Preparing to unpack .../elasticsearch_7.17.20_amd64.deb ...
Creating elasticsearch group... OK
Creating elasticsearch user... OK
Unpacking elasticsearch (7.17.20) ...
Setting up elasticsearch (7.17.20) ...
### NOT starting on installation, please execute the following statements to configure elasticse
 sudo systemctl daemon-reload
 sudo systemctl enable elasticsearch.service
### You can start elasticsearch service by executing
 sudo systemctl start elasticsearch.service
warning: usage of JAVA_HOME is deprecated, use ES_JAVA_HOME
Created elasticsearch keystore in /etc/elasticsearch/elasticsearch.keystore
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

Restarting services...
Service restarts being deferred:
 systemctl restart unattended-upgrades.service

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
```

- Configure Elasticsearch yaml file and change `network.host` as `localhost`
  `sudo nano /etc/elasticsearch/elasticsearch.yml`

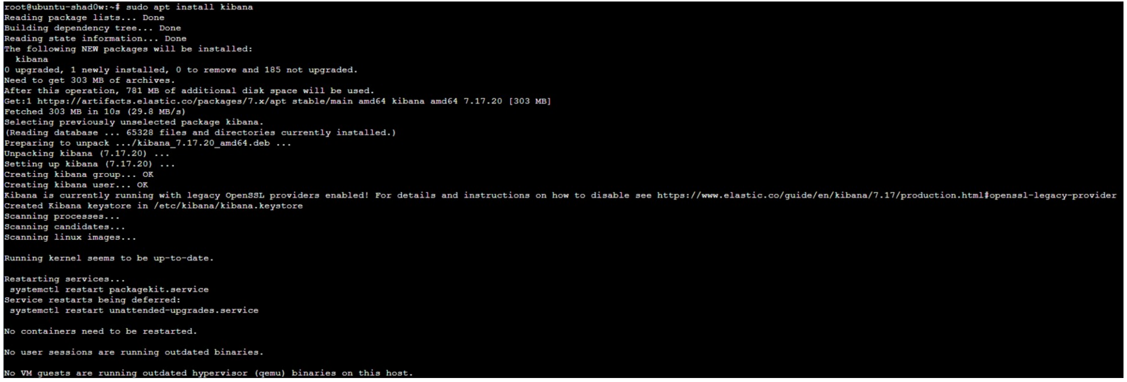


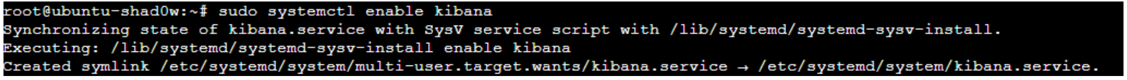
```
#
# ---------------------------------- Network -----------------------------------
#
# By default Elasticsearch is only accessible on localhost. Set a different
# address here to expose this node on the network:
#
network.host: localhost
#
# By default Elasticsearch listens for HTTP traffic on the first free port it
# finds starting at 9200. Set a specific HTTP port here:
#
http.port: 9200
#
# For more information, consult the network module documentation.
#
```

- Start and Enable the Elasticsearch service
  ```
  sudo systemctl start elasticsearch
  sudo systemctl enable elasticsearch
  ```
  
  ```
  root@ubuntu-shad0w:~# sudo systemctl enable elasticsearch
  Synchronizing state of elasticsearch.service with SysV service script with /lib/systemd/systemd-sysv-install.
  Executing: /lib/systemd/systemd-sysv-install enable elasticsearch
  Created symlink /etc/systemd/system/multi-user.target.wants/elasticsearch.service → /lib/systemd/system/elasticsearch.service.
  ```

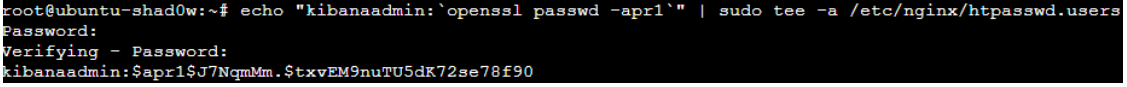
- Check if the Elasticsearch service is running by sending a HTTP request at port 9200.
  ```
  curl -X GET "localhost:9200"
  ```
  You should see the following output:
  
  ```
  root@ubuntu-shad0w:~# curl -X GET "localhost:9200"
  {
    "name" : "ubuntu-shad0w",
    "cluster_name" : "elasticsearch",
    "cluster_uuid" : "EWSIgD5-RKqhDizmEbSoWQ",
    "version" : {
      "number" : "7.17.20",
      "build_flavor" : "default",
      "build_type" : "deb",
      "build_hash" : "b26557f585b7d95c71a5549e571a6bcd2667697d",
      "build_date" : "2024-04-08T08:34:31.070382898Z",
      "build_snapshot" : false,
      "lucene_version" : "8.11.3",
      "minimum_wire_compatibility_version" : "6.8.0",
      "minimum_index_compatibility_version" : "6.0.0-beta1"
    },
    "tagline" : "You Know, for Search"
  }
  ```

Installing & Configuring Kibana

- Install Kibana
  ```
  sudo apt install kibana
  ```
  
  ```
  root@ubuntu-shad0w:~# sudo apt install kibana
  Reading package lists... Done
  Building dependency tree... Done
  Reading state information... Done
  The following NEW packages will be installed:
    kibana
  0 upgraded, 1 newly installed, 0 to remove and 185 not upgraded.
  Need to get 303 MB of archives.
  After this operation, 781 MB of additional disk space will be used.
  Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 kibana amd64 7.17.20 [303 MB]
  Fetched 303 MB in 10s (29.8 MB/s)
  Selecting previously unselected package kibana.
  (Reading database ... 65328 files and directories currently installed.)
  Preparing to unpack .../kibana_7.17.20_amd64.deb ...
  Unpacking kibana (7.17.20) ...
  Setting up kibana (7.17.20) ...
  Creating kibana group... OK
  Creating kibana user... OK
  Kibana is currently running with legacy OpenSSL providers enabled! For details and instructions on how to disable see https://www.elastic.co/guide/en/kibana/7.17/production.html#openssl-legacy-provider
  Created Kibana keystore in /etc/kibana/kibana.keystore
  Scanning processes...
  Scanning candidates...
  Scanning linux images...

  Running kernel seems to be up-to-date.

  Restarting services...
   systemctl restart packagekit.service
  Service restarts being deferred:
   systemctl restart unattended-upgrades.service

  No containers need to be restarted.

  No user sessions are running outdated binaries.

  No VM guests are running outdated hypervisor (qemu) binaries on this host.
  ```

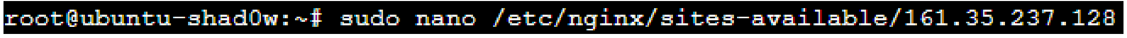
- Start and enable Kibana
  ```
  sudo systemctl start kibana
  sudo systemctl enable kibana
  ```
  
  ```
  root@ubuntu-shad0w:~# sudo systemctl enable kibana
  Synchronizing state of kibana.service with SysV service script with /lib/systemd/systemd-sysv-install.
  Executing: /lib/systemd/systemd-sysv-install enable kibana
  Created symlink /etc/systemd/system/multi-user.target.wants/kibana.service → /etc/systemd/system/kibana.service.
  ```

- Create temporary login credentials to safeguard the nginx service while installing and configuring the rest of the stack. Key in the password after entering the command. Be sure to set a strong password. After setting up minimal security on ELK, delete the configuration.
  'echo "kibanaadmin:`openssl passwd -apr1`" | sudo tee -a /etc/nginx/htpasswd.users'

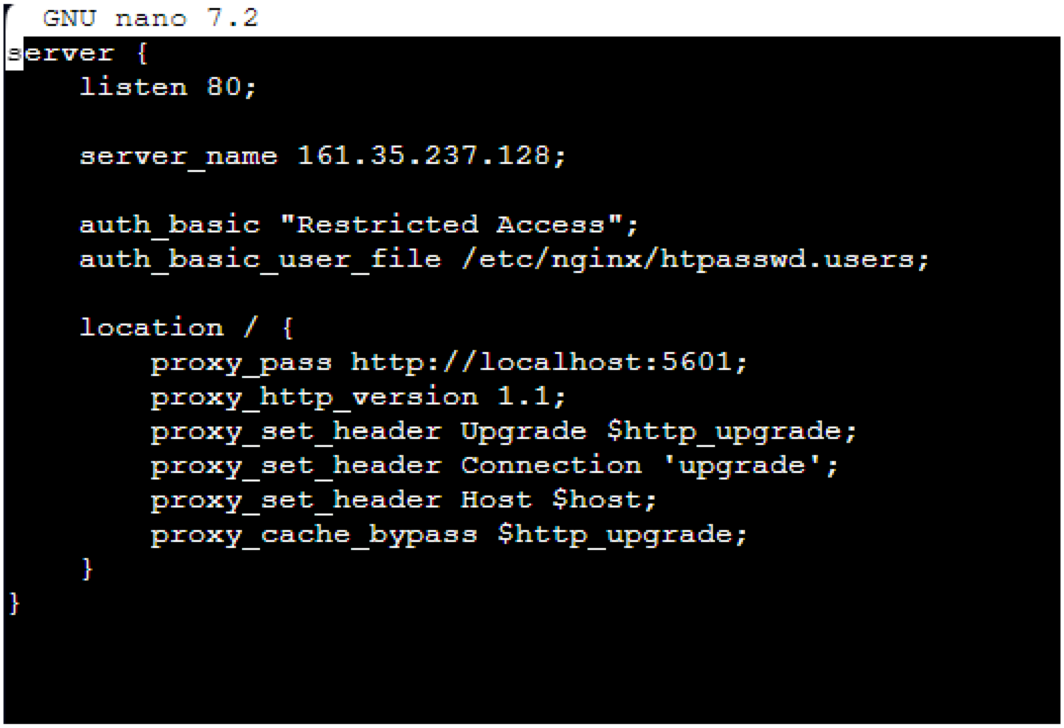


