

[illegible]

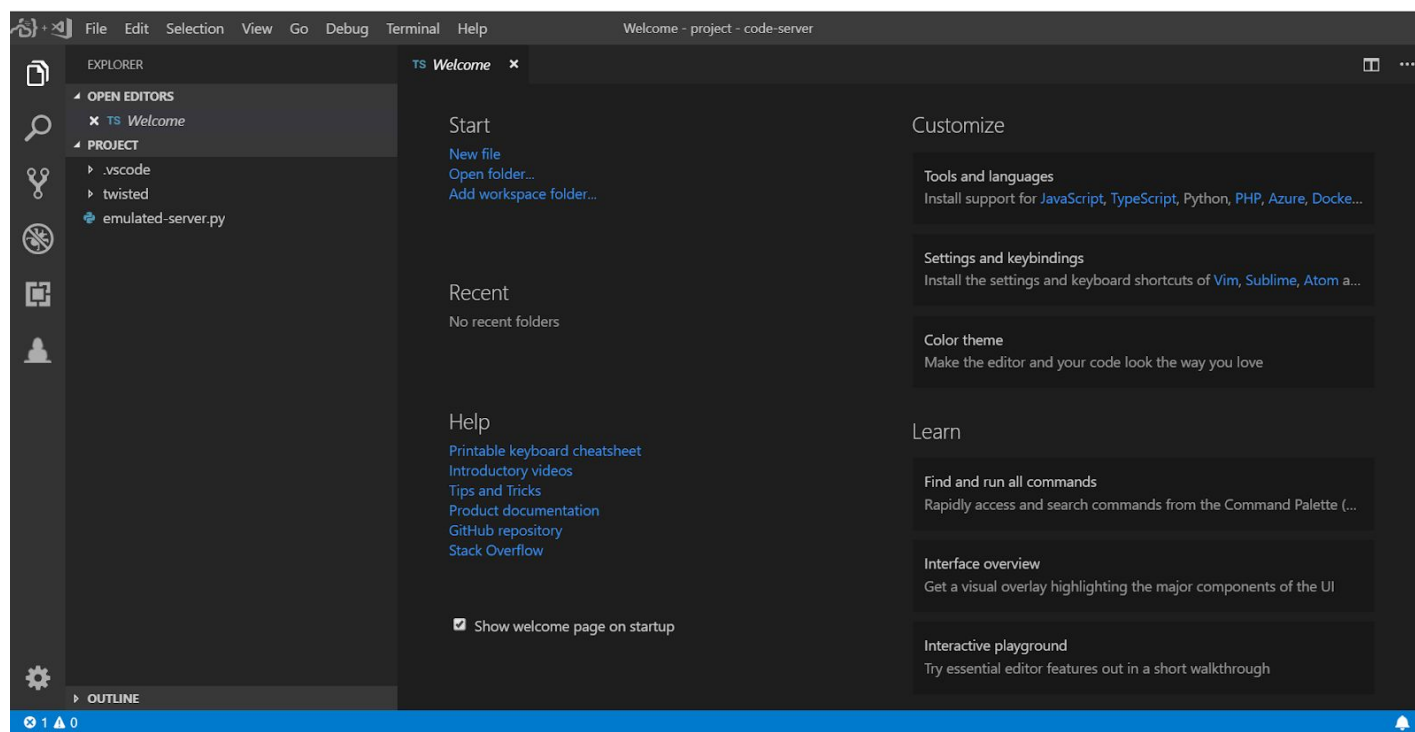
Name	SSH Server Emulation
URL	https://attackdefense.com/challengedetails?cid=1213
Type	Offensive Python : Server Emulation

Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

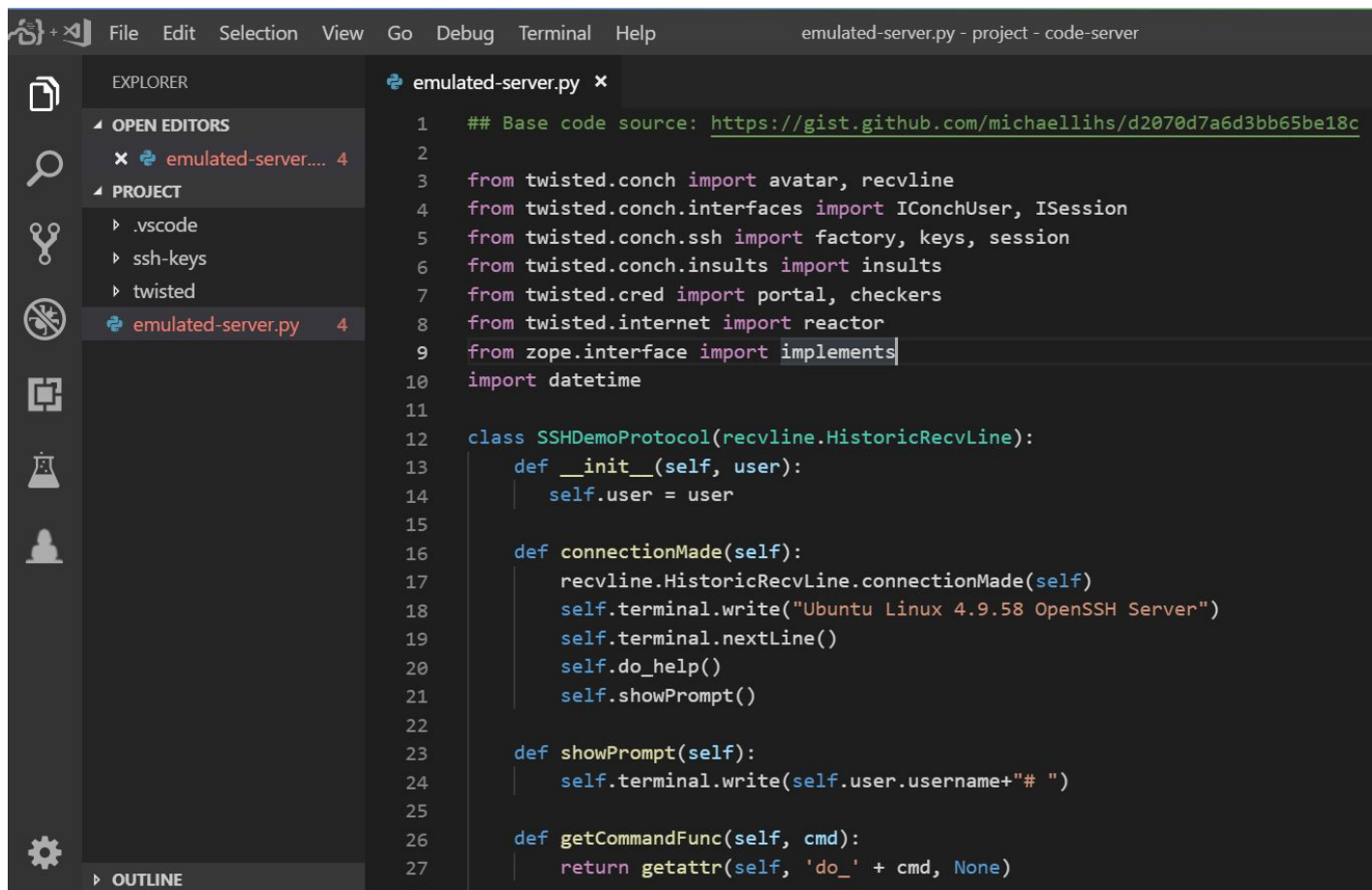
Objective: Modify the code, launch the server and use Kali Linux to interact/attack it.

