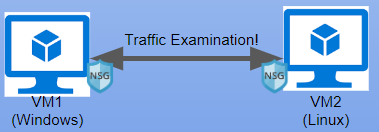
[](https://camo.githubusercontent.com/2f5a3f50869d43a2e6c1e771b7ea2026d5966d49c540a2df3438b3f8292d183a/68747470733a2f2f692e696d6775722e636f6d2f55613775646f532e706e67)

**Network Security Groups (NSGs) and Inspecting Traffic Between Azure Virtual Machines**

In this tutorial, we observe various network traffic to and from Azure Virtual Machines with Wireshark as well as experiment with Network Security Groups.

**Environments and Technologies Used**

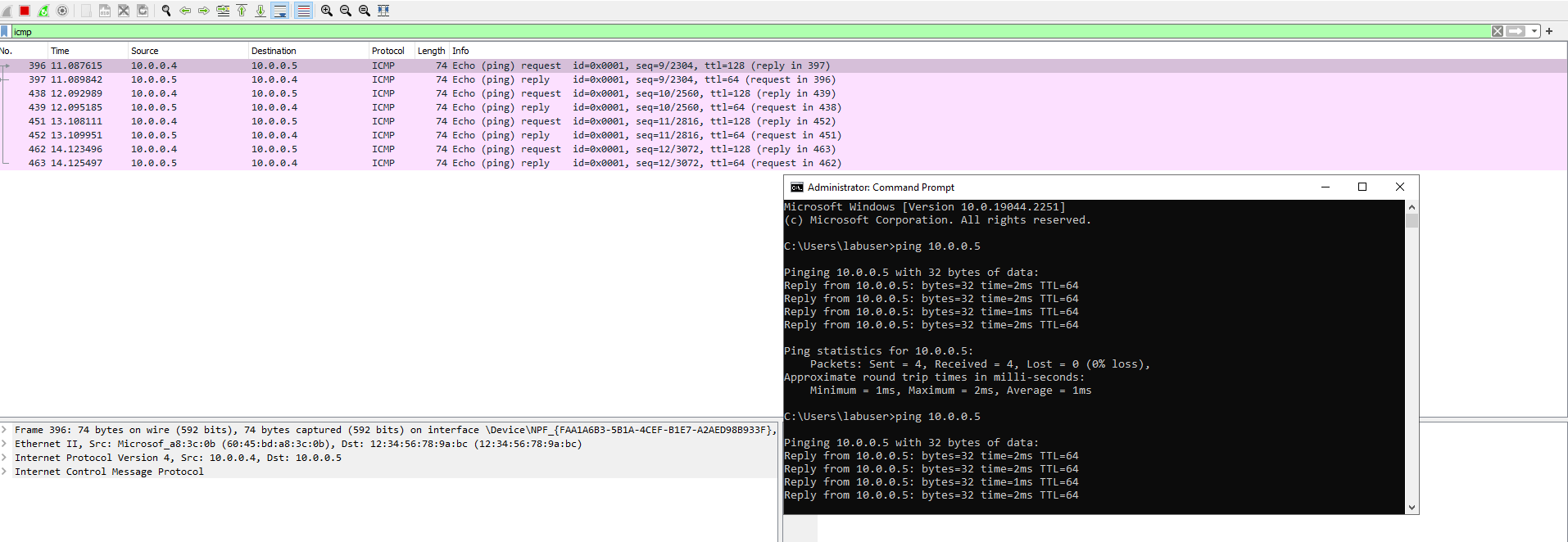
* Microsoft Azure (Virtual Machines)
* Remote Desktop
* Various Command-Line Tools
* Various Network Protocols (SSH, RDH, DNS, HTTP/S, ICMP)
* Wireshark (Protocol Analyzer)