```
root@ubuntu-shad0w:~# echo "kibanaadmin:`openssl passwd -apr1`" | sudo tee -a /etc/nginx/htpasswd.users
Password:
Verifying - Password:
kibanaadmin:$apr1$J7NqmMm.$txvEM9nuTU5dK72se78f90
```

- Create a Nginx server block. Change `server_name` to your domain
  `sudo nano /etc/nginx/sites-available/<domain>`

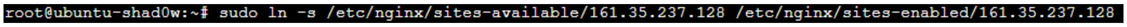


```
root@ubuntu-shad0w:~# sudo nano /etc/nginx/sites-available/161.35.237.128
```



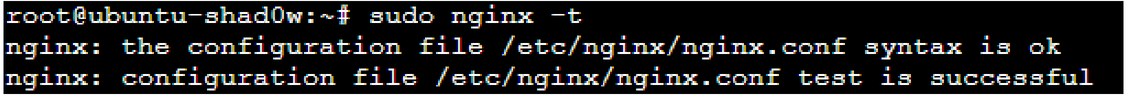
```
  GNU nano 7.2
server {
    listen 80;

    server_name 161.35.237.128;

    auth_basic "Restricted Access";
    auth_basic_user_file /etc/nginx/htpasswd.users;

    location / {
        proxy_pass http://localhost:5601;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

- Create a symbolic link (a shortcut) from the sites-available directory to the sites-enabled directory
  `sudo ln -s /etc/nginx/sites-available/<domain> /etc/nginx/sites-enabled/<domain>`



```
root@ubuntu-shad0w:~# sudo ln -s /etc/nginx/sites-available/161.35.237.128 /etc/nginx/sites-enabled/161.35.237.128
```

- Check the configuration for syntax errors
  `sudo nginx -t`



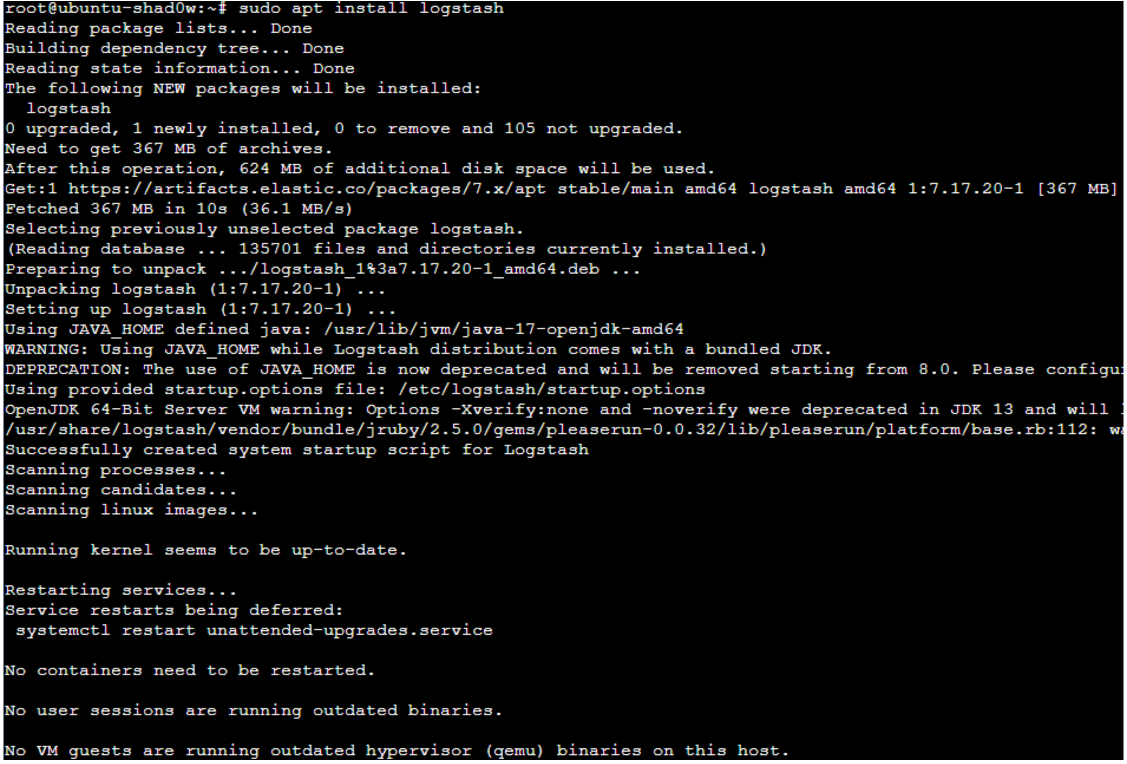
```
root@ubuntu-shad0w:~# sudo nginx -t
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok
nginx: configuration file /etc/nginx/nginx.conf test is successful
```

- Where there are no syntax errors, restart the nginx service
  `sudo systemctl restart nginx`

Installing & Configuring Logstash

- Installing Logstash

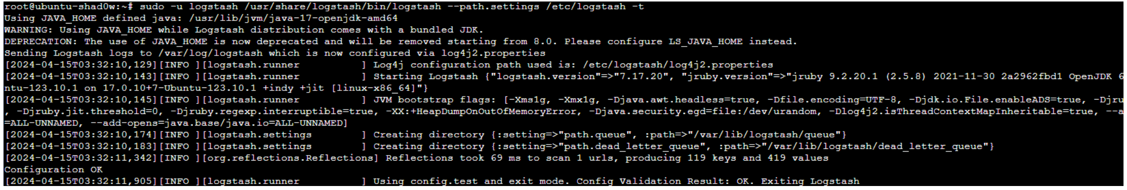
  ```
  sudo apt install logstash
  ```



```
root@ubuntu-shad0w:~# sudo apt install logstash
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  logstash
0 upgraded, 1 newly installed, 0 to remove and 105 not upgraded.
Need to get 367 MB of archives.
After this operation, 624 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 logstash amd64 1:7.17.20-1 [367 MB]
Fetched 367 MB in 10s (36.1 MB/s)
Selecting previously unselected package logstash.
(Reading database ... 135701 files and directories currently installed.)
Preparing to unpack .../logstash_1%3a7.17.20-1_amd64.deb ...
Unpacking logstash (1:7.17.20-1) ...
Setting up logstash (1:7.17.20-1) ...
Using JAVA_HOME defined java: /usr/lib/jvm/java-17-openjdk-amd64
WARNING: Using JAVA_HOME while Logstash distribution comes with a bundled JDK.
DEPRECATION: The use of JAVA_HOME is now deprecated and will be removed starting from 8.0. Please configu
Using provided startup.options file: /etc/logstash/startup.options
OpenJDK 64-Bit Server VM warning: Options -Xverify:none and -noverify were deprecated in JDK 13 and will
/usr/share/logstash/vendor/bundle/jruby/2.5.0/gems/pleaserun-0.0.32/lib/pleaserun/platform/base.rb:112: wa
Successfully created system startup script for Logstash
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.

Restarting services...
Service restarts being deferred:
 systemctl restart unattended-upgrades.service

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
```

- Testing Logstash connection. Output should display `Config Validation Result: OK. Exiting Logstash`

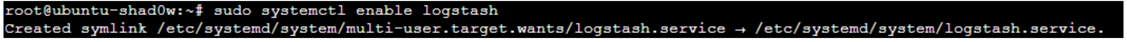
  ```
  sudo -u logstash /usr/share/logstash/bin/logstash --path.settings /etc/logstash -t
  ```



```
root@ubuntu-shad0w:~# sudo -u logstash /usr/share/logstash/bin/logstash --path.settings /etc/logstash -t
Using JAVA_HOME defined java: /usr/lib/jvm/java-17-openjdk-amd64
WARNING: Using JAVA_HOME while Logstash distribution comes with a bundled JDK.
DEPRECATION: The use of JAVA_HOME is now deprecated and will be removed starting from 8.0. Please configure LS_JAVA_HOME instead.
Sending Logstash logs to /var/log/logstash which is now configured via log4j2.properties
[2024-04-15T03:32:10,129][INFO ][logstash.runner          ] Log4j configuration path used is: /etc/logstash/log4j2.properties
[2024-04-15T03:32:10,143][INFO ][logstash.runner          ] Starting Logstash {"logstash.version"=>"7.17.20", "jruby.version"=>"jruby 9.2.20.1 (2.5.8) 2021-11-30 2a2962fbd1 OpenJDK 6
ntu-123.10.1 on 17.0.10+7-Ubuntu-123.10.1 +indy +jit [linux-x86_64]"}
[2024-04-15T03:32:10,145][INFO ][logstash.runner          ] JVM bootstrap flags: [-Xms1g, -Xmx1g, -Djava.awt.headless=true, -Dfile.encoding=UTF-8, -Djdk.io.File.enableADS=true, -Djru
, -Djruby.jit.threshold=0, -Djruby.regexp.interruptible=true, -XX:+HeapDumpOnOutOfMemoryError, -Djava.security.egd=file:/dev/urandom, -Dlog4j2.isThreadContextMapInheritable=true, --a
=ALL-UNNAMED, --add-opens=java.base/java.io=ALL-UNNAMED]
[2024-04-15T03:32:10,174][INFO ][logstash.settings        ] Creating directory {:setting=>"path.queue", :path=>"/var/lib/logstash/queue"}
[2024-04-15T03:32:10,183][INFO ][logstash.settings        ] Creating directory {:setting=>"path.dead_letter_queue", :path=>"/var/lib/logstash/dead_letter_queue"}
[2024-04-15T03:32:11,342][INFO ][org.reflections.Reflections] Reflections took 69 ms to scan 1 urls, producing 119 keys and 419 values
Configuration OK
[2024-04-15T03:32:11,905][INFO ][logstash.runner          ] Using config.test and exit mode. Config Validation Result: OK. Exiting Logstash
```

- Starting and enabling Logstash

  ```
  sudo systemctl start logstash
  sudo systemctl enable logstash
  ```

