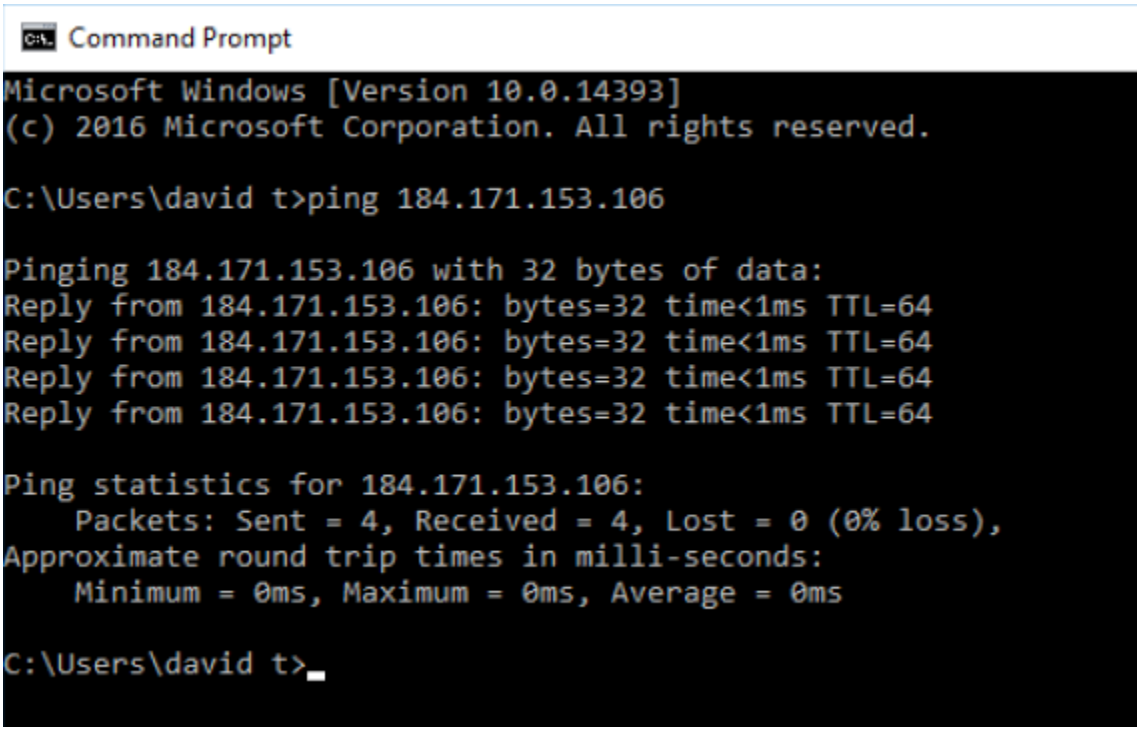
Remote Access Lab

We will explore briefly two popular remote solutions in the field: SSH & RDP.