Solution:

Landing Page:



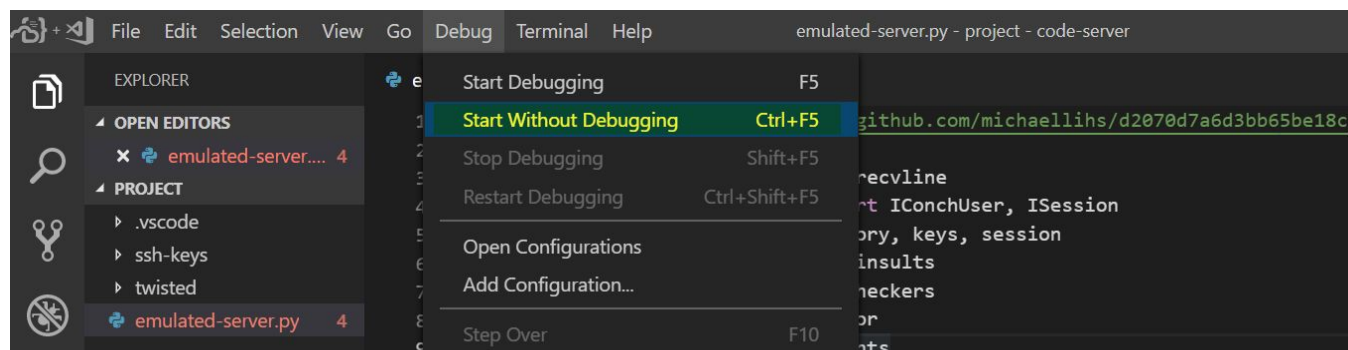
Step 1: Select “emulated-server.py” from the Project Explorer.



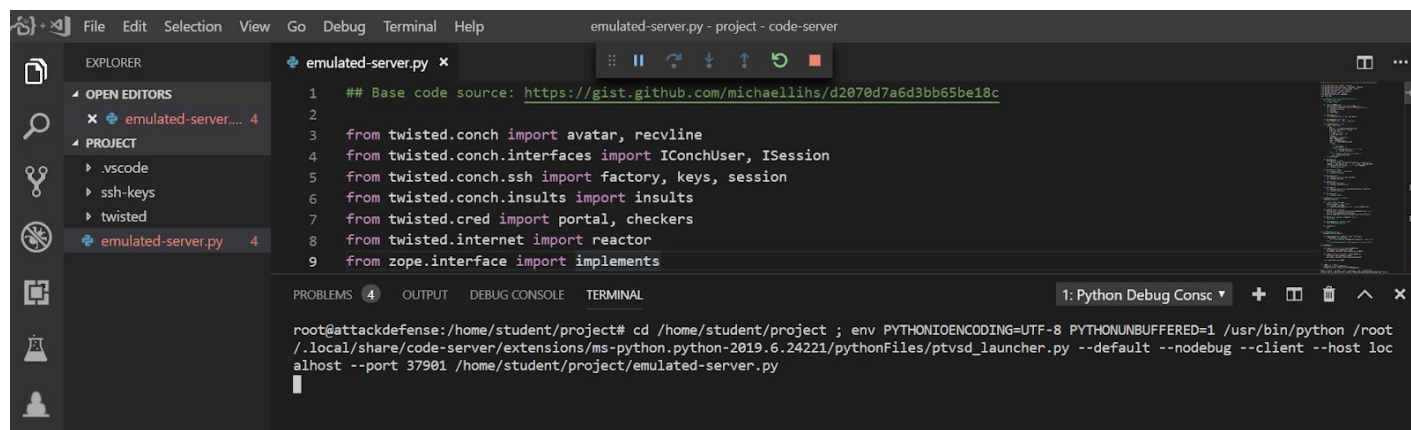
The screenshot shows the Visual Studio Code interface. On the left, the Project Explorer is open, showing the file structure of the 'project - code-server' workspace. The file 'emulated-server.py' is selected under the 'PROJECT' section. The main editor area displays the code for 'emulated-server.py'. The code is a Python script using the Twisted framework to create a simple SSH server. It imports necessary modules from Twisted and defines a class 'SSHDemoProtocol' that inherits from 'HistoricRecvLine'. The class has methods for initialization, connection handling, and command execution.

```
1  ## Base code source: https://gist.github.com/michaellihs/d2070d7a6d3bb65be18c
2
3  from twisted.conch import avatar, recvline
4  from twisted.conch.interfaces import IConchUser, ISession
5  from twisted.conch.ssh import factory, keys, session
6  from twisted.conch.insults import insults
7  from twisted.cred import portal, checkers
8  from twisted.internet import reactor
9  from zope.interface import implements
10 import datetime
11
12 class SSHDemoProtocol(recvline.HistoricRecvLine):
13     def __init__(self, user):
14         self.user = user
15
16     def connectionMade(self):
17         recvline.HistoricRecvLine.connectionMade(self)
18         self.terminal.write("Ubuntu Linux 4.9.58 OpenSSH Server")
19         self.terminal.nextLine()
20         self.do_help()
21         self.showPrompt()
22
23     def showPrompt(self):
24         self.terminal.write(self.user.username+"# ")
25
26     def getCommandFunc(self, cmd):
27         return getattr(self, 'do_' + cmd, None)
```