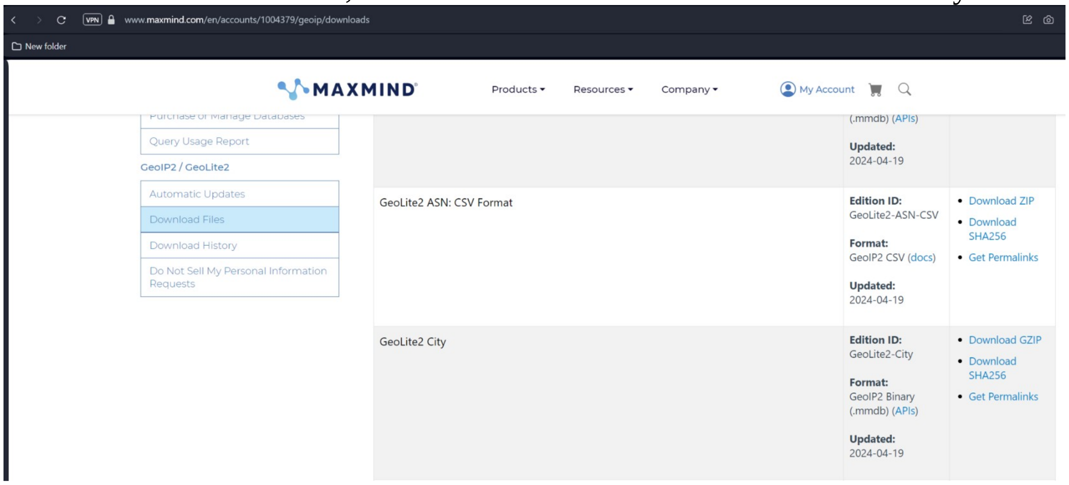


```
root@ubuntu-shad0w:~# sudo systemctl enable logstash
Created symlink /etc/systemd/system/multi-user.target.wants/logstash.service → /etc/systemd/system/logstash.service.
```

Enrich Cowrie logs with GeoIP information

- Enriching cowrie logs provide approximate geographic location of IP addresses based on the MAXMIND database

- Go to www.maxmind.com, create an account and download 'GeoLite2 City' GZIP file

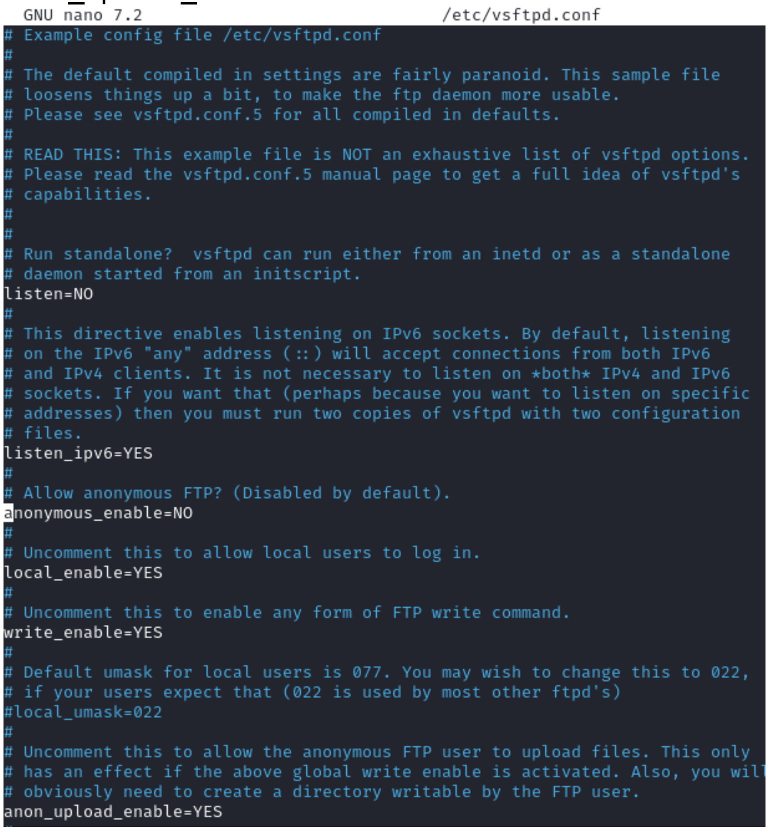


www.maxmind.com/en/accounts/1004379/geoip/downloads

	(.mmdb) (APIs) **Updated:** 2024-04-19	
GeoLite2 ASN: CSV Format	**Edition ID:** GeoLite2-ASN-CSV **Format:** GeoIP2 CSV (docs) **Updated:** 2024-04-19	• Download ZIP • Download SHA256 • Get Permalinks
GeoLite2 City	**Edition ID:** GeoLite2-City **Format:** GeoIP2 Binary (.mmdb) (APIs) **Updated:** 2024-04-19	• Download GZIP • Download SHA256 • Get Permalinks

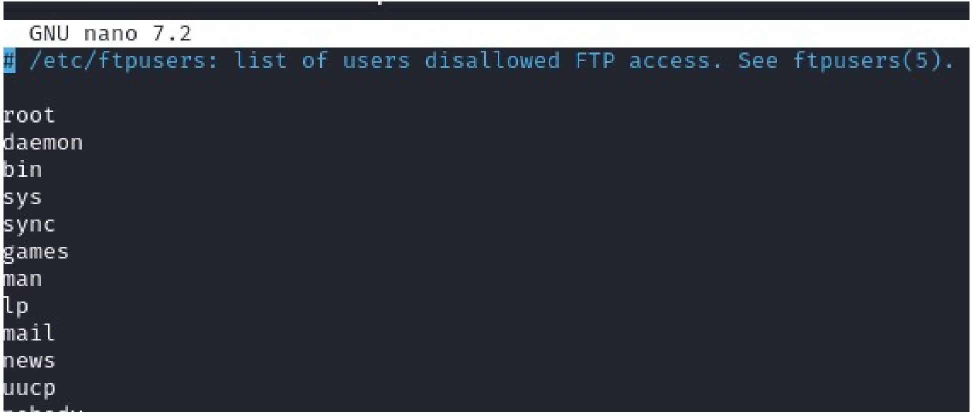
- You can choose any methods to transfer the file into the machine. For this demonstration, the `ftp` service will be used here.
  `nano /etc/vsftpd.conf`

- Change the following lines to: `local_enable=YES`, `write_enable=YES`, `anon_upload_enable=YES`



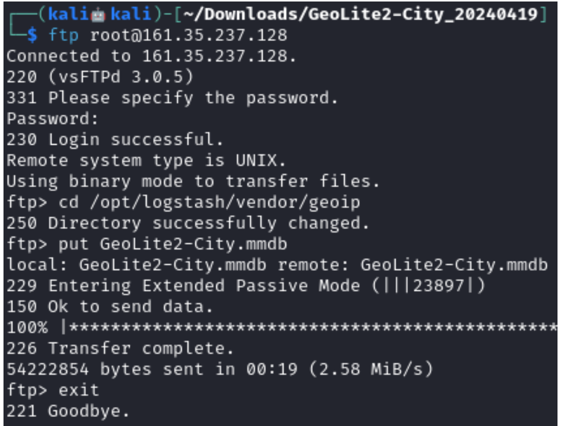
```
  GNU nano 7.2                          /etc/vsftpd.conf
# Example config file /etc/vsftpd.conf
#
# The default compiled in settings are fairly paranoid. This sample file
# loosens things up a bit, to make the ftp daemon more usable.
# Please see vsftpd.conf.5 for all compiled in defaults.
#
# READ THIS: This example file is NOT an exhaustive list of vsftpd options.
# Please read the vsftpd.conf.5 manual page to get a full idea of vsftpd's
# capabilities.
#
#
# Run standalone?  vsftpd can run either from an inetd or as a standalone
# daemon started from an initscript.
listen=NO
#
# This directive enables listening on IPv6 sockets. By default, listening
# on the IPv6 "any" address (::) will accept connections from both IPv6
# and IPv4 clients. It is not necessary to listen on *both* IPv4 and IPv6
# sockets. If you want that (perhaps because you want to listen on specific
# addresses) then you must run two copies of vsftpd with two configuration
# files.
listen_ipv6=YES
#
# Allow anonymous FTP? (Disabled by default).
anonymous_enable=NO
#
# Uncomment this to allow local users to log in.
local_enable=YES
#
# Uncomment this to enable any form of FTP write command.
write_enable=YES
#
# Default umask for local users is 077. You may wish to change this to 022,
# if your users expect that (022 is used by most other ftpd's)
#local_umask=022
#
# Uncomment this to allow the anonymous FTP user to upload files. This only
# has an effect if the above global write enable is activated. Also, you will
# obviously need to create a directory writable by the FTP user.
anon_upload_enable=YES
```

- If you are using logging as as a root user, you should remove `root` from the ftpusers list
  `nano /etc/ftpusers`

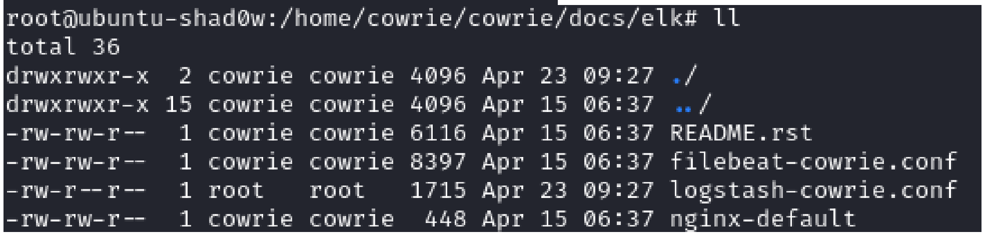


```
  GNU nano 7.2
# /etc/ftpusers: list of users disallowed FTP access. See ftpusers(5).

root
daemon
bin
sys
sync
games
man
lp
mail
news
uucp
```

