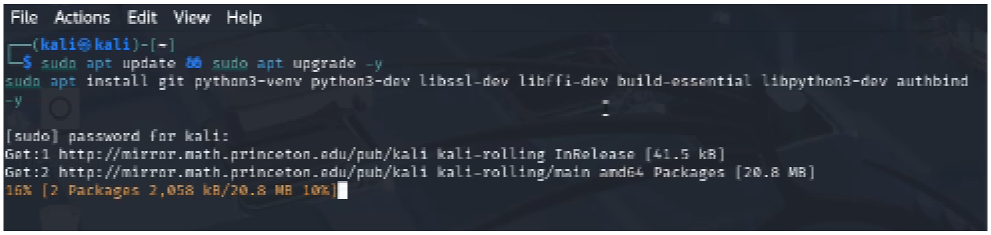
This guide walks you through setting up the Cowrie SSH honeypot on a Linux virtual machine. It also includes how to configure the honeypot to listen on port 22 using authbind.

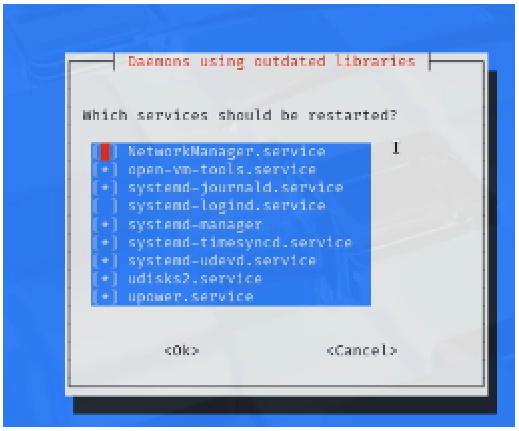
---

1. Install Dependencies

Run the following commands to update your system and install required packages:



```
┌──(kali㉿kali)-[~]
└─$ sudo apt update && sudo apt upgrade -y
sudo apt install git python3-venv python3-dev libssl-dev libffi-dev build-essential libpython3-dev authbind
-y

[sudo] password for kali:
Get:1 http://mirror.math.princeton.edu/pub/kali kali-rolling InRelease [41.5 kB]
Get:2 http://mirror.math.princeton.edu/pub/kali kali-rolling/main amd64 Packages [20.8 MB]
16% [2 Packages 2,058 kB/20.8 MB 10%]
```



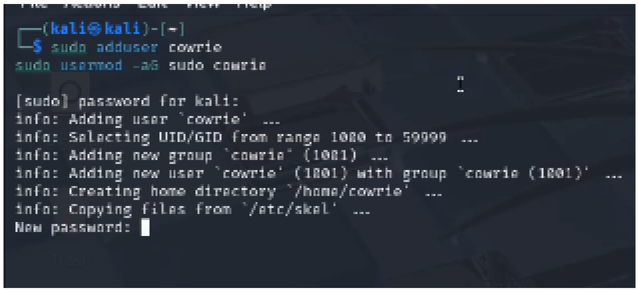
```
Daemons using outdated libraries

Which services should be restarted?

[ ] NetworkManager.service
[*] open-vm-tools.service
[*] systemd-journald.service
[ ] systemd-logind.service
[*] systemd-manager
[*] systemd-timesyncd.service
[*] systemd-udevd.service
[*] udisks2.service
[*] upower.service
```

Note: You may be prompted during the update. Press `Enter` to continue. After the update, a VM reboot may be required.