**Operating Systems Used**

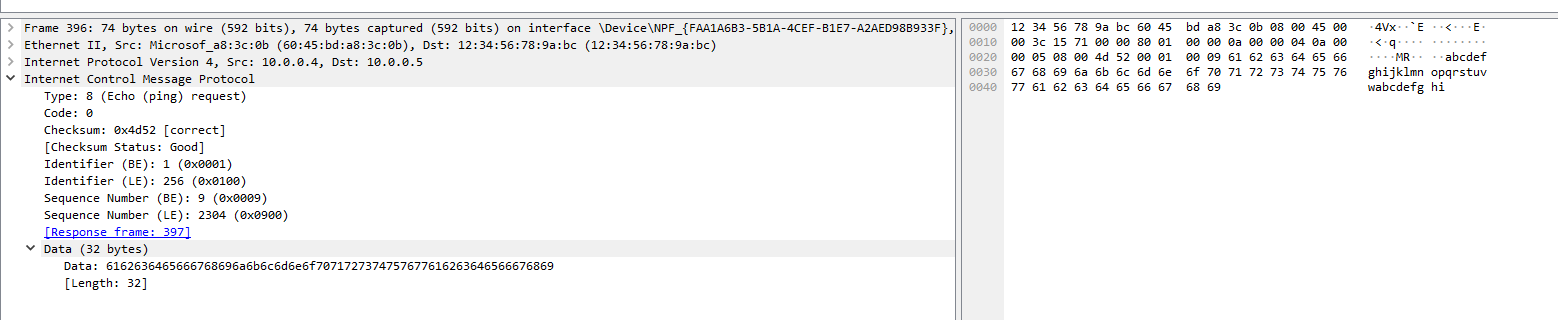
* Windows 10 (21H2)
* Ubuntu Server 20.04

**Actions and Observations**

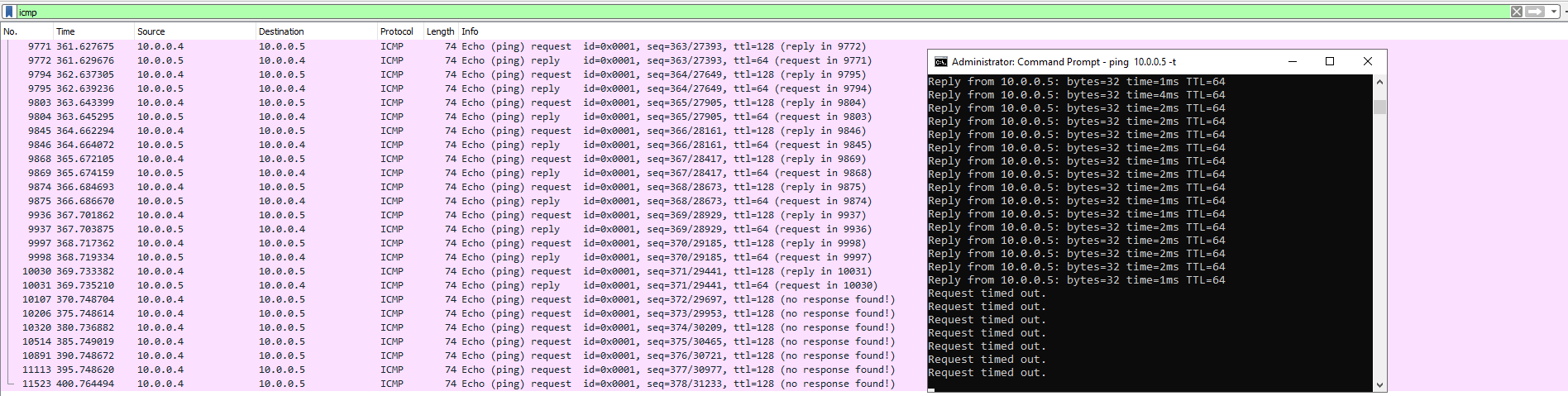
Welcome to my tutorial on Network Security Groups and Inspecting Network Protocols. First you will need to create two VMs on Azure. One machine will be a Linux machine and the other will be a Windows 10 machine. Both will have two cpus and they must be on the same VNET. Once that is done go on the Windows machine and download Wireshark. I will attatch a link to the wireshark download. <https://www.wireshark.org/download.html> Once installed open Wireshark and filter for ICMP Traffic only. ICMP is a network layer protocol that relays messages concerning network connection issues. Ping uses this protocol, ping tests connectivity between hosts. When we filter wirehsark to only capture ICMP packets and ping the private IP address of our linux machine we can visually see the packets on wireshark.