Step 2: Navigate to Debug Menu and click on “Start Without Debugging option” to run the program.



The python script will run and start an SSH Server on port 22.



The screenshot shows the Visual Studio Code interface. The Explorer sidebar on the left shows the project structure with files like .vscode, ssh-keys, twisted, and emulated-server.py. The main editor area displays the emulated-server.py file, which contains Python code for setting up an SSH server using Twisted. The terminal window at the bottom shows the command to run the script: `root@attackdefense:/home/student/project# cd /home/student/project ; env PYTHONIOENCODING=UTF-8 PYTHONUNBUFFERED=1 /usr/bin/python /root/.local/share/code-server/extensions/ms-python.python-2019.6.24221/pythonFiles/ptvsd_launcher.py --default --nodebug --client --host localhost --port 37901 /home/student/project/emulated-server.py`

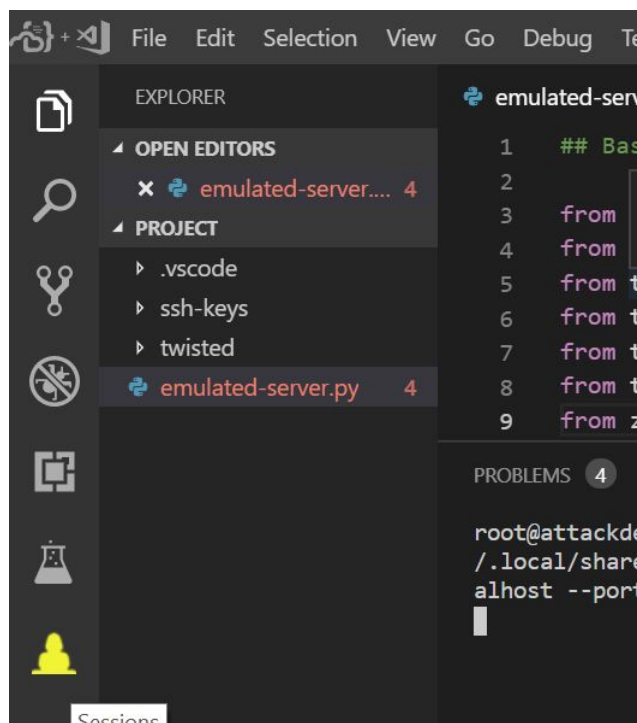
```
1  ## Base code source: https://gist.github.com/michaellihs/d2070d7a6d3bb65be18c
2
3  from twisted.conch import avatar, recvline
4  from twisted.conch.interfaces import IConchUser, ISession
5  from twisted.conch.ssh import factory, keys, session
6  from twisted.conch.insults import insults
7  from twisted.cred import portal, checkers
8  from twisted.internet import reactor
9  from zope.interface import implements
```

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL

1: Python Debug Consc

root@attackdefense:/home/student/project# cd /home/student/project ; env PYTHONIOENCODING=UTF-8 PYTHONUNBUFFERED=1 /usr/bin/python /root/.local/share/code-server/extensions/ms-python.python-2019.6.24221/pythonFiles/ptvsd_launcher.py --default --nodebug --client --host localhost --port 37901 /home/student/project/emulated-server.py