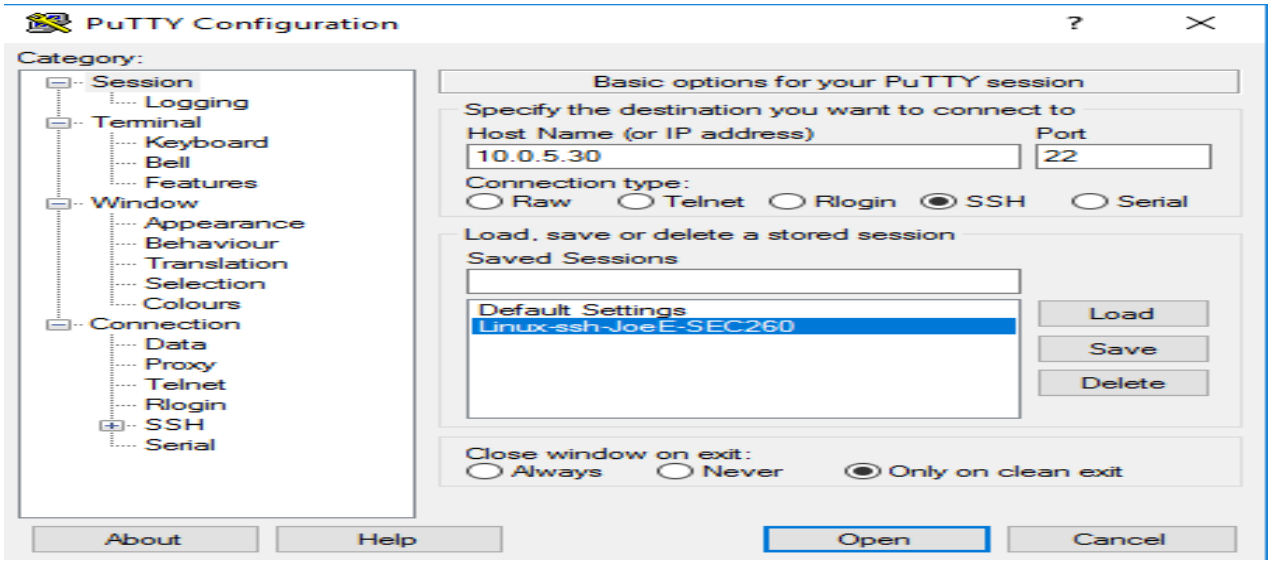
I. Pre-flight check -

1. First, got to setup our "cabling" …
2. Setup Win10 & Kali VMs vmnet connections to the same *custom* vmnet which is NAT'd.
3. Snapshot Win10 & Kali VMs for good luck.
4. Power both VMs up.
5. Cool, now make sure both VMs can communication with each other.
6. *Snip* of both VMs successfully using only ONE ping request to each other using ping syntax via their Command Line Interface (CLI). And no, CTL-C does not count.
    a. Hint: Linux & Windows has different command-line arguments for this, but they rather similar, so completely researchable & best test them prior to submitting.