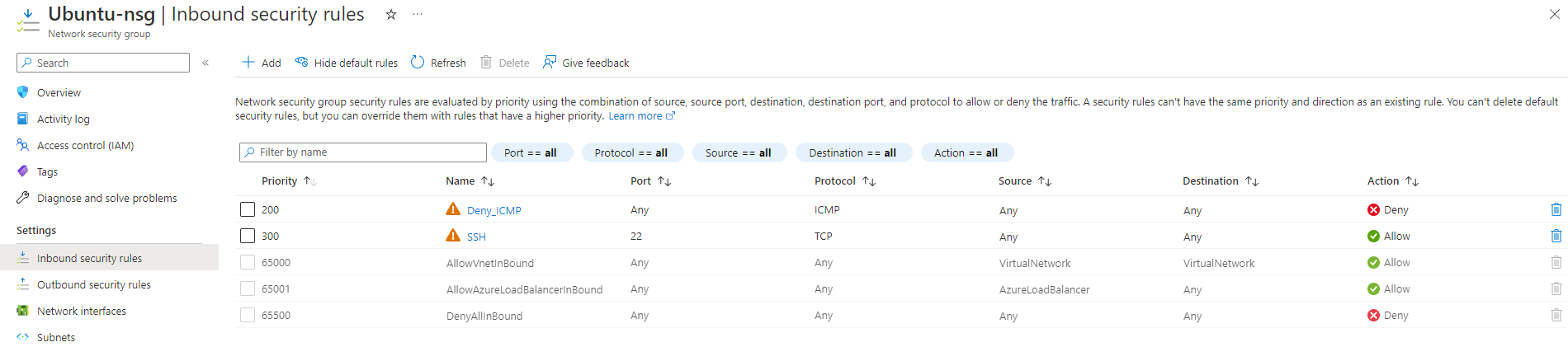
[](https://camo.githubusercontent.com/a44d2dd8241c52f586b54b30ad2c6c3cd93be5332e5afea393085b552f5d6e75/68747470733a2f2f692e696d6775722e636f6d2f494955536878702e706e67)

We can inspect each individual packet and see the actual data that is being sent in each ping. the picture below demonstrates just that.

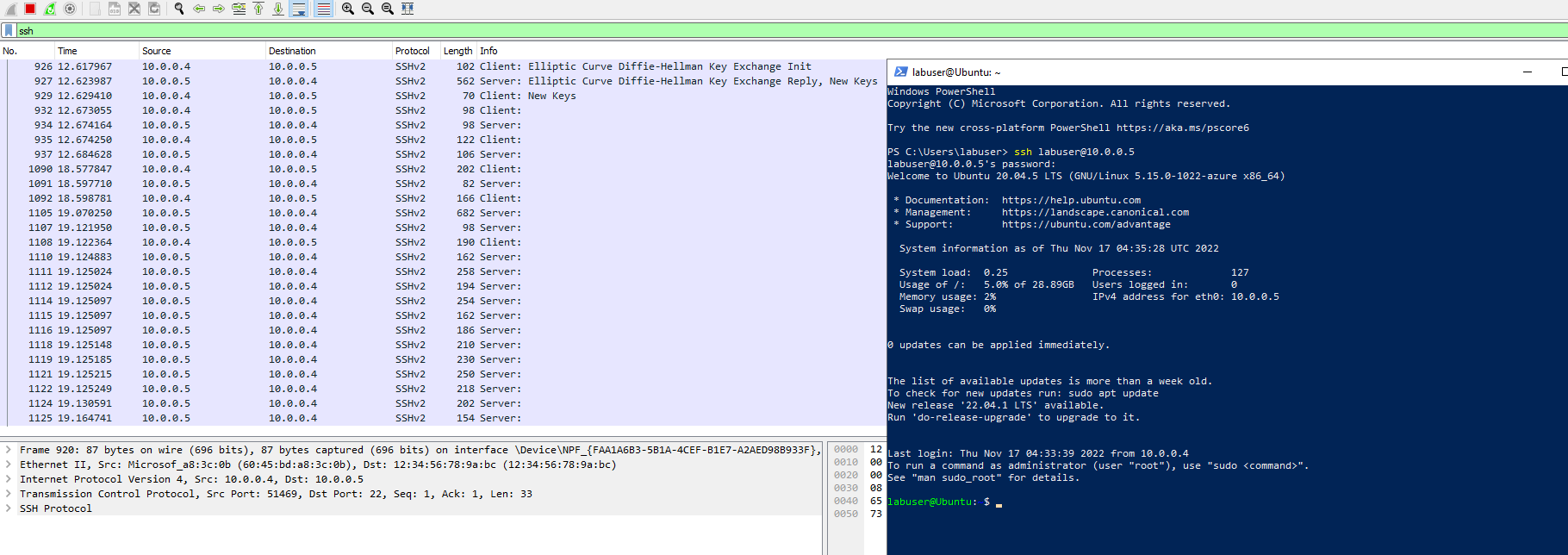
[](https://camo.githubusercontent.com/0f4f386d55d502aafcc28909f8011e7aa91c355bc2fb7cd40bd8516017cf559d/68747470733a2f2f692e696d6775722e636f6d2f474c78534947332e706e67)