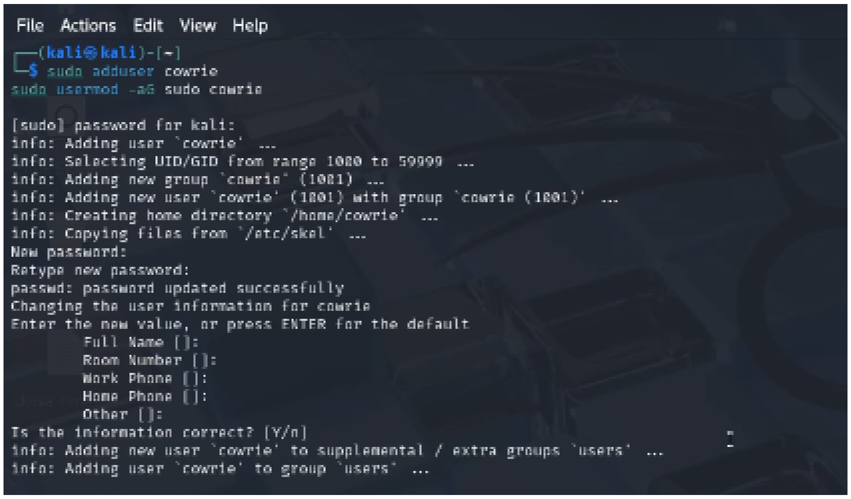
---

2. Create Cowrie User and Download Cowrie

sudo adduser cowrie
sudo su - cowrie



```
┌──(kali㉿kali)-[~]
└─$ sudo adduser cowrie
sudo usermod -aG sudo cowrie

[sudo] password for kali:
info: Adding user `cowrie' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `cowrie' (1001) ...
info: Adding new user `cowrie' (1001) with group `cowrie (1001)' ...
info: Creating home directory `/home/cowrie' ...
info: Copying files from `/etc/skel' ...
New password:
```

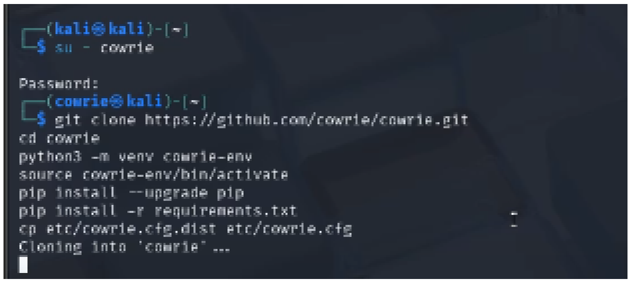


```
┌──(kali㉿kali)-[~]
└─$ sudo adduser cowrie
sudo usermod -aG sudo cowrie

[sudo] password for kali:
info: Adding user `cowrie' ...
info: Selecting UID/GID from range 1000 to 59999 ...
info: Adding new group `cowrie' (1001) ...
info: Adding new user `cowrie' (1001) with group `cowrie (1001)' ...
info: Creating home directory `/home/cowrie' ...
info: Copying files from `/etc/skel' ...
New password:
Retype new password:
passwd: password updated successfully
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]
info: Adding new user `cowrie' to supplemental / extra groups `users' ...
info: Adding user `cowrie' to group `users' ...
```

Note: You may be asked to set a password. You can skip the full profile setup by repeatedly pressing `Enter`.

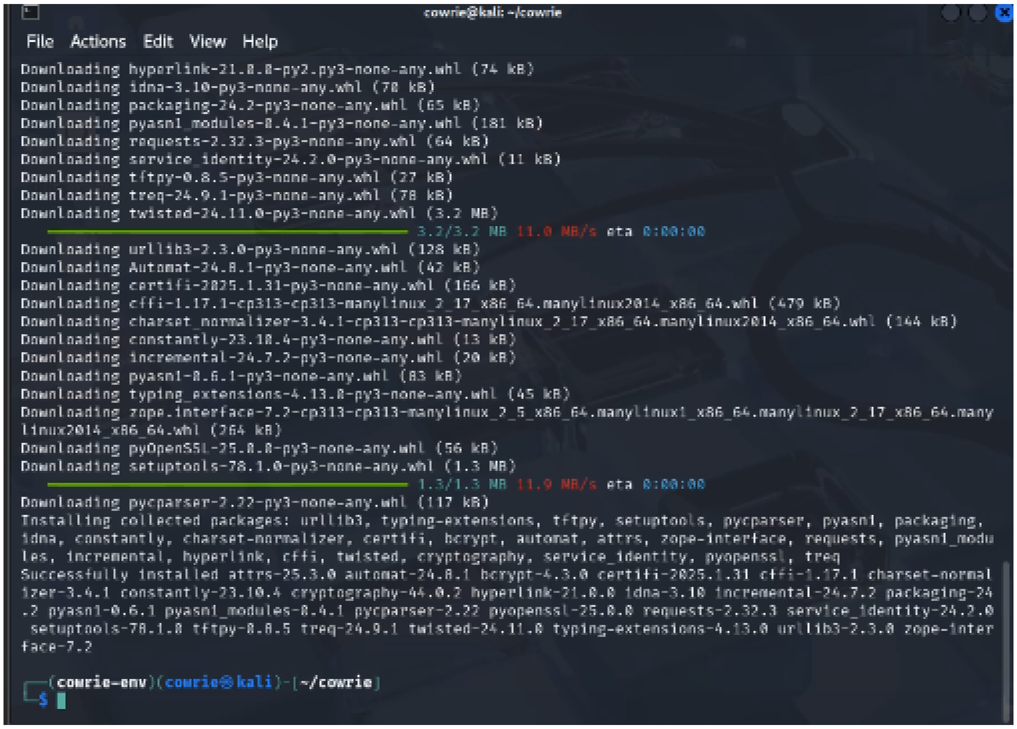
Clone and Set up the Cowrie repository:

```
git clone https://github.com/cowrie/cowrie.git
cd cowrie
python3 -m venv cowrie-env
source cowrie-env/bin/activate
pip install --upgrade pip
pip install -r requirements.txt
cp etc/cowrie.cfg.dist etc/cowrie.cfg
```



```
┌──(kali㉿kali)-[~]
└─$ su - cowrie

Password:
┌──(cowrie㉿kali)-[~]
└─$ git clone https://github.com/cowrie/cowrie.git
cd cowrie
python3 -m venv cowrie-env
source cowrie-env/bin/activate
pip install --upgrade pip
pip install -r requirements.txt
cp etc/cowrie.cfg.dist etc/cowrie.cfg
Cloning into 'cowrie'...
```

