PACKET-SNIFFING BACKDOOR

Testing

Danny Lieu

A00831407 | MAY 22, 2018

Table of Contents

Test outline

Test #	Description	Tools	Expected Results	Actual Results	Status
1	Root privilege checking for attacker script.	Python3	A message will be printed to ask for root access.	The message is printed for the normal user.	Pass. See test description for more details.
2	Root privilege checking for backdoor script.	Python3	A message will be printed to ask for root access.	The message is printed for the normal user.	Pass. See test description for more details.
3	The port knocking procedure is done with correct sequence.	Python3/ Wireshark	The attacker will knock the backdoor 3 times with given ports.	After 3 knocks, the backdoor is ready to	Pass. See test description for more details.

				receive command.	
4	The attack is able to send encrypted command to the backdoor. After that, the backdoor decrypt the payload to get the command and print to stdout.	Python3/ Wireshark	Looking at Wireshark capture, the payload of the packet should be encrypted. The backdoor is able to print the command to stdout.	The backdoor printed out the command.	Pass. See test description for more details.
5	The backdoor sends the encrypted result of the command to the attacker. After that, the attacker decrypts the payload to get the result of the command and print to stdout.	Python3/ Wireshark	Looking at Wireshark capture, the traffic between backdoor and attacker should be encrypted. The attacker should be able to print the result to stdout.	The attacker printed the result to stdout.	Pass. See test description for more details.
6	The communication between the attacker and the backdoor will be	Python3	After the attacker send “close” command, both	Both the attack and the	Pass. See test description

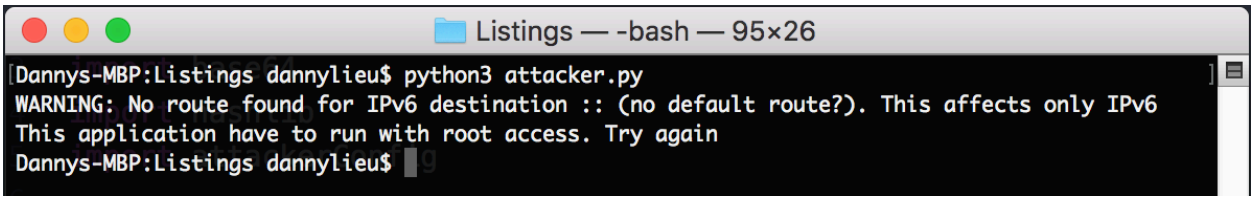
	stopped by “close” command.		script will print an appropriate message and exit.	backdoor printed out the closing messages.	for more details.

Test description

Test #1

Root privilege checking for attacker script.

To test for root privilege, the username is dannylieu which is not the root user. We run the script under dannylieu username:



The script is unable to run and ask for root access.

Test #2

Root privilege checking for backdoor script.

To test for root privilege, the username is dannylieu which is not the root user. We run the script under dannylieu username:

```
Listings — -bash — 95×26
[Dannys-MBP:Listings dannylieu$ python3 backdoor.py
WARNING: No route found for IPv6 destination :: (no default route?). This affects only IPv6
Most common process for ps command: /usr/lib64/firefox64
This application have to run with root access. Try again
Dannys-MBP:Listings dannylieu$
```

As expected, the script is unable to run because the user is not root user.

Test #3

The port knocking procedure is done with correct sequence.