In the next portion of the lab we will perpetually ping the Linux machine with the command ping -t. This will continually ping the machine until we decide to stop it, while the Windows machine is pinging the Linux machine we will go to the Linux machine and block inbound ICMP traffic on it's firewall. Once we do that we will stop recieving echo replys from the Linux machine. We will block ICMP by creating a new Network Security Group on the Linux machine that will be set to block ICMP. We can allow the traffic by allowing ICMP on the Linux Network Security Groups page on Azure.

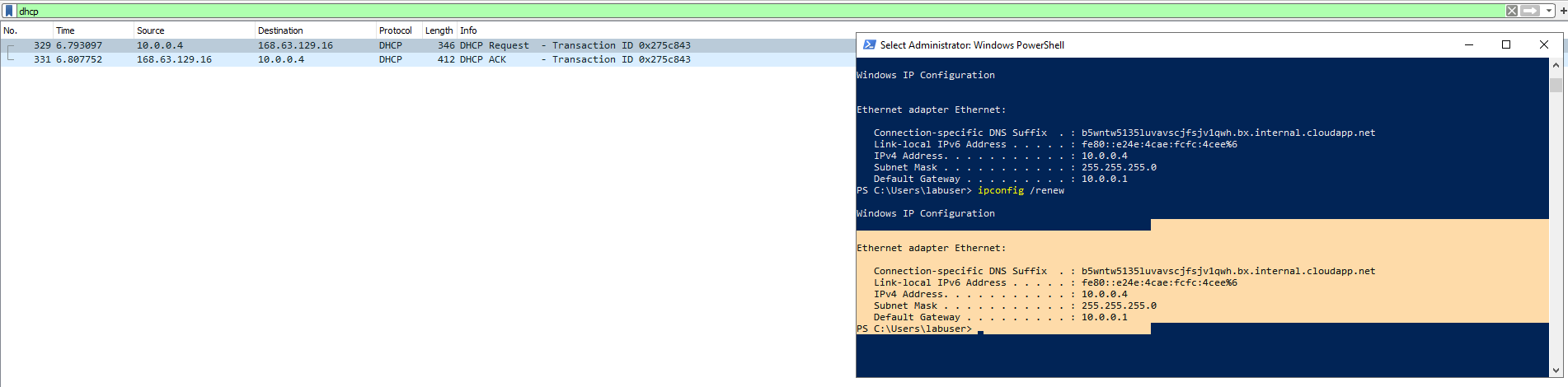
[](https://camo.githubusercontent.com/ce200b682d31bed8fc68a34f3f63b733548c5e1b6f42f74eef3356ab32ef593f/68747470733a2f2f692e696d6775722e636f6d2f3576584f3735522e706e67)

[](https://camo.githubusercontent.com/184a8c53b6448f632c28157891141b8172c0403044cd9aa2dea977ee87ddbfa8/68747470733a2f2f692e696d6775722e636f6d2f41736c3830744e2e706e67)

