# Remote SSH Tunneling with Plink.exe

U.Y. · Follow

So, let's start it:

Command: `kali@kali:~$ sudo service ssh start`

And now check with *ss tool* to see the difference.

Running the following command shows us that *sshd* is active and running:

Command: `kali@kali:~$ sudo ss -antlp | grep sshd`

and download the file:



# Medium

## Sign up to discover human stories that deepen your understanding of the world.
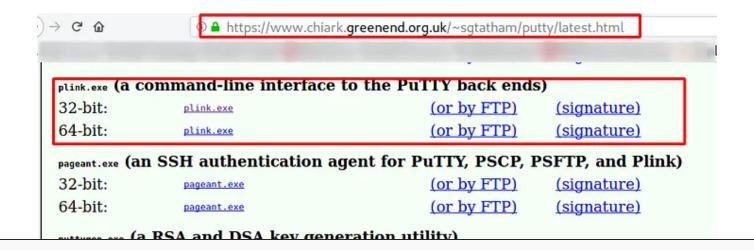
### Free

✓ Distraction-free reading. No ads.

✓ Organize your knowledge with lists and highlights.

✓ Tell your story. Find your audience.

### ✦ Membership

✓ Read member-only stories

✓ Support writers you read most

✓ Earn money for your writing

✓ Listen to audio narrations

✓ Read offline with the Medium app

Getting OS Architecture using wmic

Holy Cow! I have 32-bit OS architecture.

After getting the architecture information, download 32-bit or 64 bit compiled plink.exe (Type of exe is independent of what we are going to do here, we just want to have the correct executable).

. . .

# Medium

## Sign up to discover human stories that deepen your understanding of the world.

**Free**

✓ Distraction-free reading. No ads.

✓ Organize your knowledge with lists and highlights.
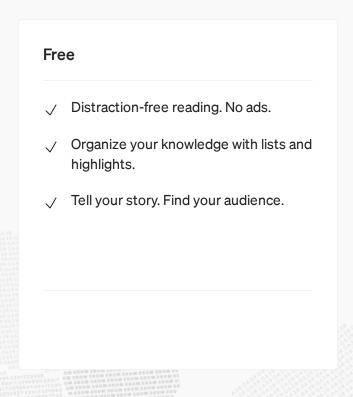
✓ Tell your story. Find your audience.

**✦ Membership**

✓ Read member-only stories

✓ Support writers you read most

✓ Earn money for your writing

✓ Listen to audio narrations
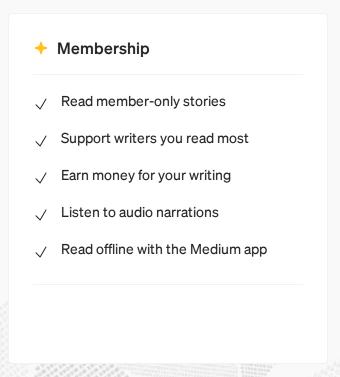
✓ Read offline with the Medium app

```
C:\Windows\system32>netsh advfirewall firewall add rule name="BlockSecretServerPort"
protocol=TCP dir=in localport=8090 action=block
Ok.
```

Blocking a port (inbound)

Let's check if we could block it successfully or command prompt lied to us:

Use the following command and scroll to find your new rule:

`C:\Windows\system32>netsh firewall show config`

# Medium

## Sign up to discover human stories that deepen your understanding of the world.

### Free

✓ Distraction-free reading. No ads.

✓ Organize your knowledge with lists and highlights.

✓ Tell your story. Find your audience.

### ✦ Membership

✓ Read member-only stories

✓ Support writers you read most

✓ Earn money for your writing

✓ Listen to audio narrations

✓ Read offline with the Medium app

# Medium

Sign up to discover human stories that deepen your understanding of the world.

## Free

✓ Distraction-free reading. No ads.

✓ Organize your knowledge with lists and highlights.

✓ Tell your story. Find your audience.

## ✦ Membership

✓ Read member-only stories

✓ Support writers you read most

✓ Earn money for your writing

✓ Listen to audio narrations
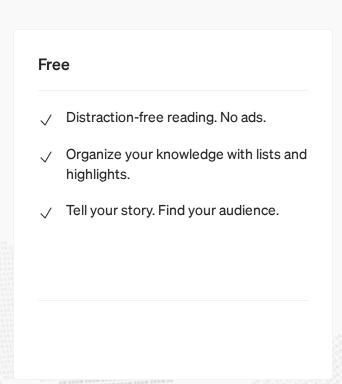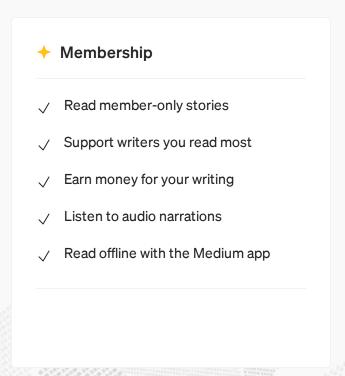
✓ Read offline with the Medium app

**Important notes about the above command:**

1. We will run this command on the machine (i.e. Windows machine) where we blocked the server and enable a remote ssh tunnel.

2. All variables above starting with "MY" (e.g. MYUSERNAME) are related to your development PC dependent (the PC where we used browser to check the connection initially before blocking the port, i.e. Kali machine).

3. You can choose any port for the ssh server side (Kali machine) but the port
~~you tunnel on your side must be the port you want to make reachable from~~

# Medium

## Sign up to discover human stories that deepen your understanding of the world.

Remote SSH Tunneling using plink.exe

# Medium

Sign up to discover human stories that deepen your understanding of the world.

Successful SSH Remote Tunnel Session

Lastly, you can call **plink.exe** on command prompt to see what other options we have. I hope this exercise was useful and you can easily use it whenever needed.

# Medium

## Sign up to discover human stories that deepen your understanding of the world.

**Free**

✓ Distraction-free reading. No ads.

✓ Organize your knowledge with lists and highlights.

✓ Tell your story. Find your audience.

✦ **Membership**

✓ Read member-only stories

✓ Support writers you read most

✓ Earn money for your writing

✓ Listen to audio narrations

✓ Read offline with the Medium app