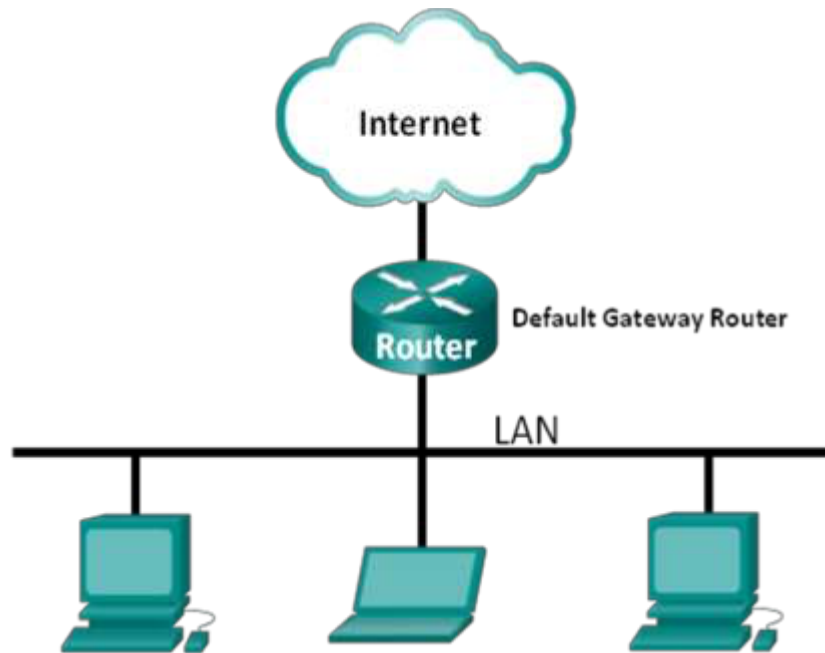


# Lab – How to analyze Network Traffic with Wireshark

## Topology



## Background / Scenario

Wireshark is a software protocol analyzer, or "packet sniffer" application, used for network troubleshooting, analysis, software and protocol development, and education. As data streams travel back and forth over the network, the sniffer "captures" each protocol data unit (PDU) and can decode and analyze its content according to the appropriate RFC or other specifications.

Wireshark is a useful tool for anyone working with networks and can be used with most labs in the CCNA courses for data analysis and troubleshooting. In this lab, you will use Wireshark to capture ICMP data packet IP addresses and Ethernet frame MAC addresses.

NB: If you perform this exercise on your personal computer (not in the ISEP lab) you can ignore the existence of the virtual machine. Some instructions should also be personalized depending to your OS.

## Part 1: Ping a local machine

In Part 1 of this lab, you will ping another PC on the LAN and capture ICMP requests and replies in Wireshark. You will also look inside the frames captured for specific information. This

analysis should help to clarify how packet headers are used to transport data to their destination.

## Step 1: Retrieve your PC's interface addresses.

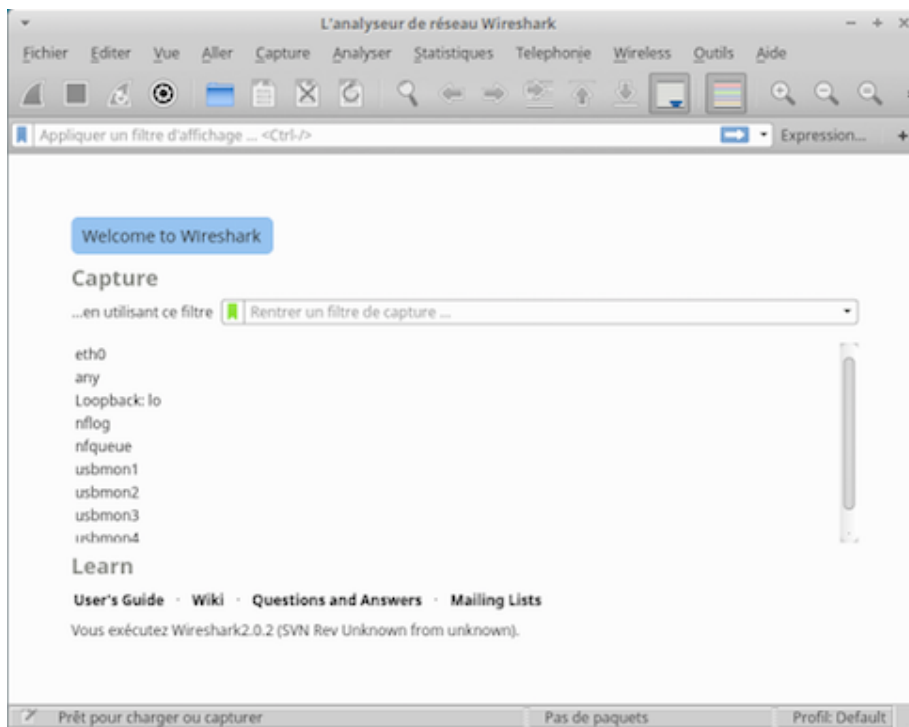
For this lab, you will need to retrieve your PC's IP address and its network interface card (NIC) physical address, also called the MAC address.