After the install run the following commands individually or all together; these commands will log into the cowrie user and, clone and set up cowrie



```
Downloading hyperlink-21.0.0-py2.py3-none-any.whl (74 kB)
Downloading idna-3.10-py3-none-any.whl (70 kB)
Downloading packaging-24.2-py3-none-any.whl (65 kB)
Downloading pyasn1_modules-0.4.1-py3-none-any.whl (181 kB)
Downloading requests-2.32.3-py3-none-any.whl (64 kB)
Downloading service_identity-24.2.0-py3-none-any.whl (11 kB)
Downloading tftpy-0.8.5-py3-none-any.whl (27 kB)
Downloading treq-24.9.1-py3-none-any.whl (78 kB)
Downloading twisted-24.11.0-py3-none-any.whl (3.2 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 3.2/3.2 MB 11.0 MB/s eta 0:00:00
Downloading urllib3-2.3.0-py3-none-any.whl (128 kB)
Downloading Automat-24.8.1-py3-none-any.whl (42 kB)
Downloading certifi-2025.1.31-py3-none-any.whl (166 kB)
Downloading cffi-1.17.1-cp313-cp313-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (479 kB)
Downloading charset_normalizer-3.4.1-cp313-cp313-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (144 kB)
Downloading constantly-23.10.4-py3-none-any.whl (13 kB)
Downloading incremental-24.7.2-py3-none-any.whl (20 kB)
Downloading pyasn1-0.6.1-py3-none-any.whl (83 kB)
Downloading typing_extensions-4.13.0-py3-none-any.whl (45 kB)
Downloading zope.interface-7.2-cp313-cp313-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_17_x86_64.many
linux2014_x86_64.whl (264 kB)
Downloading pyOpenSSL-25.0.0-py3-none-any.whl (56 kB)
Downloading setuptools-78.1.0-py3-none-any.whl (1.3 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.3/1.3 MB 11.9 MB/s eta 0:00:00
Downloading pycparser-2.22-py3-none-any.whl (117 kB)
Installing collected packages: urllib3, typing-extensions, tftpy, setuptools, pycparser, pyasn1, packaging,
idna, constantly, charset-normalizer, certifi, bcrypt, automat, attrs, zope-interface, requests, pyasn1_modu
les, incremental, hyperlink, cffi, twisted, cryptography, service_identity, pyopenssl, treq
Successfully installed attrs-25.3.0 automat-24.8.1 bcrypt-4.3.0 certifi-2025.1.31 cffi-1.17.1 charset-normal
izer-3.4.1 constantly-23.10.4 cryptography-44.0.2 hyperlink-21.0.0 idna-3.10 incremental-24.7.2 packaging-24
.2 pyasn1-0.6.1 pyasn1_modules-0.4.1 pycparser-2.22 pyopenssl-25.0.0 requests-2.32.3 service_identity-24.2.0
 setuptools-78.1.0 tftpy-0.8.5 treq-24.9.1 twisted-24.11.0 typing-extensions-4.13.0 urllib3-2.3.0 zope-inter
face-7.2

┌──(cowrie-env)(cowrie㉿kali)-[~/cowrie]
└─$ 
```