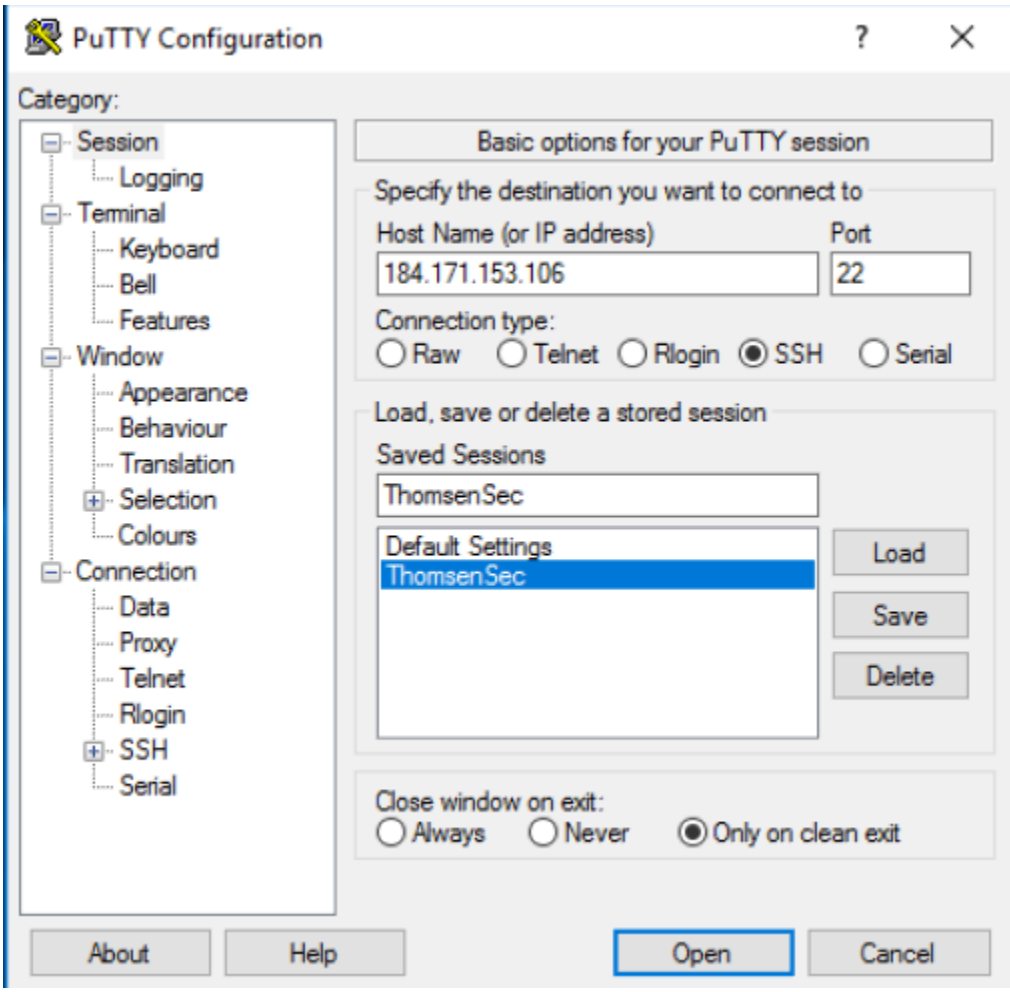


II. Remotely Accessing Linux via Windows

1. On your Win10 VM, download Putty: https://putty.org/ and save it. FYI: Windows does not do 32-bit any more by default. Now install it w/ lots of default settings.
2. Launch Wireshark on Win10 VM.
3. Launch Putty, and insert your Linux IP for port 22, then name & save that session similar to screenshot above with the according name change:



a. *Snip* of your successfully stored SSH session settings.

b. 

4. Now let's load the SSH stored session, trust the Security Alert, and insert Linux's username & password.

5. Sweet, you are now remoting to your Linux VM from your Win10 VM via SSH thru Putty.

```
dthom@kali: ~
login as: dthom
dthom@184.171.153.5's password:
Linux kali 5.9.0-kali1-amd64 #1 SMP Debian 5.9.1-1kali2 (2020-10-29) x86_64

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
┌─(Message from Kali developers)
│
│ We have kept /usr/bin/python pointing to Python 2 for backwards
│ compatibility. Learn how to change this and avoid this message:
│ → https://www.kali.org/docs/general-use/python3-transition/
│
└─(Run "touch ~/.hushlogin" to hide this message)
┌──(dthom㉿kali)-[~]
└─$
```

6. In Linux CLI, we can execute use several commands on one line at once for efficiency & straight up coolness factor.
7. Insert the following Linux syntax on one line, changing your name accordingly:

```
root@derp2:~# echo "howdy"; mkdir testdir01; cd testdir01; touch testfile01; echo
"Woot, parked some data via one megaline for SEC260! - JoeE" > testfile01; ls -l >
> testfile01; pwd >> testfile01; date >> testfile01; cat testfile01
howdy
Woot, parked some data via one megaline for SEC260! - JoeE
total 4
-rw-r--r-- 1 root root 59 Nov 19 19:44 testfile01
/root/testdir01
Mon Nov 19 19:44:23 EST 2018
```