- Start the ftp server and ftp into it
  `sudo systemctl start vsftpd`
  `ftp <user>@<ip_address>`



```
┌──(kali㉿kali)-[~/Downloads/GeoLite2-City_20240419]
└─$ ftp root@161.35.237.128
Connected to 161.35.237.128.
220 (vsFTPd 3.0.5)
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd /opt/logstash/vendor/geoip
250 Directory successfully changed.
ftp> put GeoLite2-City.mmdb
local: GeoLite2-City.mmdb remote: GeoLite2-City.mmdb
229 Entering Extended Passive Mode (|||23897|)
150 Ok to send data.
100% |***************************************************
226 Transfer complete.
54222854 bytes sent in 00:19 (2.58 MiB/s)
ftp> exit
221 Goodbye.
```



```
root@ubuntu-shad0w:/home/cowrie/cowrie/docs/elk# ll
total 36
drwxrwxr-x  2 cowrie cowrie 4096 Apr 23 09:27 ./
drwxrwxr-x 15 cowrie cowrie 4096 Apr 15 06:37 ../
-rw-rw-r--  1 cowrie cowrie 6116 Apr 15 06:37 README.rst
-rw-rw-r--  1 cowrie cowrie 8397 Apr 15 06:37 filebeat-cowrie.conf
-rw-r--r--  1 root   root   1715 Apr 23 09:27 logstash-cowrie.conf
-rw-rw-r--  1 cowrie cowrie  448 Apr 15 06:37 nginx-default
```

- Remember to reverse all of the previous steps and turn off the ftp service:
  `sudo systemctl stop vsftpd`

- Create the file path `/opt/logstash/vendor/geoip/` and move the `GeoLite2-City.mmdb`file to the created file path. Then, chmod the file so that logstash has the permission to 'execute' it

  ```
  sudo mkdir-p /opt/logstash/vendor/geoip/

  sudo mv GeoLite2-City.mmdb /opt/logstash/vendor/geoip

  sudo chmod 777 /opt/logstash/vendor/geoip/GeoLite2-City.mmdb
  ```

Configuring Logstash.yml

- Copy and paste the following code into the logstash.yml file:
  `nano /etc/logstash/conf.d/logstash-cowrie.conf`

```
input {
       # filebeats
       beats {
             port => 5044
             type => "cowrie"
       }

       # if you don't want to use filebeat: this is the actual live log file to monitor
       #file {
       #       path => ["/home/cowrie/cowrie-git/log/cowrie.json"]
       #       codec => json
       #       type => "cowrie"
       #}
}

filter {
    if [type] == "cowrie" {
        json {
        source => message
            target => honeypot
    }

        date {
            match => [ "timestamp", "ISO8601" ]
        }

        if [honeypot][src_ip]  {

            mutate {
                add_field => { "src_host" => "%{[honeypot][src_ip]}" }
            }

            dns {
                reverse => [ "src_host" ]
                nameserver => [ "8.8.8.8", "8.8.4.4" ]
                action => "replace"
                hit_cache_size => 4096
                hit_cache_ttl => 900
                failed_cache_size => 512
                failed_cache_ttl => 900
            }


            geoip {
                source => "[honeypot][src_ip]"
                target => "geoip"
                database => "/opt/logstash/vendor/geoip/GeoLite2-City.mmdb"
            }

        }

        mutate {
        # cut out useless tags/fields
            remove_tag => [ "beats_input_codec_plain_applied"]
        remove_field => [ "[log][file][path]", "[log][offset]" ]
        }
    }
}

output {
    if [type] == "cowrie" {
        elasticsearch {
            hosts => ["localhost:9200"]
        ilm_enabled => auto
        ilm_rollover_alias => "cowrie-logstash"
         # uncomment user and password here to set up minimal security
           # user => "elastic"
           # password => "<password>"
        }
        file {
            path => "/tmp/cowrie-logstash.log"
            codec => json
        }
        stdout {
            codec => rubydebug
        }
    }
}
```

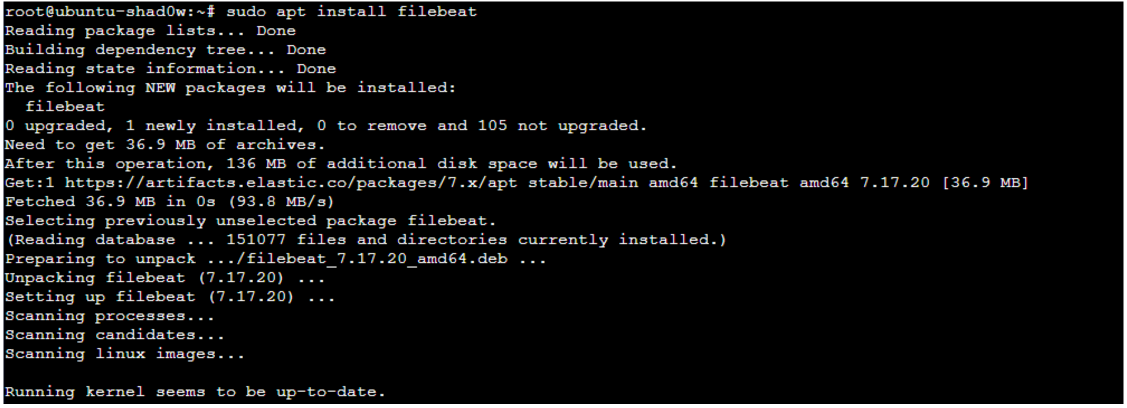
IMPORTANT NOTE

The code can change over time as the honeypot continues to develop. To get the latest configuration, visit the github page. Do note that at the time of installation, the configuration has been modified as the original config file had issues.

- Restart logstash
  `sudo systemctl start logstash`

Installing & Configuring Filebeat

- Installing Filebeat
  `sudo apt install filebeat`



```
root@ubuntu-shad0w:~# sudo apt install filebeat
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  filebeat
0 upgraded, 1 newly installed, 0 to remove and 105 not upgraded.
Need to get 36.9 MB of archives.
After this operation, 136 MB of additional disk space will be used.
Get:1 https://artifacts.elastic.co/packages/7.x/apt stable/main amd64 filebeat amd64 7.17.20 [36.9 MB]
Fetched 36.9 MB in 0s (93.8 MB/s)
Selecting previously unselected package filebeat.
(Reading database ... 151077 files and directories currently installed.)
Preparing to unpack .../filebeat_7.17.20_amd64.deb ...
Unpacking filebeat (7.17.20) ...
Setting up filebeat (7.17.20) ...
Scanning processes...
Scanning candidates...
Scanning linux images...

Running kernel seems to be up-to-date.
```