---

3. Configure Cowrie to Listen on Port 22

Edit the configuration file:

nano etc/cowrie.etc

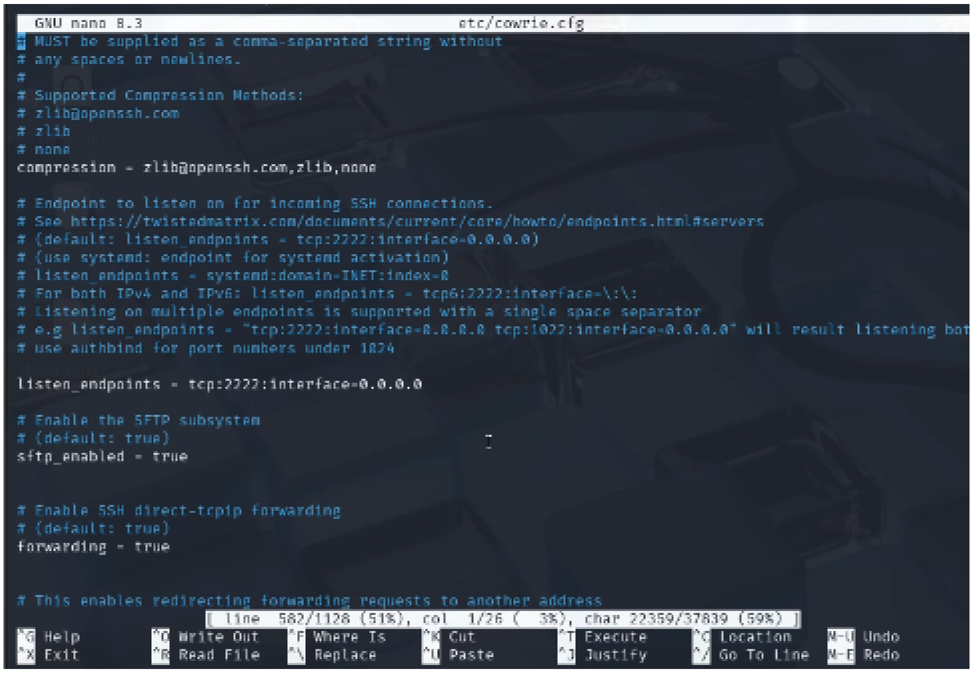
Find the line:

listen_endpoints = 2222:interface=0.0.0.0

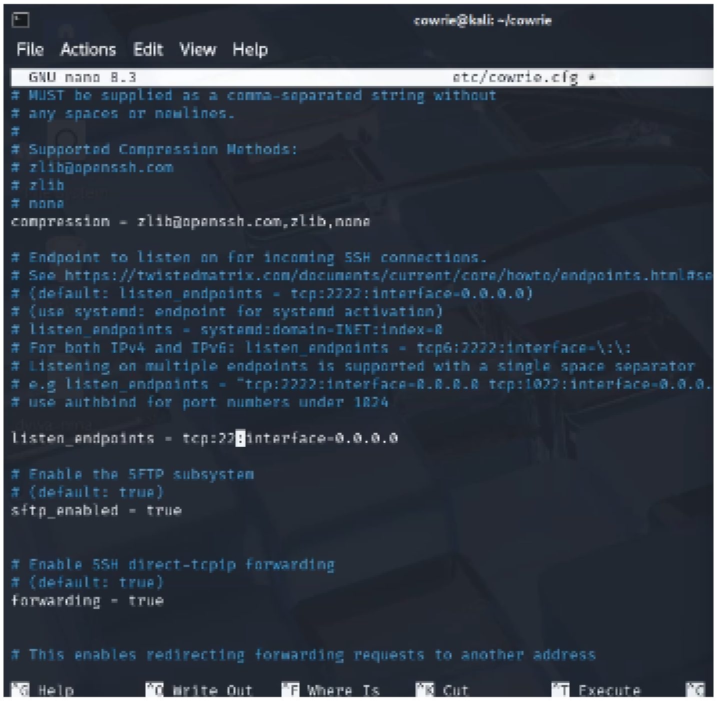
Change it to:

listen_endpoints = 22:interface=0.0.0.0

This ensures Cowrie listens on the standard SSH port.



```
  GNU nano 8.3                               etc/cowrie.cfg
# MUST be supplied as a comma-separated string without
# any spaces or newlines.
#
# Supported Compression Methods:
# zlib@openssh.com
# zlib
# none
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 separator
# e.g listen_endpoints = "tcp:2222:interface=0.0.0.0 tcp:1022:interface=0.0.0.0" will result listening bot
# use authbind for port numbers under 1024

listen_endpoints = tcp:2222:interface=0.0.0.0

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


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


# This enables redirecting forwarding requests to another address
                    [ line  582/1128 (51%), col  1/26 (  3%), char 22359/37839 (59%) ]
^G Help       ^O Write Out  ^F Where Is   ^K Cut        ^T Execute    ^C Location   M-U Undo
^X Exit       ^R Read File  ^\ Replace    ^U Paste      ^J Justify    ^/ Go To Line M-E Redo
```



```
 GNU nano 8.3                               etc/cowrie.cfg *
# MUST be supplied as a comma-separated string without
# any spaces or newlines.
#
# Supported Compression Methods:
# zlib@openssh.com
# zlib
# none
compression = zlib@openssh.com,zlib,none

# Endpoint to listen on for incoming SSH connections.
# See https://twistedmatrix.com/documents/current/core/howto/endpoints.html#se
# (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 separator
# e.g listen_endpoints = "tcp:2222:interface=0.0.0.0 tcp:1022:interface=0.0.0.
# 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


# This enables redirecting forwarding requests to another address

^G Help       ^O Write Out  ^F Where Is   ^K Cut        ^T Execute    ^C
```

---

4. Enable `authbind` to Use Port 22

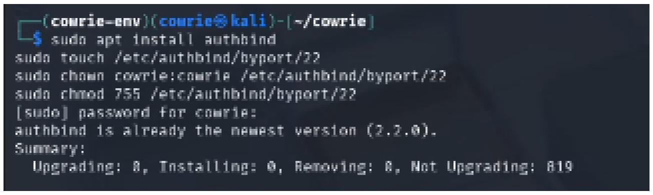
Run the following commands:

sudo apt install authbind

sudo touch /etc/authbind/byport/22

sudo chown cowrie:cowrie /etc/authbind/byport/22

sudo chmod 755 /etc/authbind/byport/22

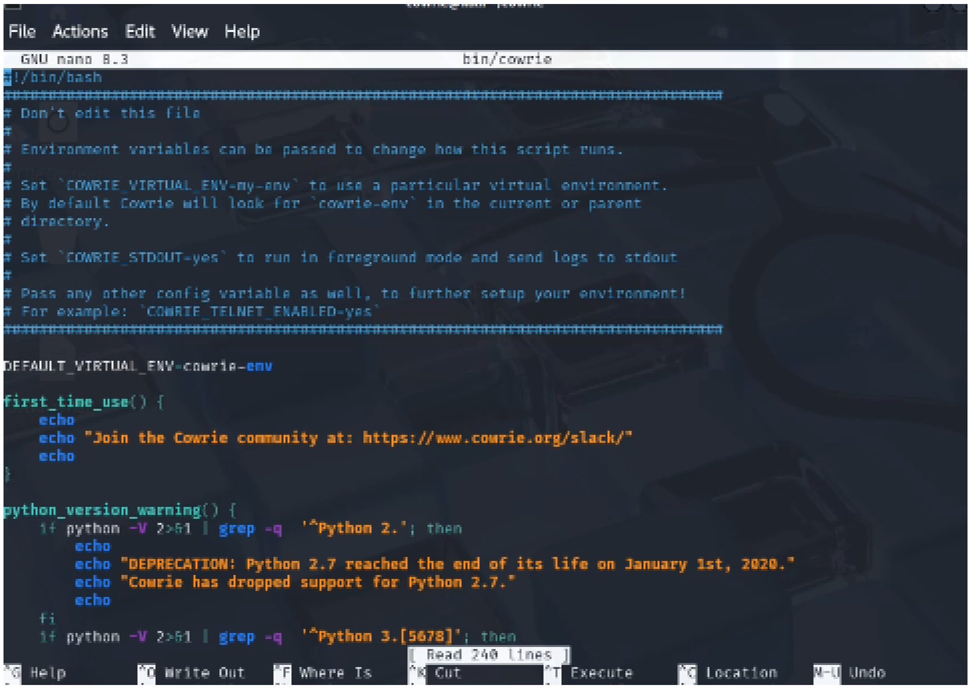


```
┌──(cowrie-env)(cowrie㉿kali)-[~/cowrie]
└─$ sudo apt install authbind
sudo touch /etc/authbind/byport/22
sudo chown cowrie:cowrie /etc/authbind/byport/22
sudo chmod 755 /etc/authbind/byport/22
[sudo] password for cowrie:
authbind is already the newest version (2.2.0).
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 819
```