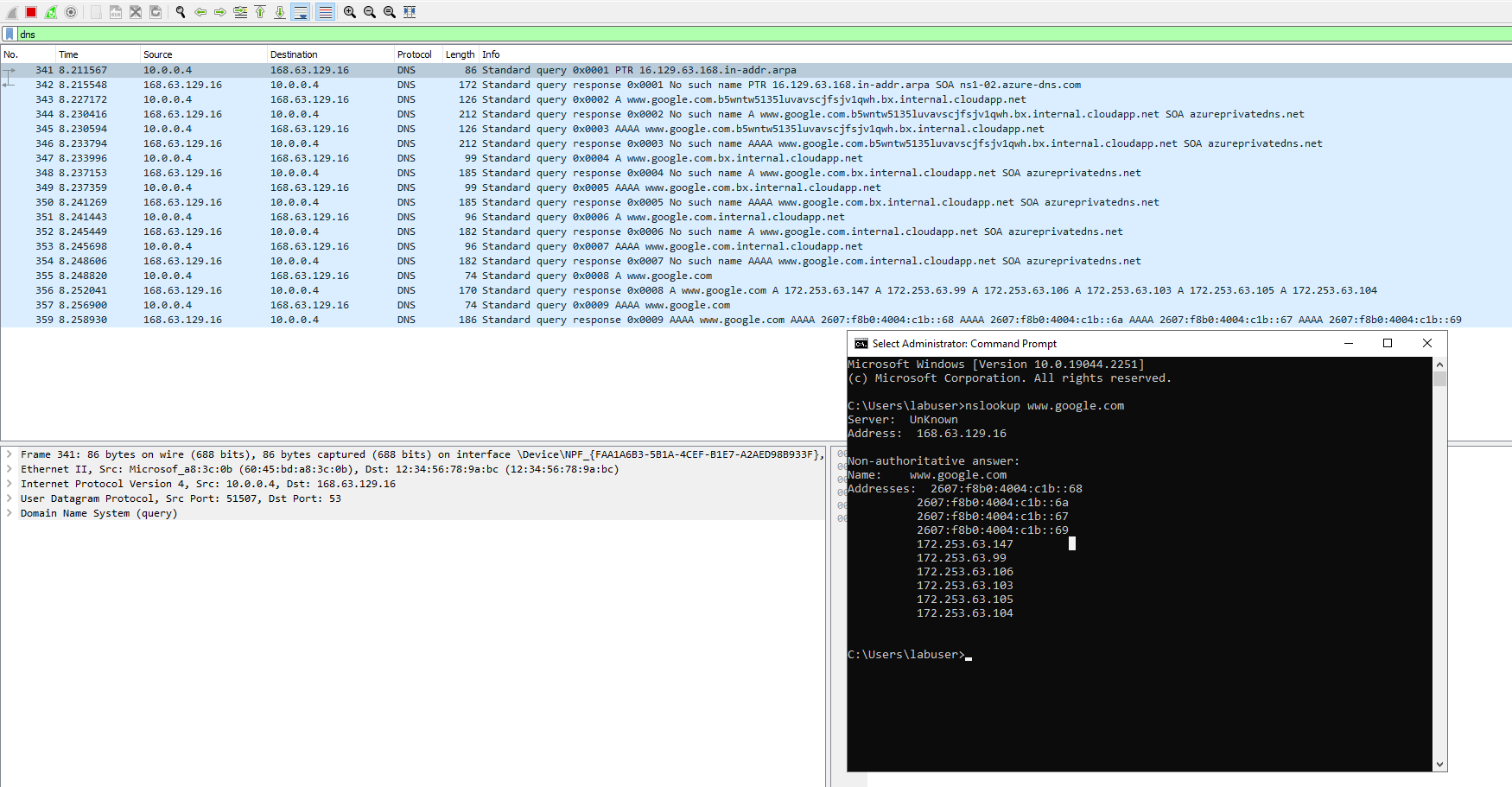
Next we will use our Windows machine to SSH to the Linux machine. SSH has no GUI it just gives the user access to the machines CLI. We will set the wireshark filter to capture SSH packets only. When we ssh into the Linux machine with the command prompt "ssh labuser@10.0.0.5" we can see that wireshark starts to immediately capture SSH packets.