- Copy the following cowrie filebeat configuration into filebeat:
  `nano /etc/filebeat/filebeat.yml`

```
###################### Filebeat Configuration Example #########################

# This file is an example configuration file highlighting only the most common
# options. The filebeat.reference.yml file from the same directory contains all the
# supported options with more comments. You can use it as a reference.
#
# You can find the full configuration reference here:
# https://www.elastic.co/guide/en/beats/filebeat/index.html

# For more available modules and options, please see the filebeat.reference.yml sample
# configuration file.

#=========================== Filebeat inputs =============================

filebeat.inputs:

# Each - is an input. Most options can be set at the input level, so
# you can use different inputs for various configurations.
# Below are the input specific configurations.

- type: log

  # Change to true to enable this input configuration.
  enabled: true

  # Paths that should be crawled and fetched. Glob based paths.
  paths:
    - /home/cowrie/cowrie/var/log/cowrie/cowrie.json*
    #- c:\programdata\elasticsearch\logs\*

  # Exclude lines. A list of regular expressions to match. It drops the lines that are
  # matching any regular expression from the list.
  #exclude_lines: ['^DBG']

  # Include lines. A list of regular expressions to match. It exports the lines that are
  # matching any regular expression from the list.
  #include_lines: ['^ERR', '^WARN']

  # Exclude files. A list of regular expressions to match. Filebeat drops the files that
  # are matching any regular expression from the list. By default, no files are dropped.
  #exclude_files: ['.gz$']

  # Optional additional fields. These fields can be freely picked
  # to add additional information to the crawled log files for filtering
  #fields:
  #  level: debug
  #  review: 1
  ### Multiline options

  # Multiline can be used for log messages spanning multiple lines. This is common
  # for Java Stack Traces or C-Line Continuation

  # The regexp Pattern that has to be matched. The example pattern matches all lines starting with [
  #multiline.pattern: ^\[

  # Defines if the pattern set under pattern should be negated or not. Default is false.
  #multiline.negate: false

  # Match can be set to "after" or "before". It is used to define if lines should be append to a pattern
  # that was (not) matched before or after or as long as a pattern is not matched based on negate.
  # Note: After is the equivalent to previous and before is the equivalent to to next in Logstash
  #multiline.match: after


#============================= Filebeat modules ===============================

filebeat.config.modules:
  # Glob pattern for configuration loading
  path: ${path.config}/modules.d/*.yml

  # Set to true to enable config reloading
  reload.enabled: false

  # Period on which files under path should be checked for changes
  #reload.period: 10s

#==================== Elasticsearch template setting ==========================

setup.template.settings:
  index.number_of_shards: 1
  #index.codec: best_compression
  #_source.enabled: false

#================================ General =====================================

# The name of the shipper that publishes the network data. It can be used to group
# all the transactions sent by a single shipper in the web interface.
#name:

# The tags of the shipper are included in their own field with each
# transaction published.
#tags: ["service-X", "web-tier"]

# Optional fields that you can specify to add additional information to the
# output.
#fields:
#  env: staging


#============================== Dashboards =====================================
# These settings control loading the sample dashboards to the Kibana index. Loading
# the dashboards is disabled by default and can be enabled either by setting the
# options here or by using the `setup` command.
#setup.dashboards.enabled: false

# The URL from where to download the dashboards archive. By default this URL
# has a value which is computed based on the Beat name and version. For released
# versions, this URL points to the dashboard archive on the artifacts.elastic.co
# website.
#setup.dashboards.url:

#============================== Kibana =====================================

# Starting with Beats version 6.0.0, the dashboards are loaded via the Kibana API.
# This requires a Kibana endpoint configuration.
setup.kibana:

  # Kibana Host
  # Scheme and port can be left out and will be set to the default (http and 5601)
  # In case you specify and additional path, the scheme is required: http://localhost:5601/path
  # IPv6 addresses should always be defined as: https://[2001:db8::1]:5601
  #host: "localhost:5601"

  # Kibana Space ID
  # ID of the Kibana Space into which the dashboards should be loaded. By default,
  # the Default Space will be used.
  #space.id:

#============================= Elastic Cloud ==================================

# These settings simplify using Filebeat with the Elastic Cloud (https://cloud.elastic.co/).

# The cloud.id setting overwrites the `output.elasticsearch.hosts` and
# `setup.kibana.host` options.
# You can find the `cloud.id` in the Elastic Cloud web UI.
#cloud.id:

# The cloud.auth setting overwrites the `output.elasticsearch.username` and
# `output.elasticsearch.password` settings. The format is `<user>:<pass>`.
#cloud.auth:

#================================ Outputs =====================================

# Configure what output to use when sending the data collected by the beat.

#-------------------------- Elasticsearch output ------------------------------
output.elasticsearch:
  # Array of hosts to connect to.
  enabled: false
  #hosts: ["localhost:9200"]

  # Protocol - either `http` (default) or `https`.
  #protocol: "https"

  # Authentication credentials - either API key or username/password.
  # uncomment to set up minimal security for elasticsearch
  #api_key: "id:api_key"
  #username: "elastic"
  #password: "<password>"

#----------------------------- Logstash output --------------------------------
output.logstash:
  enabled: true
  # The Logstash hosts
  hosts: ["localhost:5044"]

  # Optional SSL. By default is off.
  # List of root certificates for HTTPS server verifications
  #ssl.certificate_authorities: ["/etc/pki/root/ca.pem"]

  # Certificate for SSL client authentication
  #ssl.certificate: "/etc/pki/client/cert.pem"

  # Client Certificate Key
  #ssl.key: "/etc/pki/client/cert.key"

#================================ Processors =====================================

# Configure processors to enhance or manipulate events generated by the beat.

processors:
  - add_host_metadata: ~
  - add_cloud_metadata: ~
  - add_docker_metadata: ~
  - add_kubernetes_metadata: ~

#================================ Logging =====================================

# Sets log level. The default log level is info.
# Available log levels are: error, warning, info, debug
#logging.level: debug

# At debug level, you can selectively enable logging only for some components.
# To enable all selectors use ["*"]. Examples of other selectors are "beat",
# "publish", "service".
#logging.selectors: ["*"]

#============================== X-Pack Monitoring ===============================
# filebeat can export internal metrics to a central Elasticsearch monitoring
# cluster.  This requires xpack monitoring to be enabled in Elasticsearch.  The
# reporting is disabled by default.

# Set to true to enable the monitoring reporter.
#monitoring.enabled: false

# Sets the UUID of the Elasticsearch cluster under which monitoring data for this
# Filebeat instance will appear in the Stack Monitoring UI. If output.elasticsearch
# is enabled, the UUID is derived from the Elasticsearch cluster referenced by output.elasticsearch.
#monitoring.cluster_uuid:

# Uncomment to send the metrics to Elasticsearch. Most settings from the
# Elasticsearch output are accepted here as well.
# Note that the settings should point to your Elasticsearch *monitoring* cluster.
# Any setting that is not set is automatically inherited from the Elasticsearch
# output configuration, so if you have the Elasticsearch output configured such
# that it is pointing to your Elasticsearch monitoring cluster, you can simply
# uncomment the following line.
#monitoring.elasticsearch:

#================================= Migration ==================================

# This allows to enable 6.7 migration aliases
#migration.6_to_7.enabled: true
```

IMPORTANT NOTE

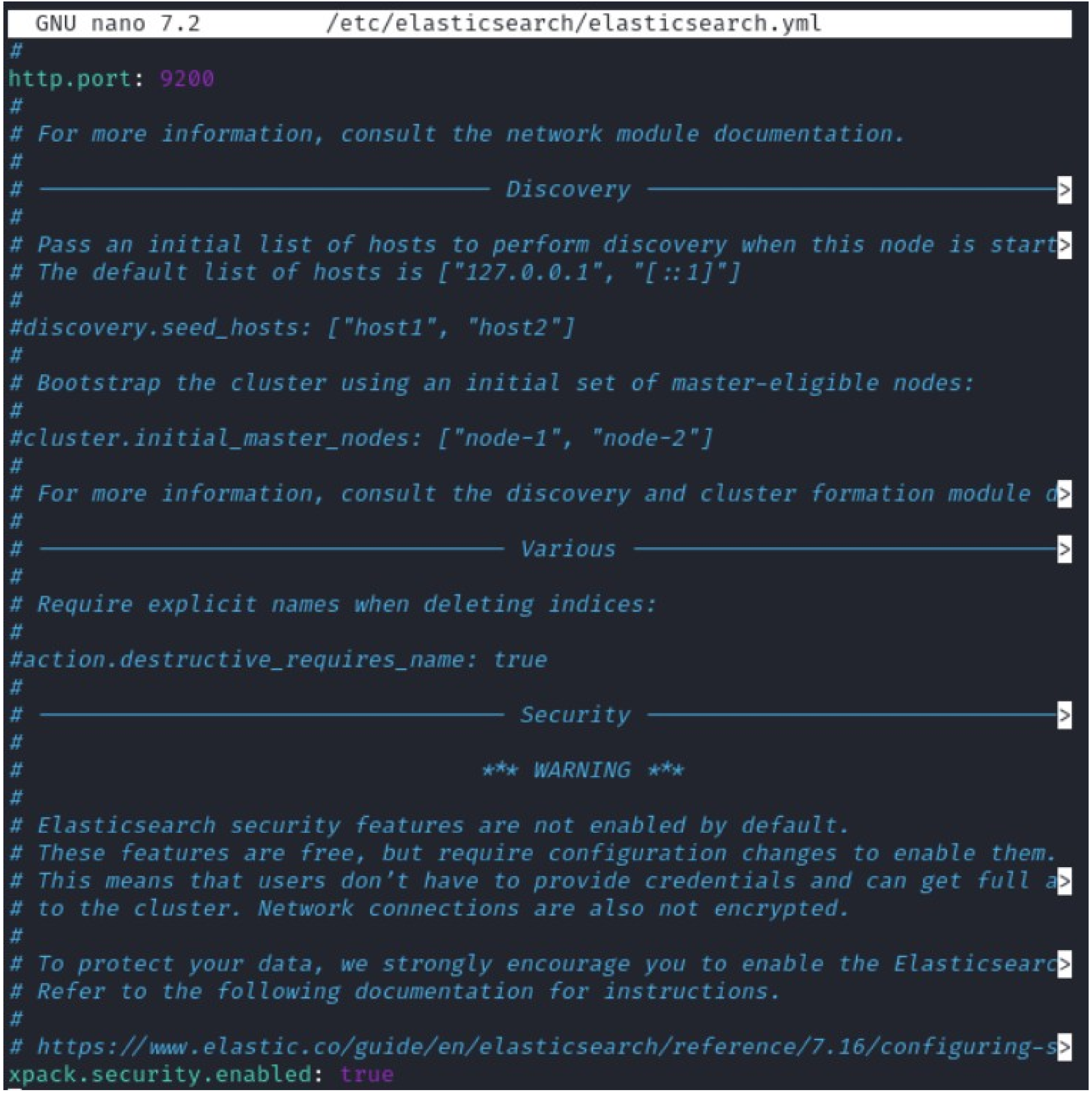
The code can change over time as the honeypot continues to develop. To get the latest configuration, visit the github page. Do note that at the time of installation, the configuration has been modified as the original config file had issues.

- Start and enable filebeat
  ```
  sudo systemctl start filebeat
  sudo systemctl enable filebeat
  ```

- Filebeat commands not necessary for the installation but good to know:
  ```
  sudo filebeat modules list
  sudo filebeat modules enable system
  sudo filebeat setup --pipelines --modules system
  ```

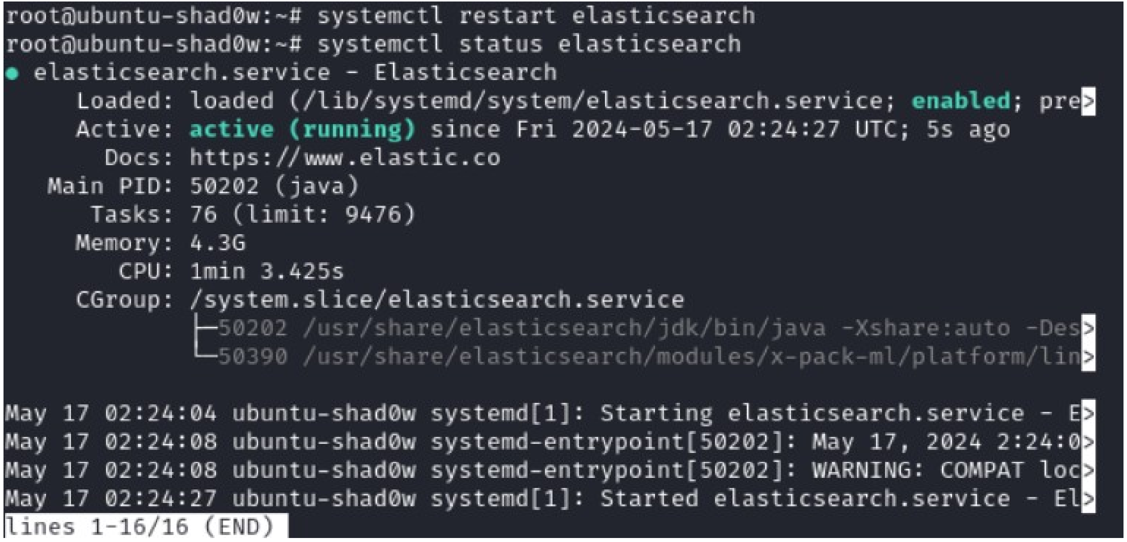
Creating Minimal Security in Elasticsearch