- a) Open a command window : menu → accessoires → emulateur de terminal
- b) Type the command **ifconfig**. Note your IP and physical (MAC) address
- c) What is the format of these addresses (how many bits)
- d) Note the network mask. What is it useful for?
- e) How many hosts can we have at most in this LAN ?

## Step 2: Start Wireshark and begin capturing data.

For security reasons Wireshark is installed on the windows virtual machine (VM)

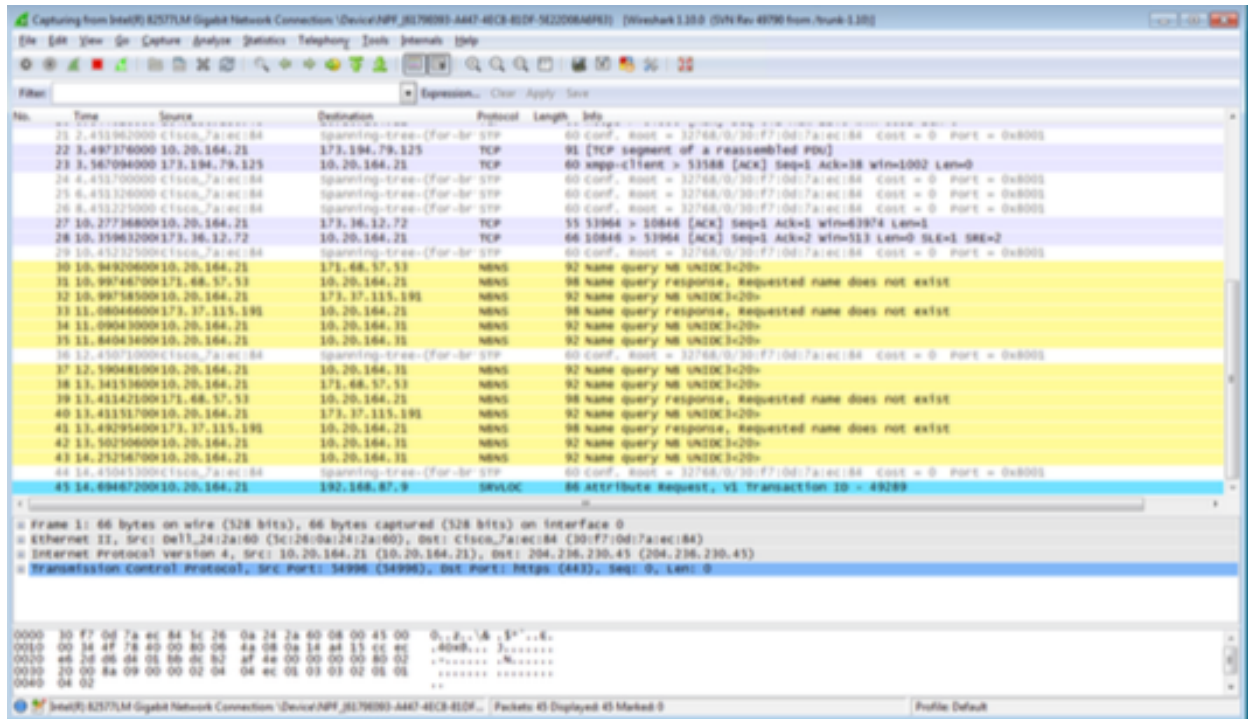
- a) Start the windows VM: **menu → ISEP → VM windows**
- b) Start Wireshark.