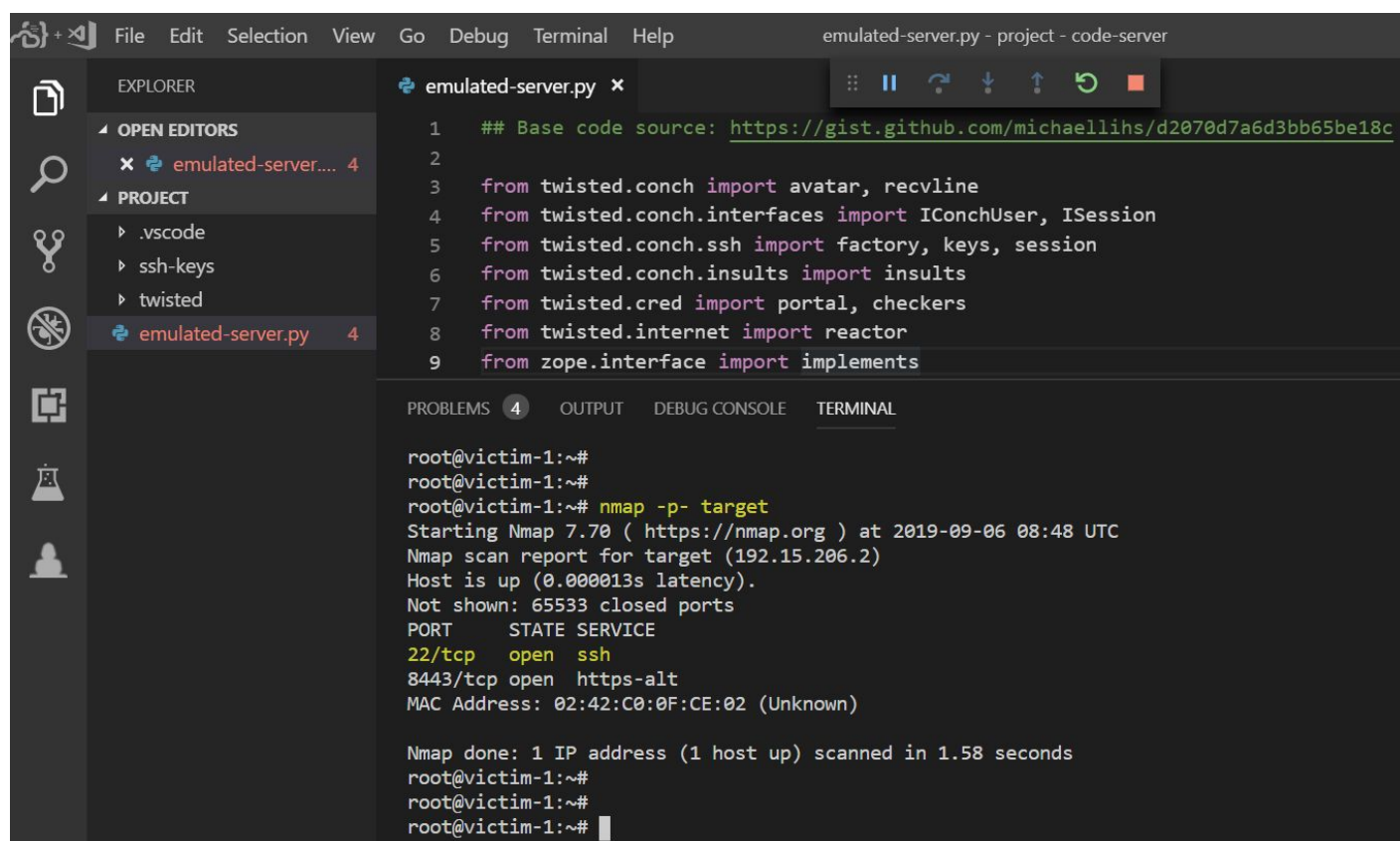
Step 3: Click on the “Sessions” icon on the activity bar to gain access to Kali machine.



This will spawn a new Terminal “Session 1” which will provide a bash shell on a remote Kali machine.