- Edit elasticsearch.yml and add `xpack.security.enabled: true` at the last line
  ```
  nano /etc/elasticsearch/elasticsearch.yml
  ```



```
  GNU nano 7.2          /etc/elasticsearch/elasticsearch.yml
#
http.port: 9200
#
# For more information, consult the network module documentation.
#
# ---------------------------------- Discovery -------------------------------->
#
# Pass an initial list of hosts to perform discovery when this node is start>
# The default list of hosts is ["127.0.0.1", "[::1]"]
#
#discovery.seed_hosts: ["host1", "host2"]
#
# Bootstrap the cluster using an initial set of master-eligible nodes:
#
#cluster.initial_master_nodes: ["node-1", "node-2"]
#
# For more information, consult the discovery and cluster formation module d>
#
# ---------------------------------- Various ---------------------------------->
#
# Require explicit names when deleting indices:
#
#action.destructive_requires_name: true
#
# ---------------------------------- Security --------------------------------->
#
#                                 *** WARNING ***
#
# Elasticsearch security features are not enabled by default.
# These features are free, but require configuration changes to enable them.
# This means that users don't have to provide credentials and can get full a>
# to the cluster. Network connections are also not encrypted.
#
# To protect your data, we strongly encourage you to enable the Elasticsearc>
# Refer to the following documentation for instructions.
#
# https://www.elastic.co/guide/en/elasticsearch/reference/7.16/configuring-s>
xpack.security.enabled: true
```

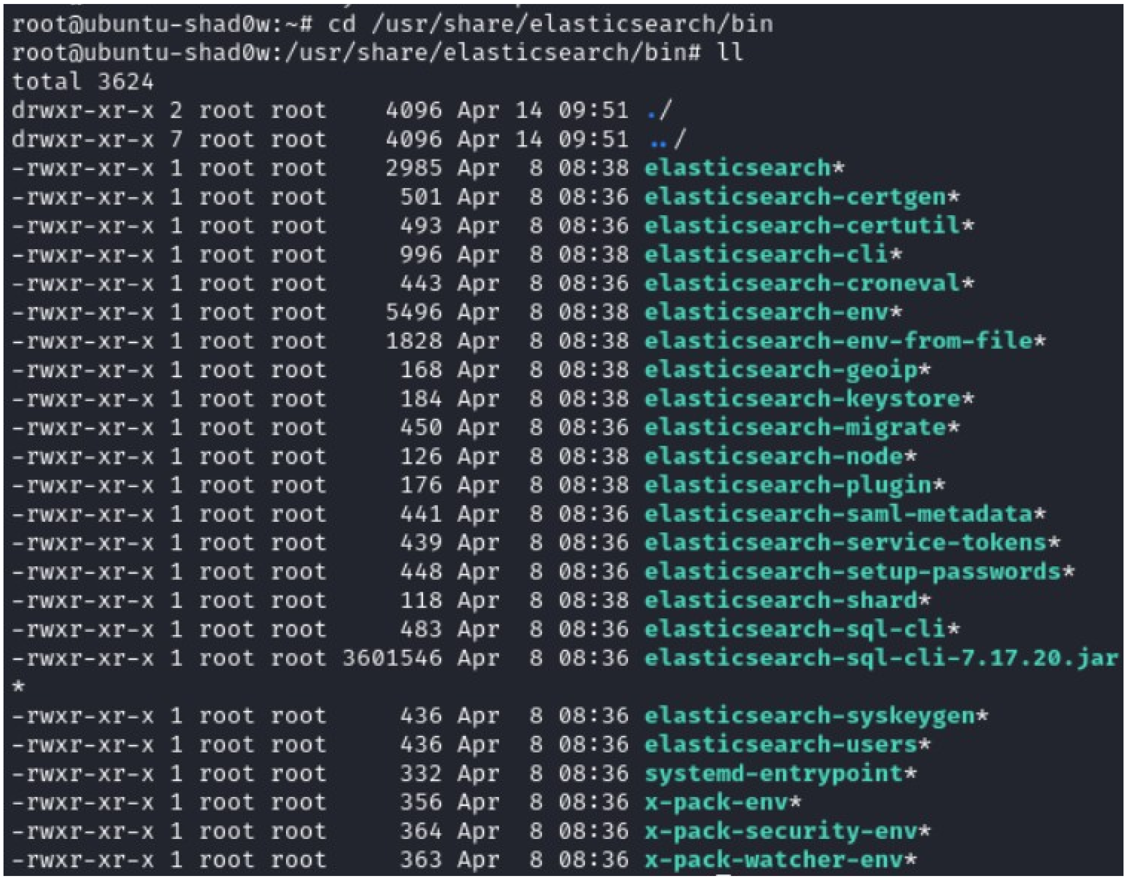
- Restart elasticsearch
  `systemctl restart elasticsearch`



```
root@ubuntu-shad0w:~# systemctl restart elasticsearch
root@ubuntu-shad0w:~# systemctl status elasticsearch
● elasticsearch.service - Elasticsearch
     Loaded: loaded (/lib/systemd/system/elasticsearch.service; enabled; pre>
     Active: active (running) since Fri 2024-05-17 02:24:27 UTC; 5s ago
       Docs: https://www.elastic.co
   Main PID: 50202 (java)
      Tasks: 76 (limit: 9476)
     Memory: 4.3G
        CPU: 1min 3.425s
     CGroup: /system.slice/elasticsearch.service
             ├─50202 /usr/share/elasticsearch/jdk/bin/java -Xshare:auto -Des>
             └─50390 /usr/share/elasticsearch/modules/x-pack-ml/platform/lin>

May 17 02:24:04 ubuntu-shad0w systemd[1]: Starting elasticsearch.service - E>
May 17 02:24:08 ubuntu-shad0w systemd-entrypoint[50202]: May 17, 2024 2:24:0>
May 17 02:24:08 ubuntu-shad0w systemd-entrypoint[50202]: WARNING: COMPAT loc>
May 17 02:24:27 ubuntu-shad0w systemd[1]: Started elasticsearch.service - El>
lines 1-16/16 (END)
```

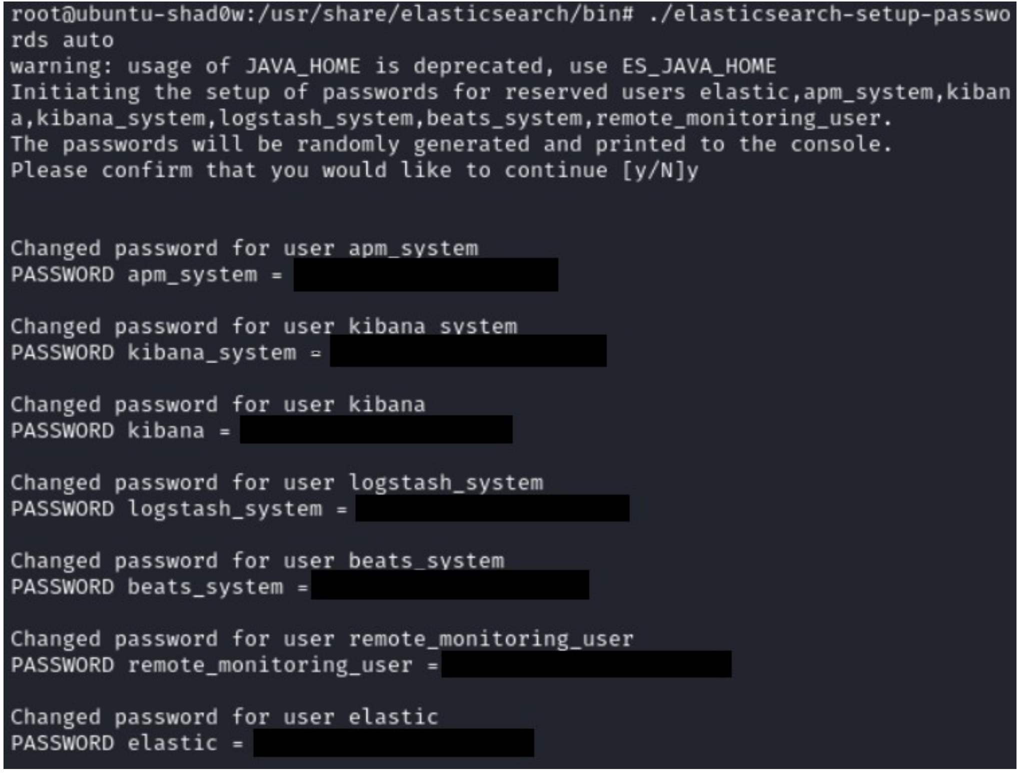
- Stop the following services:
  `stop kibana filebeat logstash`