Edit the startup script to enable authbind:

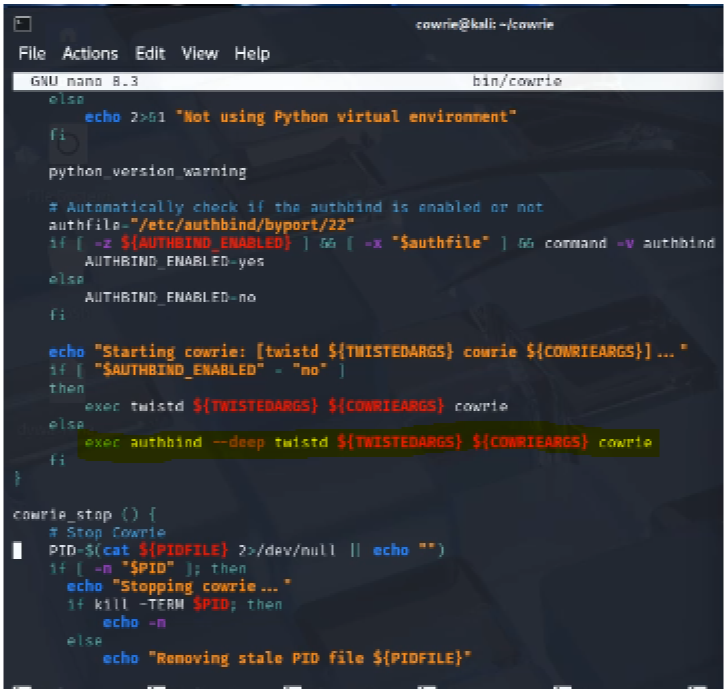
nano bin/cowrie

it may be hard to find this line of code to edit in the script but it should be towards the bottom



```
  GNU nano 8.3                                             bin/cowrie
#!/bin/bash
################################################################################
# Don't edit this file
#
# Environment variables can be passed to change how this script runs.
#
# Set `COWRIE_VIRTUAL_ENV=my-env` to use a particular virtual environment.
# By default Cowrie will look for `cowrie-env` in the current or parent
# directory.
#
# Set `COWRIE_STDOUT=yes` to run in foreground mode and send logs to stdout
#
# Pass any other config variable as well, to further setup your environment!
# For example: `COWRIE_TELNET_ENABLED=yes`
################################################################################

DEFAULT_VIRTUAL_ENV=cowrie-env

first_time_use() {
    echo
    echo "Join the Cowrie community at: https://www.cowrie.org/slack/"
    echo
}

python_version_warning() {
    if python -V 2>&1 | grep -q  '^Python 2.'; then
        echo
        echo "DEPRECATION: Python 2.7 reached the end of its life on January 1st, 2020."
        echo "Cowrie has dropped support for Python 2.7."
        echo
    fi
    if python -V 2>&1 | grep -q  '^Python 3.[5678]'; then
                                        [ Read 240 lines ]
^G Help         ^O Write Out    ^F Where Is     ^K Cut          ^T Execute      ^C Location     M-U Undo
```

It should look something like this:

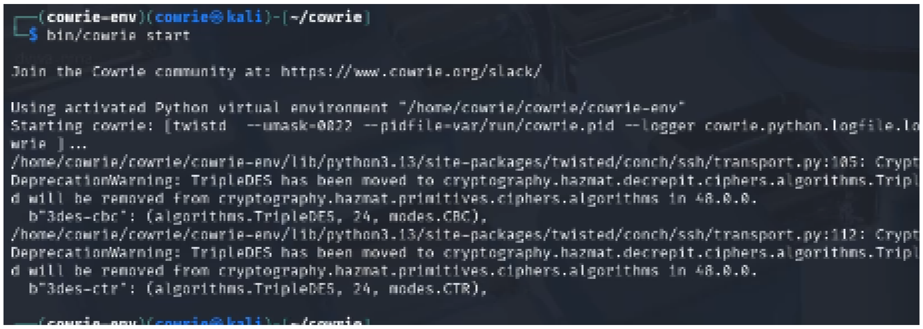


```
 GNU nano 8.3                                         bin/cowrie
    else
        echo 2>&1 "Not using Python virtual environment"
    fi

    python_version_warning

    # Automatically check if the authbind is enabled or not
    authfile="/etc/authbind/byport/22"
    if [ -z ${AUTHBIND_ENABLED} ] && [ -x "$authfile" ] && command -v authbind
        AUTHBIND_ENABLED=yes
    else
        AUTHBIND_ENABLED=no
    fi

    echo "Starting cowrie: [twistd ${TWISTEDARGS} cowrie ${COWRIEARGS}]..."
    if [ "$AUTHBIND_ENABLED" = "no" ]
    then
        exec twistd ${TWISTEDARGS} ${COWRIEARGS} cowrie
    else
        exec authbind --deep twistd ${TWISTEDARGS} ${COWRIEARGS} cowrie
    fi
}

cowrie_stop () {
    # Stop Cowrie
    PID=$(cat ${PIDFILE} 2>/dev/null || echo "")
    if [ -n "$PID" ]; then
      echo "Stopping cowrie..."
      if kill -TERM $PID; then
          echo -n
      else
          echo "Removing stale PID file ${PIDFILE}"
```

5. Start Cowrie

Start the honeypot:

bin/cowrie start

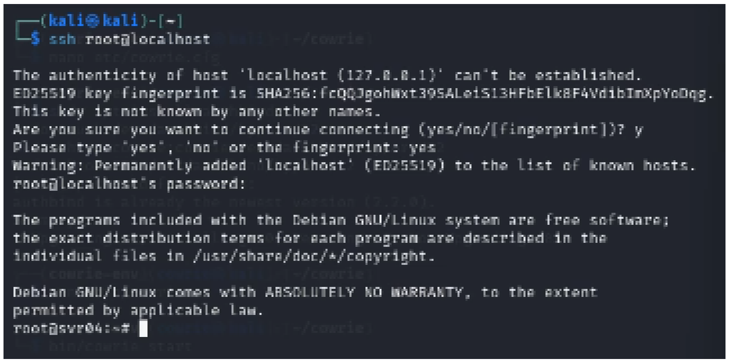


```
┌──(cowrie-env)(cowrie㉿kali)-[~/cowrie]
└─$ bin/cowrie start

Join the Cowrie community at: https://www.cowrie.org/slack/

Using activated Python virtual environment "/home/cowrie/cowrie/cowrie-env"
Starting cowrie: [twistd  --umask=0022 --pidfile=var/run/cowrie.pid --logger cowrie.python.logfile.lo
wrie ]...
/home/cowrie/cowrie/cowrie-env/lib/python3.13/site-packages/twisted/conch/ssh/transport.py:105: Crypt
DeprecationWarning: TripleDES has been moved to cryptography.hazmat.decrepit.ciphers.algorithms.Tripl
d will be removed from cryptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
  b"3des-cbc": (algorithms.TripleDES, 24, modes.CBC),
/home/cowrie/cowrie/cowrie-env/lib/python3.13/site-packages/twisted/conch/ssh/transport.py:112: Crypt
DeprecationWarning: TripleDES has been moved to cryptography.hazmat.decrepit.ciphers.algorithms.Tripl
d will be removed from cryptography.hazmat.primitives.ciphers.algorithms in 48.0.0.
  b"3des-ctr": (algorithms.TripleDES, 24, modes.CTR),

┌──(cowrie-env)(cowrie㉿kali)-[~/cowrie]
```

6. Test Your Honeypot

From the same machine, test your honeypot:

ssh cowrie@localhost



```
┌──(kali㉿kali)-[~]
└─$ ssh root@localhost
The authenticity of host 'localhost (127.0.0.1)' can't be established.
ED25519 key fingerprint is SHA256:fcQQJgohWxt39SALeiS13HFbElk8F4VdibImXpYoDqg.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? y
Please type 'yes', 'no' or the fingerprint: yes
Warning: Permanently added 'localhost' (ED25519) to the list of known hosts.
root@localhost's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
root@svr04:~# 
```

Note: You should be able to log in regardless of the password you enter.

✅ Success!

Congratulations, yayyy! You have successfully set up a Cowrie Honeypot.

You can further configure Cowrie to display fake directories and files to capture attacker behaviors and techniques.

➡️ **Next Step:** Follow the Slack Integration guide to receive real-time alerts every time someone interacts with your honeypot.

By cyberboss