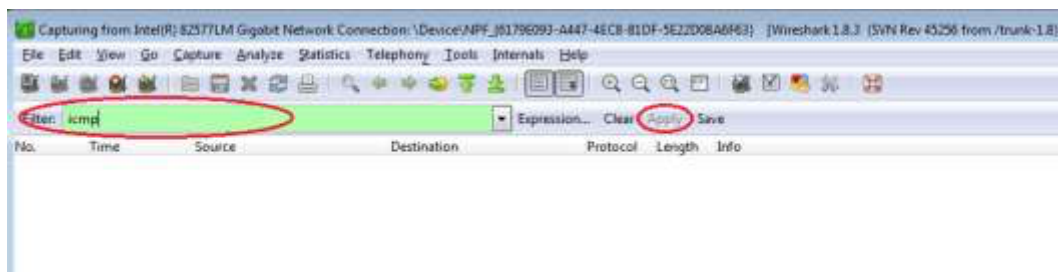
- c) Display the list of interfaces : menu « **capture** » → « **options** »
- d) Note the IP address of the interface « connexion au réseau local ».

- e) Select the interface « connexion au réseau local » and start the capture : « **capture** » → « **démarrer** »

Information will start scrolling down the top section in Wireshark. The data lines will appear in different colors based on protocol.



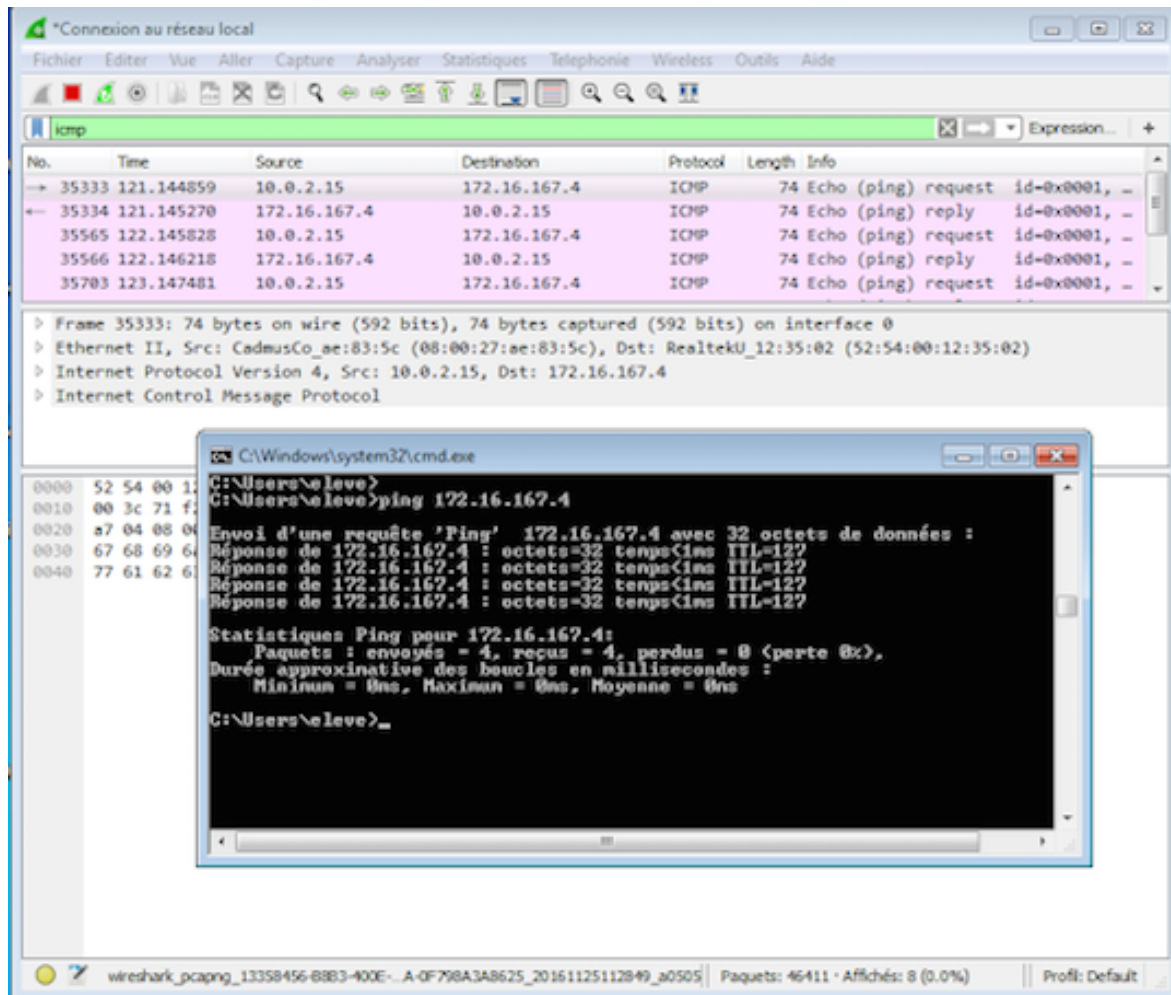
This information can scroll by very quickly depending on what communication is taking place between your PC and the LAN. We can apply a filter to make it easier to view and work with the data that is being captured by Wireshark. For this lab, we are only interested in displaying ICMP (ping) PDUs. Type `icmp` in the Filter box at the top of Wireshark and press Enter or click on the Apply button to view only ICMP (ping) PDUs.