Step 4: Perform Nmap scan from the Kali machine. Identify the services running on the machine on which the IDE (and code) is running. The IDE machine is mapped to “target” hostname. So, “target” can be used while launching scans or tools on this machine. Alternatively, the IP address of the both machines can be found by running “ip addr” command on respective machine. The IDE machine should be on 192.x.y.2 and Kali machine should be on 192.x.y.3.

Command: nmap -p- target



The screenshot shows the Visual Studio Code interface. The Explorer panel on the left shows the project structure with files like .vscode, ssh-keys, twisted, and emulated-server.py. The main editor area displays the emulated-server.py file, which contains a Twisted SSH server implementation. The Terminal panel at the bottom shows the execution of the nmap -p- target command from a root shell on a victim machine. The output of the scan indicates that port 22/tcp is open and running SSH.

```
emulated-server.py x
1  ## Base code source: https://gist.github.com/michaelihs/d2070d7a6d3bb65be18c
2
3  from twisted.conch import avatar, recvline
4  from twisted.conch.interfaces import IConchUser, ISession
5  from twisted.conch.ssh import factory, keys, session
6  from twisted.conch.insults import insults
7  from twisted.cred import portal, checkers
8  from twisted.internet import reactor
9  from zope.interface import implements

PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL

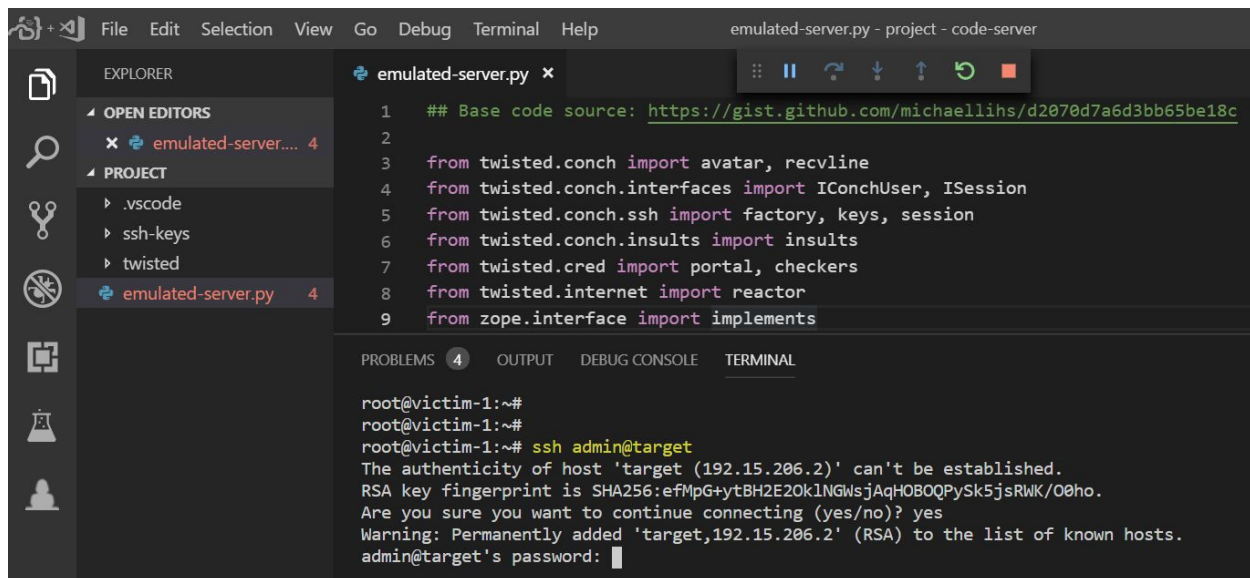
root@victim-1:~#
root@victim-1:~#
root@victim-1:~# nmap -p- target
Starting Nmap 7.70 ( https://nmap.org ) at 2019-09-06 08:48 UTC
Nmap scan report for target (192.15.206.2)
Host is up (0.000013s latency).
Not shown: 65533 closed ports
PORT      STATE SERVICE
22/tcp    open  ssh
8443/tcp   open  https-alt
MAC Address: 02:42:C0:0F:CE:02 (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.58 seconds
root@victim-1:~#
root@victim-1:~#
root@victim-1:~#
```

SSH server is running on port 22 on the target machine.