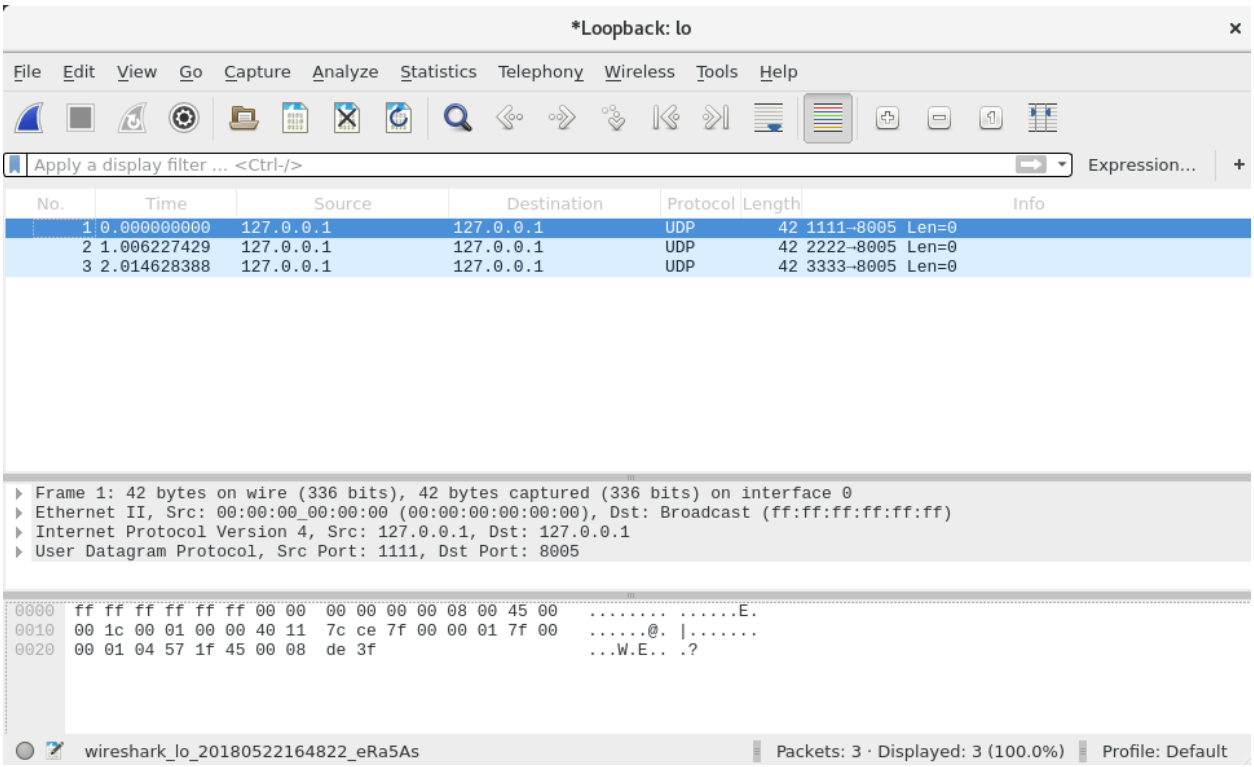
For port knocking, the configuration files for both attacker and backdoor need to have the same list of knocking ports such as [1111, 2222, 3333]. The listener port on the backdoor side will also need to define in the backdoorConfig in order for the backdoor to listen.

```
root@localhost:~/Documents/C8505/Backdoor-Python/Listings
File  Edit  View  Search  Terminal  Help
[root@localhost Listings]# python3 attacker.py
WARNING: No route found for IPv6 destination :: (no default route?). This affect
s only IPv6
Knock...1111
Knock...2222
Knock...3333
Start to enter command...
 Filter: tcp src port 8505 and dst port 8506 and src host 127.0.0.1
```

The backdoor will print a message indicating that the authentication procedure is completed.

```
root@localhost:~/Documents/C8505/Backdoor-Python/Listings
File  Edit  View  Search  Terminal  Help
[root@localhost Listings]# python3 backdoor.py
WARNING: No route found for IPv6 destination :: (no default route?). This affect
s only IPv6
Most common process for ps command: /usr/lib64/firefox64
127.0.0.1 : Knock 1
127.0.0.1 : Knock 2
127.0.0.1: Knock 3. Authetication succeed.
```

In Wireshark, we can also see the procedure happen. The attacker will send three packets with the source port is 1111, 2222, and 3333 to port 8005 of the backdoor program.