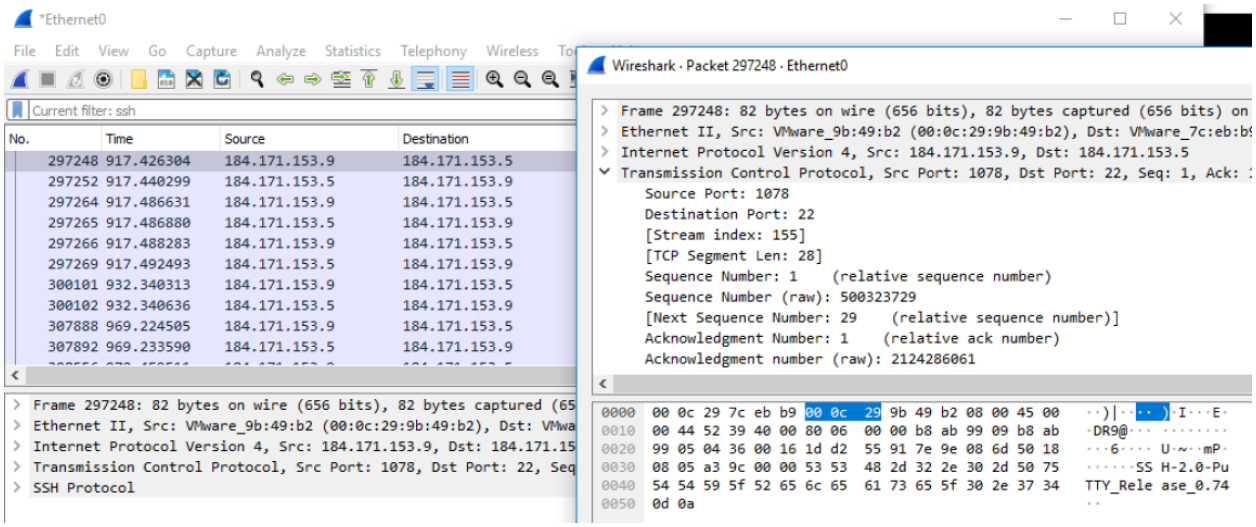
8. *Snip* your command and its output.

```
┌──(dthom㉿kali)-[~/testdir01/testdir01]
└─$ echo "howdy"; mkdir testdir01; cd testdir01; touch testfile01; echo "Woot, parked some data via one megaline for SEC2
50! -DavidT" > testfile01; ls -l >> testfile; pwd >> testfile01; date >> testfile01; cat testfile01
howdy
Woot, parked some data via one megaline for SEC250! -DavidT
/home/dthom/testdir01/testdir01/testdir01
Sun 25 Apr 2021 10:56:14 PM EDT

┌──(dthom㉿kali)-[~/testdir01/testdir01/testdir01]
└─$
```

9. Question: How many commands were utilized above? ___11____
10. Question: What current directory are you in? _____testdir01_____
11. Via Wireshrark, find the default port SSH used.

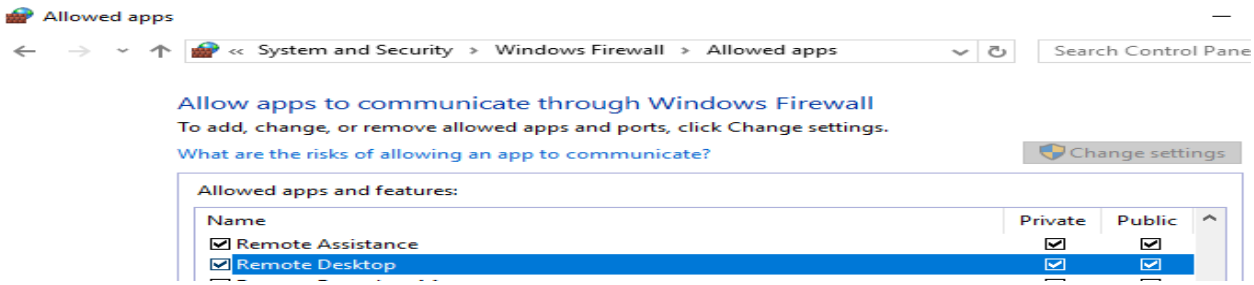
12. *Snip* a SSH's packet details displaying its default port by filtering your Win10 VM's IP address.