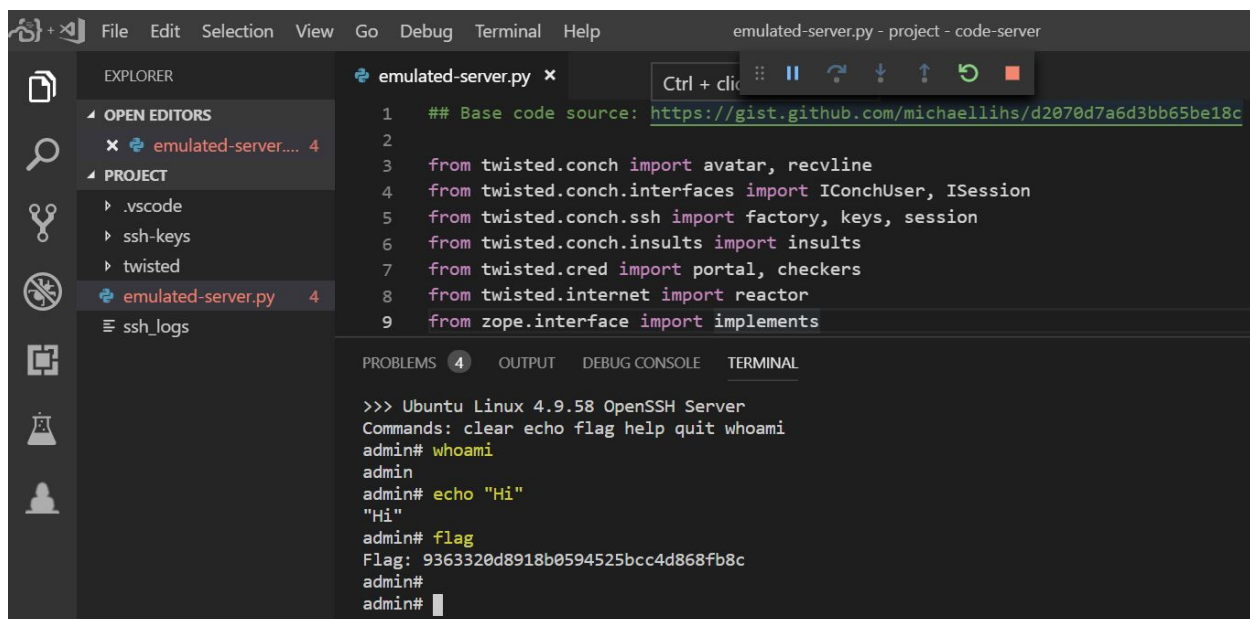
Step 5: Login to SSH server using username “admin” and password “welcome”