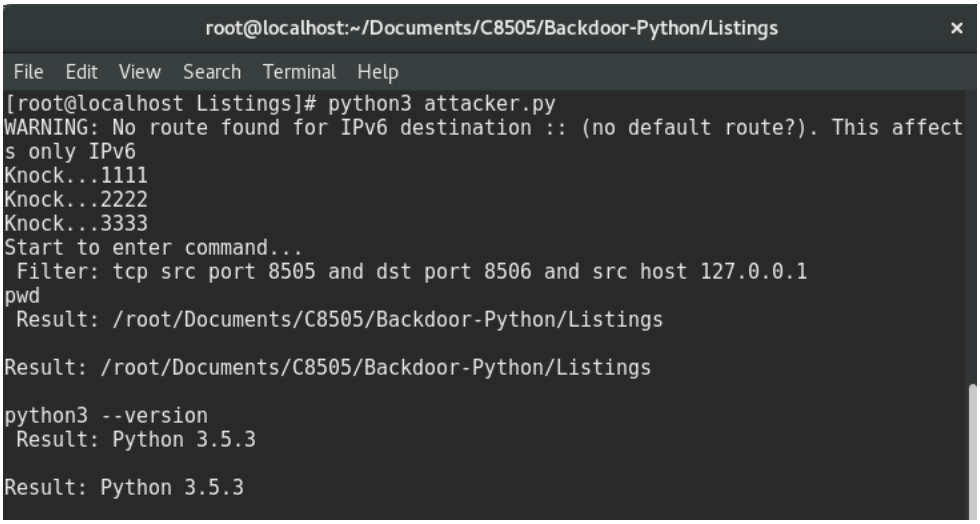


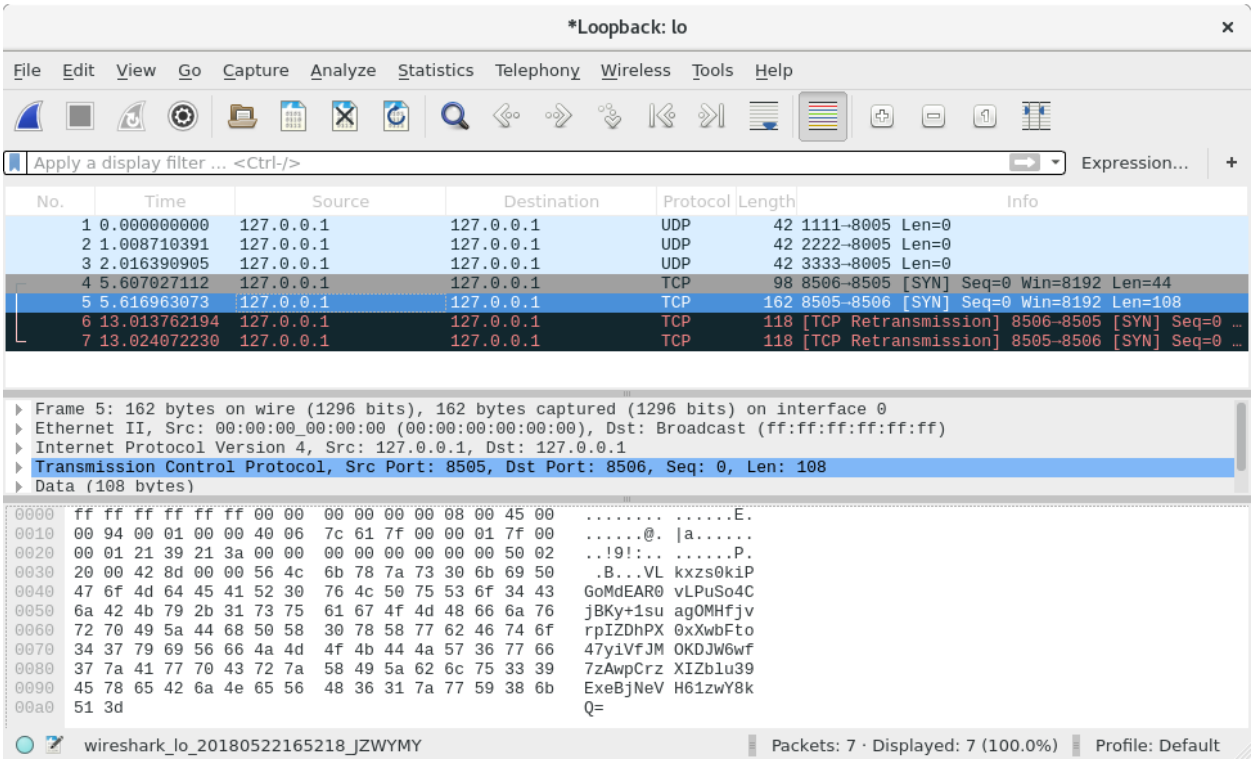
Test #4

The attack is able to send encrypted command to the backdoor. After that, the backdoor decrypt the payload to get the command and print to stdout.