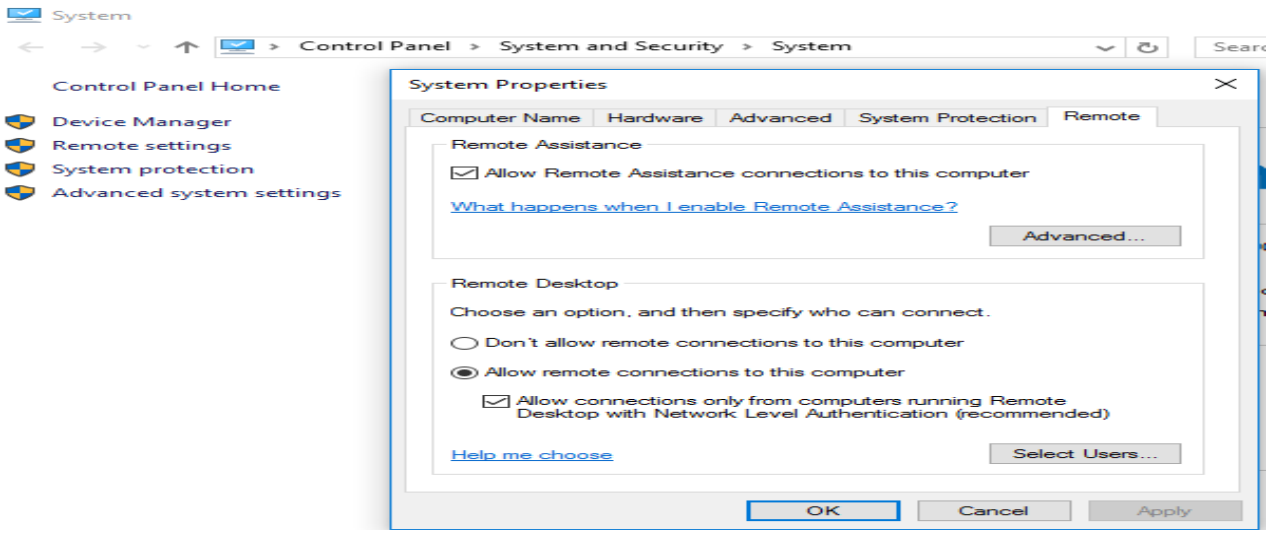
13. When done, type 'exit' to close your Putty'd SSH session.

III. Remotely Accessing Windows via LInux

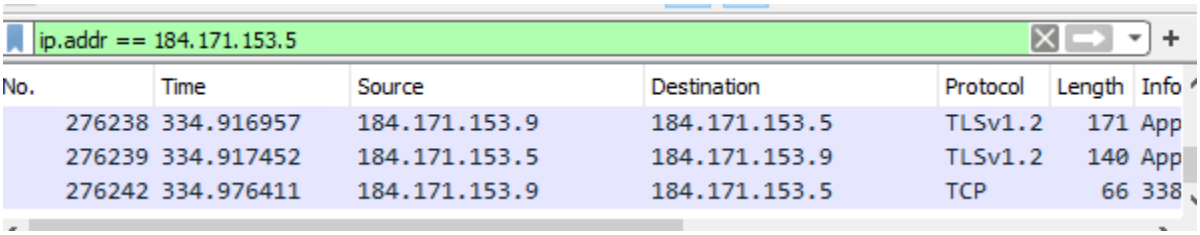
1. We know Win10 is all GUI, so we're going to access it via Linux.
2. First we need to allow Remote Desktop thru our firewall, as below:



3. While we're on Win10, we'll need to enable Remote Desktop:

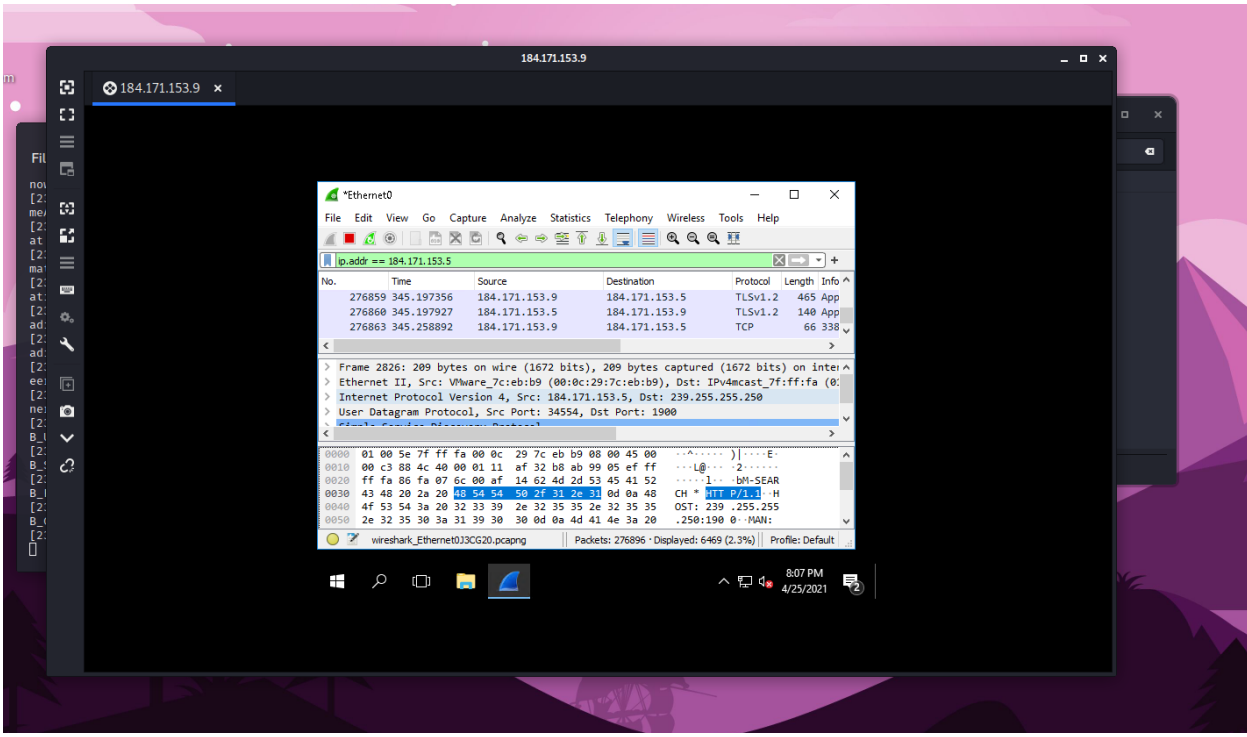


4. Launch Wireshark on Win10 VM with a display filtering only for your Linux IP.
    a. *Snip* your Wireshark display filter for Linux IP only.

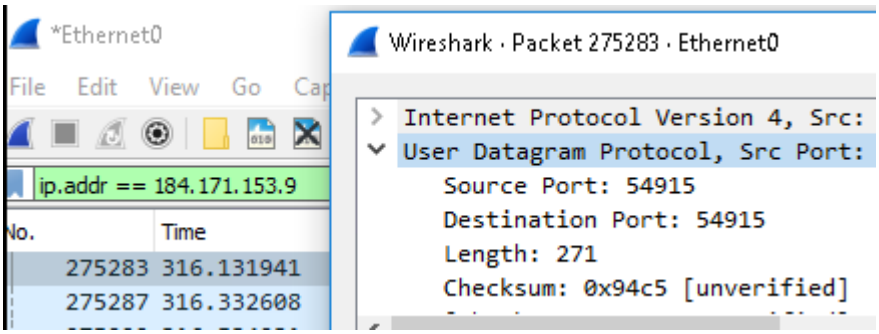


5. On Linux, run "remmina" and a GUI session of Remmina should appear. It's a live session, meaning if you close the CLI, it'll close the program as well.
6. Insert your Win10 IP in Remmina's Remote Desktop client. Hopefully you recall your Win10 username & password!

a. *Snip* a portion of your Windows & Remmina desktop



7. Via Wireshrark, find the default port RDP used.
   a. *Snip* a RDP's packet details displaying its default port by filtering your Win10 VM's IP address.



8. Ctrl+C to close the running Remmina CLI session.