- Change directory into elasticsearch/bin
  `cd /usr/share/elasticsearch/bin`



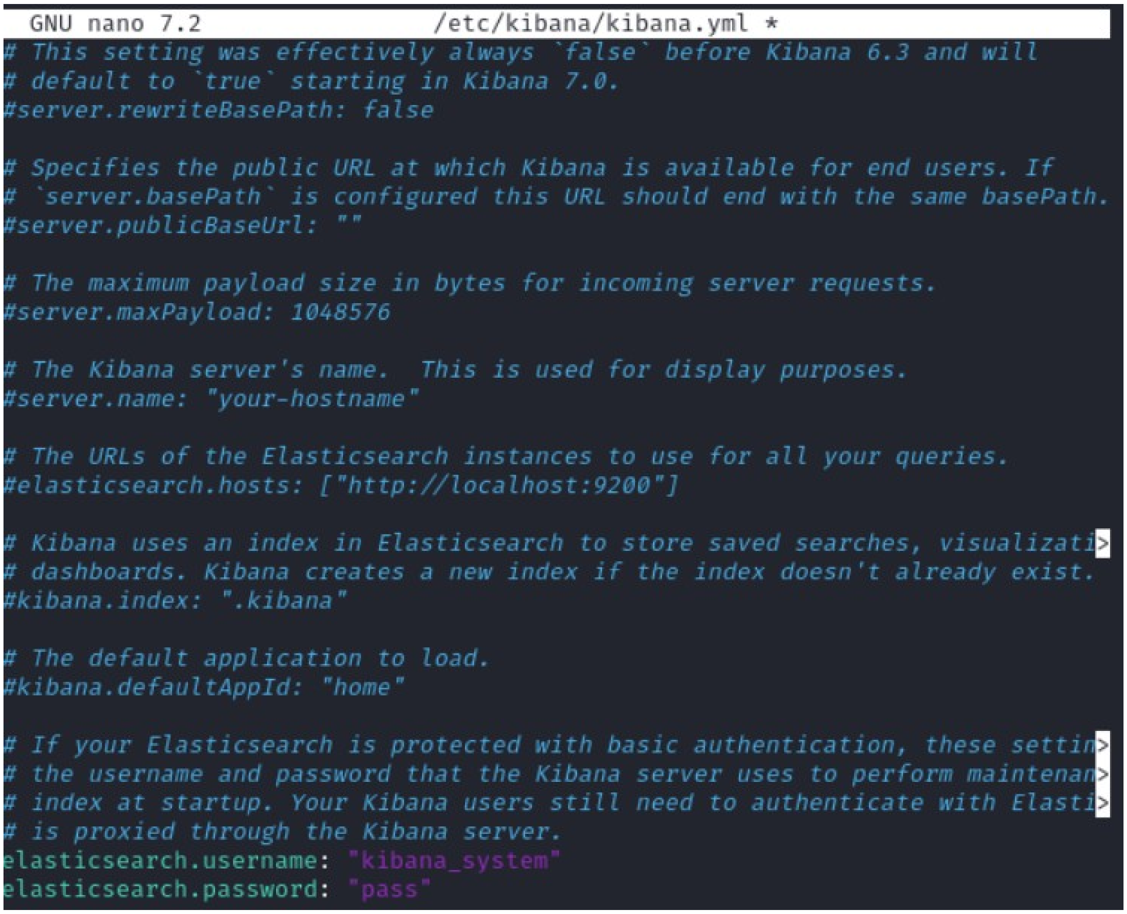
```
root@ubuntu-shad0w:~# cd /usr/share/elasticsearch/bin
root@ubuntu-shad0w:/usr/share/elasticsearch/bin# ll
total 3624
drwxr-xr-x 2 root root    4096 Apr 14 09:51 ./
drwxr-xr-x 7 root root    4096 Apr 14 09:51 ../
-rwxr-xr-x 1 root root    2985 Apr  8 08:38 elasticsearch*
-rwxr-xr-x 1 root root     501 Apr  8 08:36 elasticsearch-certgen*
-rwxr-xr-x 1 root root     493 Apr  8 08:36 elasticsearch-certutil*
-rwxr-xr-x 1 root root     996 Apr  8 08:38 elasticsearch-cli*
-rwxr-xr-x 1 root root     443 Apr  8 08:36 elasticsearch-croneval*
-rwxr-xr-x 1 root root    5496 Apr  8 08:38 elasticsearch-env*
-rwxr-xr-x 1 root root    1828 Apr  8 08:38 elasticsearch-env-from-file*
-rwxr-xr-x 1 root root     168 Apr  8 08:38 elasticsearch-geoip*
-rwxr-xr-x 1 root root     184 Apr  8 08:38 elasticsearch-keystore*
-rwxr-xr-x 1 root root     450 Apr  8 08:36 elasticsearch-migrate*
-rwxr-xr-x 1 root root     126 Apr  8 08:38 elasticsearch-node*
-rwxr-xr-x 1 root root     176 Apr  8 08:38 elasticsearch-plugin*
-rwxr-xr-x 1 root root     441 Apr  8 08:36 elasticsearch-saml-metadata*
-rwxr-xr-x 1 root root     439 Apr  8 08:36 elasticsearch-service-tokens*
-rwxr-xr-x 1 root root     448 Apr  8 08:36 elasticsearch-setup-passwords*
-rwxr-xr-x 1 root root     118 Apr  8 08:38 elasticsearch-shard*
-rwxr-xr-x 1 root root     483 Apr  8 08:36 elasticsearch-sql-cli*
-rwxr-xr-x 1 root root 3601546 Apr  8 08:36 elasticsearch-sql-cli-7.17.20.jar
*
-rwxr-xr-x 1 root root     436 Apr  8 08:36 elasticsearch-syskeygen*
-rwxr-xr-x 1 root root     436 Apr  8 08:36 elasticsearch-users*
-rwxr-xr-x 1 root root     332 Apr  8 08:36 systemd-entrypoint*
-rwxr-xr-x 1 root root     356 Apr  8 08:36 x-pack-env*
-rwxr-xr-x 1 root root     364 Apr  8 08:36 x-pack-security-env*
-rwxr-xr-x 1 root root     363 Apr  8 08:36 x-pack-watcher-env*
```

- This command generate a list of credentials. Copy and save it somewhere.
  `./elasticsearch-setup-passwords auto`



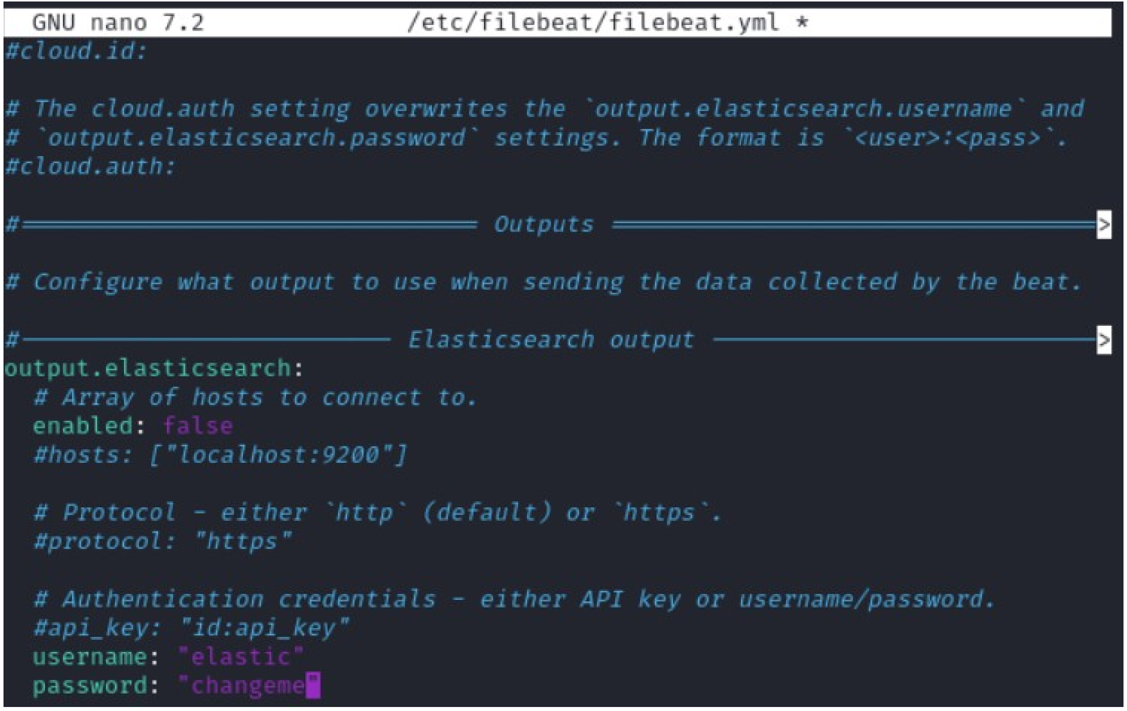
```
root@ubuntu-shad0w:/usr/share/elasticsearch/bin# ./elasticsearch-setup-passwo
rds auto
warning: usage of JAVA_HOME is deprecated, use ES_JAVA_HOME
Initiating the setup of passwords for reserved users elastic,apm_system,kiban
a,kibana_system,logstash_system,beats_system,remote_monitoring_user.
The passwords will be randomly generated and printed to the console.
Please confirm that you would like to continue [y/N]y


Changed password for user apm_system
PASSWORD apm_system = [redacted]

Changed password for user kibana_system
PASSWORD kibana_system = [redacted]

Changed password for user kibana
PASSWORD kibana = [redacted]

Changed password for user logstash_system
PASSWORD logstash_system = [redacted]

Changed password for user beats_system
PASSWORD beats_system =[redacted]

Changed password for user remote_monitoring_user
PASSWORD remote_monitoring_user =[redacted]

Changed password for user elastic
PASSWORD elastic = [redacted]
```

- Uncomment username and password, and replace current texts with previously generated credentials
  `nano /etc/kibana/kibana.yml`