After the pork knocking procedure, the user will be asked to enter a command. The command will encrypt using AES encryption and the **masterkey** (Note: the **masterkey** is the hash using md5). The payload of the packet will contain the password + the actual command.

In Wireshark, the payload of the packet will show as a random combination of character which cannot be read.





On the backdoor side, when it received the packet that contain the command. It will decrypt the packet using AES encryption and the same masterkey.

The command is printed to stdout and will be executed.

```
root@localhost:~/Documents/C8505/Backdoor-Python/Listings
File  Edit  View  Search  Terminal  Help
[root@localhost Listings]# python3 backdoor.py
WARNING: No route found for IPv6 destination :: (no default route?). This affect
s only IPv6
Most common process for ps command: /usr/lib64/firefox64
127.0.0.1 : Knock 1
127.0.0.1 : Knock 2
127.0.0.1: Knock 3. Authetication succeed.
comp8505
Executing command: pwd
Result: b'/root/Documents/C8505/Backdoor-Python/Listings\n'
comp8505
Executing command: python3 --version
Result: b'Python 3.5.3\n'
```

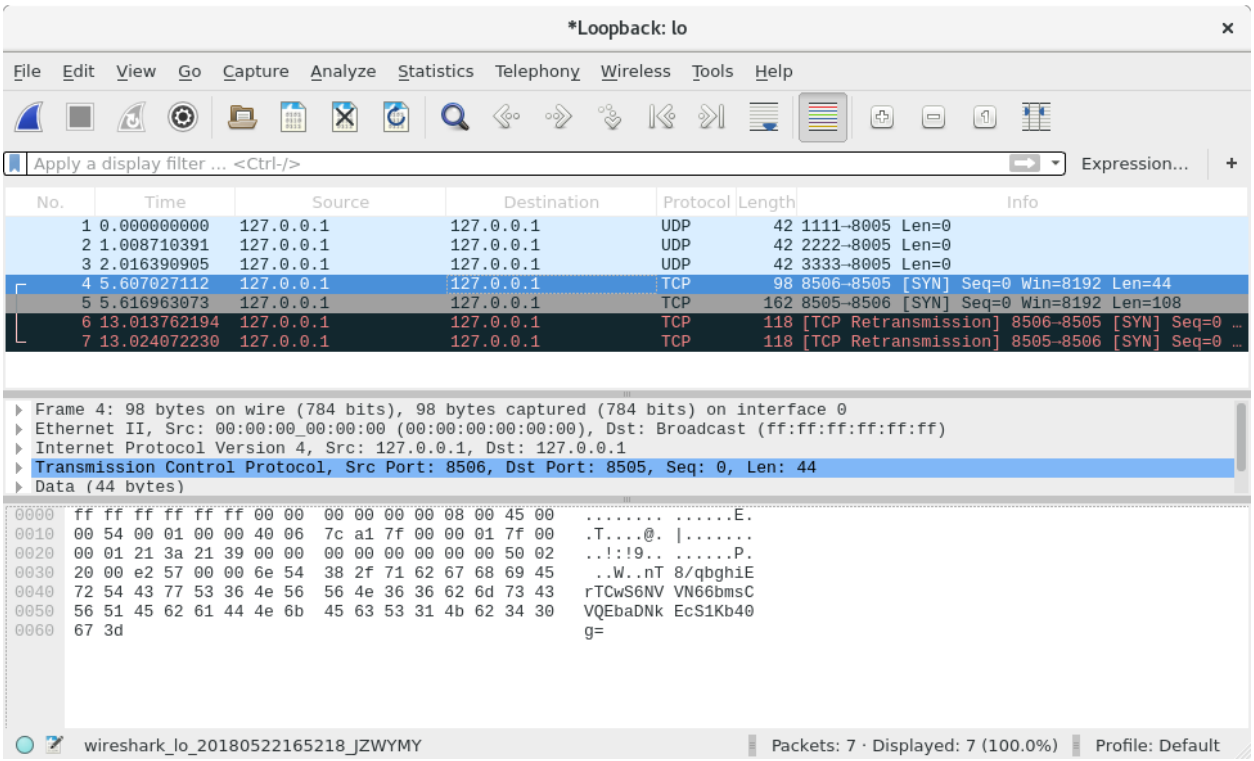
Test #5

The backdoor sends the encrypted result of the command to the attacker. After that, the attacker decrypts the payload to get the result of the command and print to stdout.