[](https://camo.githubusercontent.com/a3465cf058ace227f9c99a9d45b8d90ed173e5542b16c11d70d58e9fe7a6a4e0/68747470733a2f2f692e696d6775722e636f6d2f7a7465523431722e706e67)

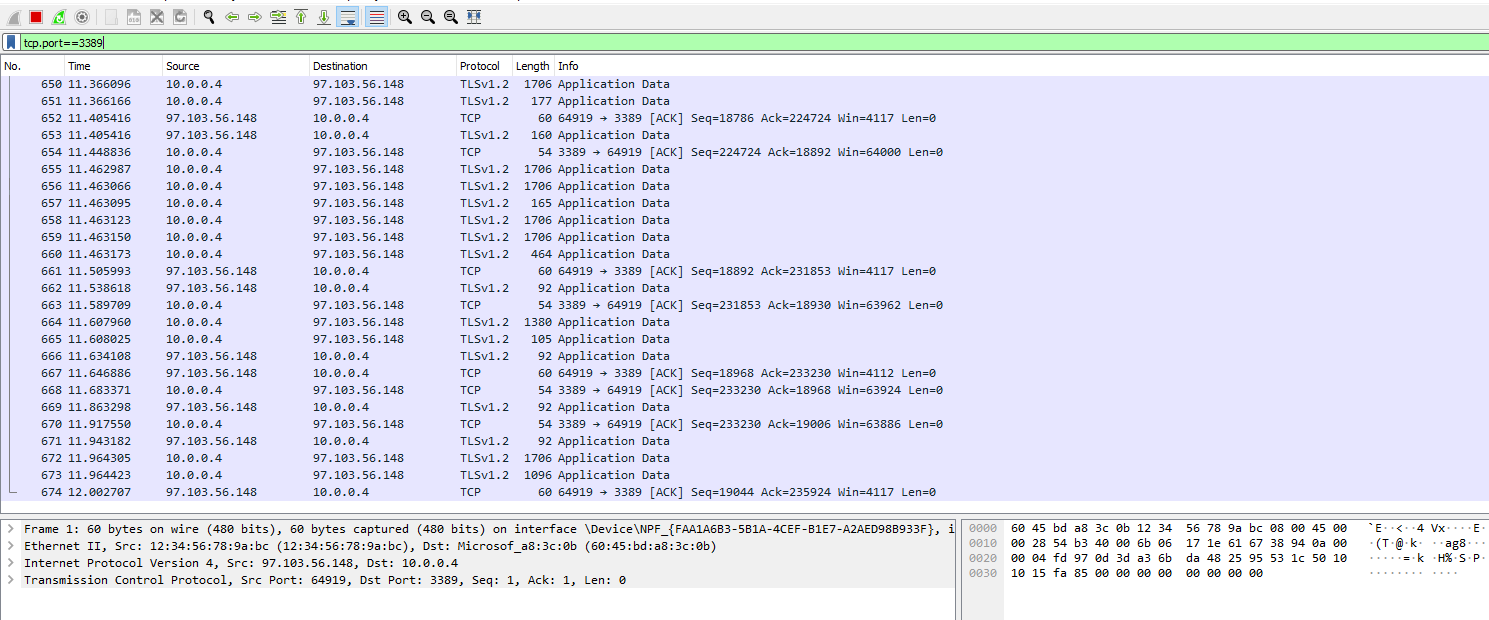
Now we will use wireshark to filter for DHCP. DHCP is the Dynamic Host Configuration Protocol this works on ports 67/68. It is used to assign IP addresses to machines. We will request a new ip address with the command "ipconfig /renew". Once we enter the command wireshark will capture DHCP traffic.

[](https://camo.githubusercontent.com/b0c8f3e5b05889d9fef07ac92b35cb6adf7c939e0b2358de7b9a5c1a65520ca2/68747470733a2f2f692e696d6775722e636f6d2f765538667051662e706e67)

Time to filter DNS traffic. We will set wireshark to filter DNS traffic. We will initiate DNS traffic by typing in the command "nslookup [www.google.com](http://www.google.com/)" this command essentially asks our DNS server what is google's IP address.

[](https://camo.githubusercontent.com/87f5d3f6f311072b4536834249acfb886eed7d1bf9409c95fabc727784e40a95/68747470733a2f2f692e696d6775722e636f6d2f564d63776d734f2e706e67)

Lastly we will filter for RDP traffic. When we enter tcp.port==3389 traffic is spammed non stop because we are using Remote Desktop Protocol to connect to our Virtual Machine.

[](https://camo.githubusercontent.com/35af87042b49acf36011d15642597cf4f0bd7c3658899aed798082abe5b69a18/68747470733a2f2f692e696d6775722e636f6d2f567858477636582e706e67)