Command: `ssh admin@target`



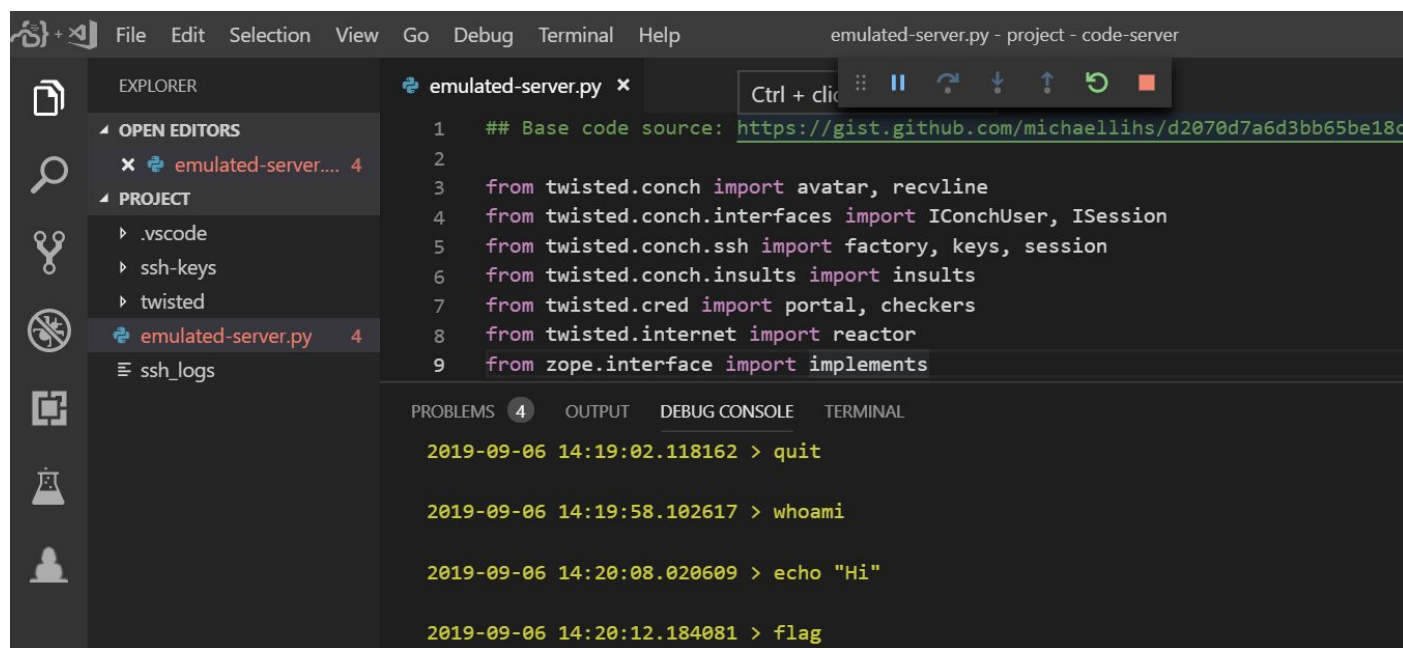
```
emulated-server.py - project - code-server
EXPLORER
  OPEN EDITORS
    x emulated-server.... 4
  PROJECT
    .vscode
    ssh-keys
    twisted
    emulated-server.py 4
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL
root@victim-1:~#
root@victim-1:~#
root@victim-1:~# ssh admin@target
The authenticity of host 'target (192.15.206.2)' can't be established.
RSA key fingerprint is SHA256:efMpG+ytBH2E20k1NGwsjAqH0BOQPysK5jsRWK/00ho.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'target,192.15.206.2' (RSA) to the list of known hosts.
admin@target's password: 
```

After successful authentication, a custom shell will be provided to the user on which custom commands can be executed.



```
emulated-server.py - project - code-server
EXPLORER
  OPEN EDITORS
    x emulated-server.... 4
  PROJECT
    .vscode
    ssh-keys
    twisted
    emulated-server.py 4
    ssh_logs
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL
>>> Ubuntu Linux 4.9.58 OpenSSH Server
Commands: clear echo flag help quit whoami
admin# whoami
admin
admin# echo "Hi"
"Hi"
admin# flag
Flag: 9363320d8918b0594525bcc4d868fb8c
admin#
admin# 
```

Step 6: On debug console, the interactions with SSH server can be viewed.



The screenshot shows the Visual Studio Code interface with the 'emulated-server.py' file open. The left sidebar shows the Explorer view with the project structure: .vscode, ssh-keys, twisted, emulated-server.py (selected), and ssh_logs. The main editor area displays the Python code for emulated-server.py, which uses Twisted to handle SSH connections. The bottom panel shows the DEBUG CONSOLE with the following output:

```
2019-09-06 14:19:02.118162 > quit
2019-09-06 14:19:58.102617 > whoami
2019-09-06 14:20:08.020609 > echo "Hi"
2019-09-06 14:20:12.184081 > flag
```

References:

1. Visual Studio Code (<https://code.visualstudio.com/>)
2. VS Code Basic Editing (<https://code.visualstudio.com/docs/editor/codebasics>)
3. Twisted (<https://www.twistedmatrix.com/trac/>)