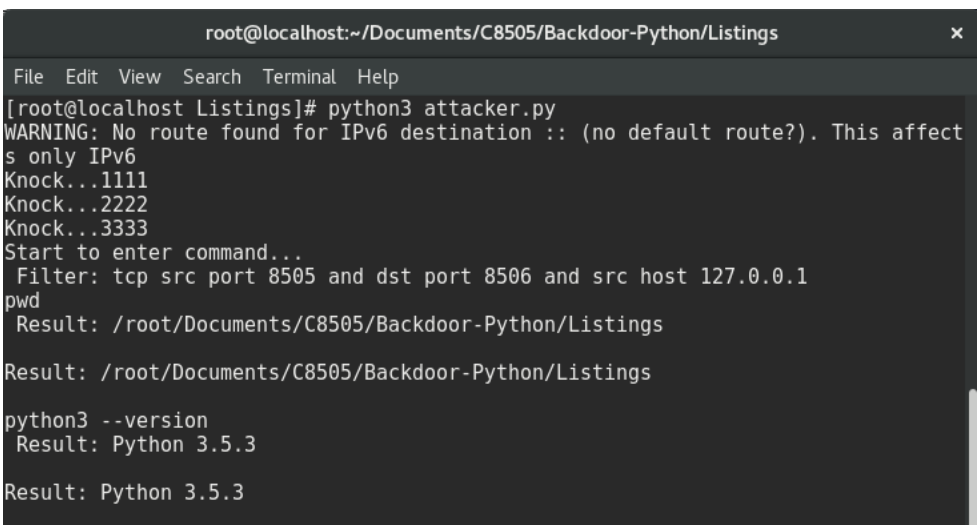
After the command is executed, the result will be encrypted using the same encryption scheme when receiving. A copy of the result is also printed to stdout.

```
root@localhost:~/Documents/C8505/Backdoor-Python/Listings
File  Edit  View  Search  Terminal  Help
[root@localhost Listings]# python3 backdoor.py
WARNING: No route found for IPv6 destination :: (no default route?). This affect
s only IPv6
Most common process for ps command: /usr/lib64/firefox64
127.0.0.1 : Knock 1
127.0.0.1 : Knock 2
127.0.0.1: Knock 3. Authetication succeed.
comp8505
Executing command: pwd
Result: b'/root/Documents/C8505/Backdoor-Python/Listings\n'
comp8505
Executing command: python3 --version
Result: b'Python 3.5.3\n'
```

In Wireshark, payload of the packet will show as a random combination of character which cannot be read.



When the attack receives the result, it will decrypt the payload to get the result and print out to stdout.



Test #6

The communication between the attacker and the backdoor will be stopped by "close" command.

Using the close command, both attacker and backdoor script will exit and print out the appropriate closing message.

Attacker

```
root@localhost:~/Documents/C8505/Backdoor-Python/Listings
File  Edit  View  Search  Terminal  Help
[root@localhost Listings]# python3 attacker.py
WARNING: No route found for IPv6 destination :: (no default route?). This affect
s only IPv6
Knock...1111
Knock...2222
Knock...3333
Start to enter command...
 Filter: tcp src port 8505 and dst port 8506 and src host 127.0.0.1
close
Attacker closed...
```

Backdoor

```
root@localhost:~/Documents/C8505/Backdoor-Python/Listings
File  Edit  View  Search  Terminal  Help
[root@localhost Listings]# python3 backdoor.py
WARNING: No route found for IPv6 destination :: (no default route?). This affect
s only IPv6
Most common process for ps command: /usr/lib64/firefox64
127.0.0.1 : Knock 1
127.0.0.1 : Knock 2
127.0.0.1: Knock 3. Authetication succeed.
comp8505
Executing command: close
Backdoor closed...
```