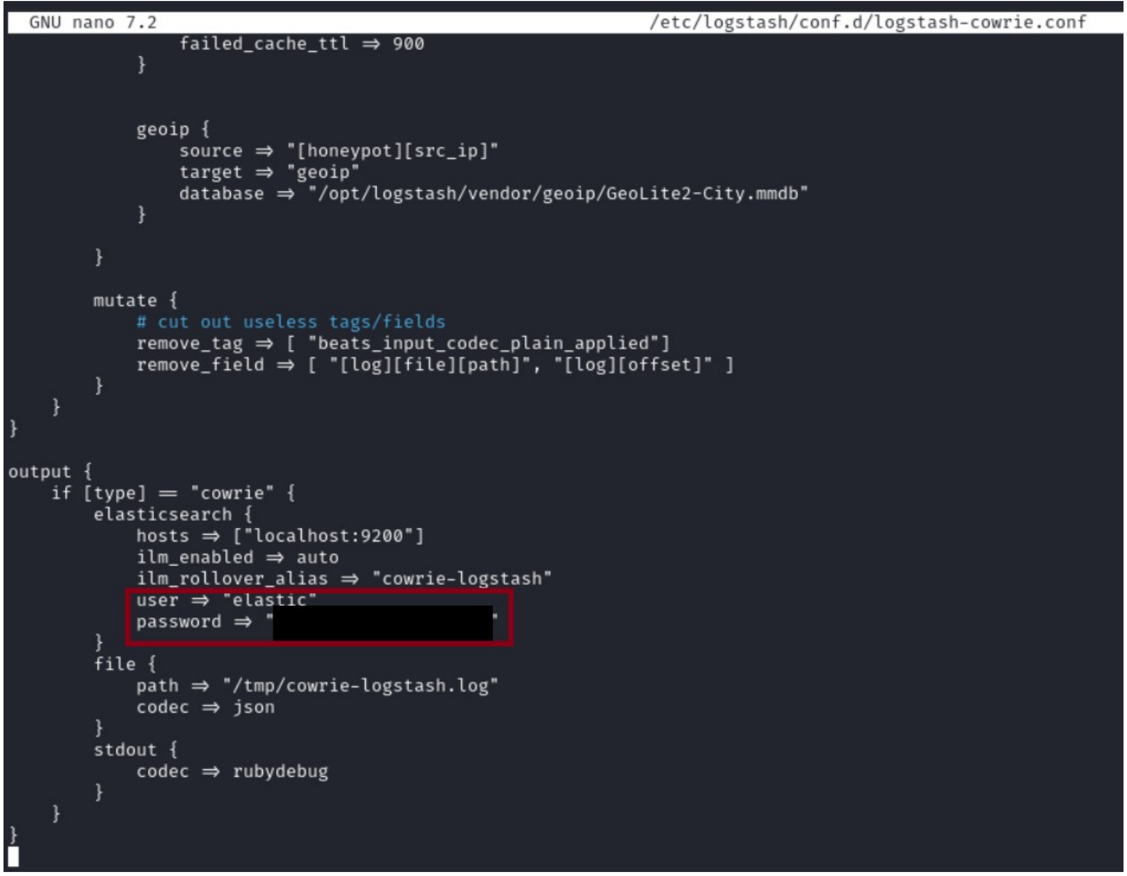
```
  GNU nano 7.2                /etc/kibana/kibana.yml *
# This setting was effectively always `false` before Kibana 6.3 and will
# default to `true` starting in Kibana 7.0.
#server.rewriteBasePath: false

# Specifies the public URL at which Kibana is available for end users. If
# `server.basePath` is configured this URL should end with the same basePath.
#server.publicBaseUrl: ""

# The maximum payload size in bytes for incoming server requests.
#server.maxPayload: 1048576

# The Kibana server's name.  This is used for display purposes.
#server.name: "your-hostname"

# The URLs of the Elasticsearch instances to use for all your queries.
#elasticsearch.hosts: ["http://localhost:9200"]

# Kibana uses an index in Elasticsearch to store saved searches, visualizati>
# dashboards. Kibana creates a new index if the index doesn't already exist.
#kibana.index: ".kibana"

# The default application to load.
#kibana.defaultAppId: "home"

# If your Elasticsearch is protected with basic authentication, these settin>
# the username and password that the Kibana server uses to perform maintenan>
# index at startup. Your Kibana users still need to authenticate with Elasti>
# is proxied through the Kibana server.
elasticsearch.username: "kibana_system"
elasticsearch.password: "pass"
```

nano /etc/filebeat/filebeat.yml



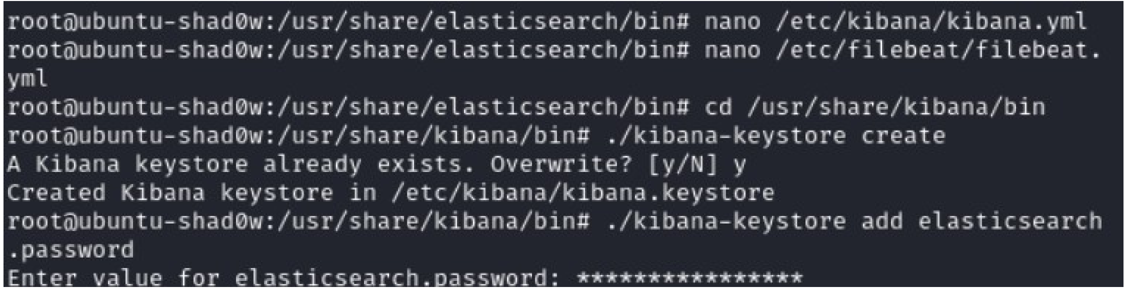
```
  GNU nano 7.2               /etc/filebeat/filebeat.yml *
#cloud.id:

# The cloud.auth setting overwrites the `output.elasticsearch.username` and
# `output.elasticsearch.password` settings. The format is `<user>:<pass>`.
#cloud.auth:

#================================ Outputs ================================>

# Configure what output to use when sending the data collected by the beat.

#-------------------------- Elasticsearch output -------------------------->
output.elasticsearch:
  # Array of hosts to connect to.
  enabled: false
  #hosts: ["localhost:9200"]

  # Protocol - either `http` (default) or `https`.
  #protocol: "https"

  # Authentication credentials - either API key or username/password.
  #api_key: "id:api_key"
  username: "elastic"
  password: "changeme"
```

`nano /etc/logstash/conf.d/logstash-cowrie.conf`



```
  GNU nano 7.2                                        /etc/logstash/conf.d/logstash-cowrie.conf
                failed_cache_ttl ⇒ 900
            }


            geoip {
                source ⇒ "[honeypot][src_ip]"
                target ⇒ "geoip"
                database ⇒ "/opt/logstash/vendor/geoip/GeoLite2-City.mmdb"
            }

        }

        mutate {
            # cut out useless tags/fields
            remove_tag ⇒ [ "beats_input_codec_plain_applied"]
            remove_field ⇒ [ "[log][file][path]", "[log][offset]" ]
        }
    }
}

output {
    if [type] == "cowrie" {
        elasticsearch {
            hosts ⇒ ["localhost:9200"]
            ilm_enabled ⇒ auto
            ilm_rollover_alias ⇒ "cowrie-logstash"
            user ⇒ "elastic"
            password ⇒ "[redacted]"
        }
        file {
            path ⇒ "/tmp/cowrie-logstash.log"
            codec ⇒ json
        }
        stdout {
            codec ⇒ rubydebug
        }
    }
}
```

- Change directory to kibana/bin and create a kibana-keystore elastic password
  `cd /usr/share/kibana/bin`
  `./kibana-keystore create`
  `./kibana-keystore add elasticsearch.password`



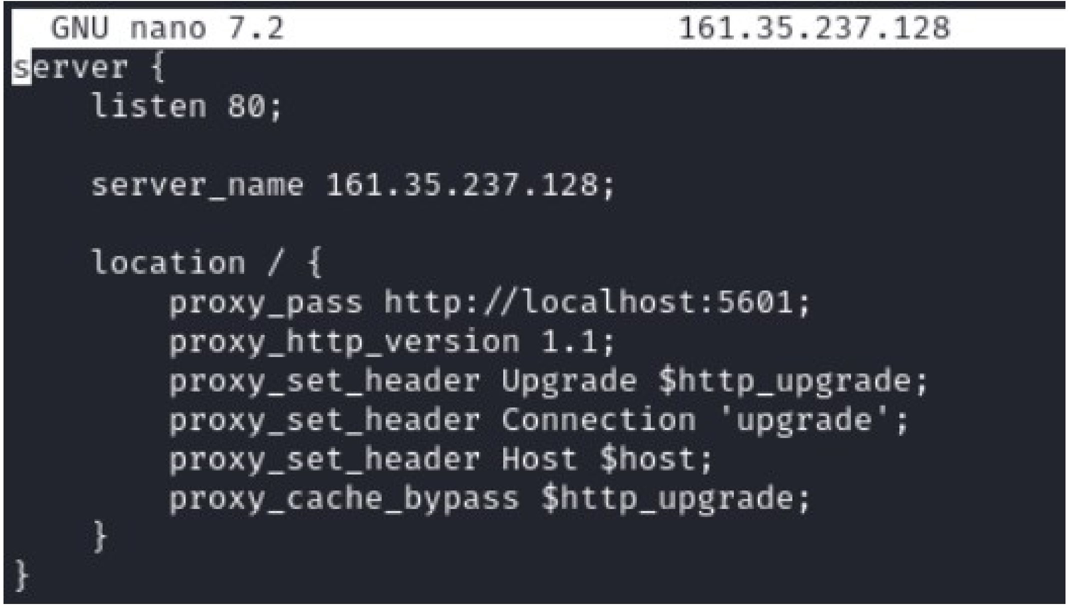
```
root@ubuntu-shad0w:/usr/share/elasticsearch/bin# nano /etc/kibana/kibana.yml
root@ubuntu-shad0w:/usr/share/elasticsearch/bin# nano /etc/filebeat/filebeat.
yml
root@ubuntu-shad0w:/usr/share/elasticsearch/bin# cd /usr/share/kibana/bin
root@ubuntu-shad0w:/usr/share/kibana/bin# ./kibana-keystore create
A Kibana keystore already exists. Overwrite? [y/N] y
Created Kibana keystore in /etc/kibana/kibana.keystore
root@ubuntu-shad0w:/usr/share/kibana/bin# ./kibana-keystore add elasticsearch
.password
Enter value for elasticsearch.password: ****************
```

- Start the stopped services
  `sudo systemctl start kibana filebeat logstash`

- Remove the nginx login requirement
  `rm /etc/nginx/htpasswd.users`

- Remove the lines `auth_basic "Restricted Access"`; `auth_basic_user_file /etc/nginx/htpasswd.users`

```
nano /etc/nginx/sites-available/<your_domain>
```



```
server {
    listen 80;

    server_name 161.35.237.128;

    location / {
        proxy_pass http://localhost:5601;
        proxy_http_version 1.1;
        proxy_set_header Upgrade $http_upgrade;
        proxy_set_header Connection 'upgrade';
        proxy_set_header Host $host;
        proxy_cache_bypass $http_upgrade;
    }
}
```

- Restart the nginx service

```
systemctl restart nginx
```