This filter causes all data in the top window to disappear, but you are still capturing the traffic on the interface. From your windows virtual machine, ping the IP address of the linux machine of your neighbors. Notice that you start seeing data appear in the top window of Wireshark again.

NB: if you are working on your personal computer, find the IP address of your gateway (a fast

research on google will allow you to find a method suitable for your OS) and start the ping to the gateway (instead of the neighbor). Your gateway is in your LAN.

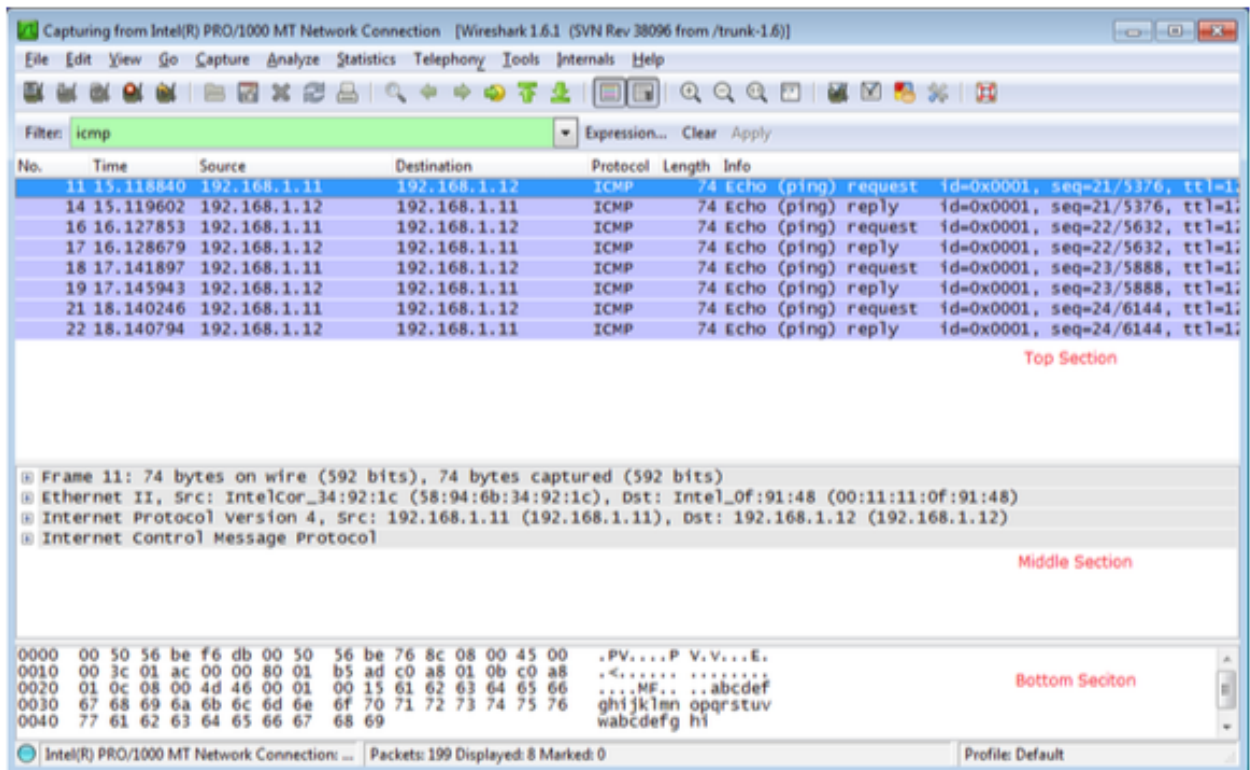


f) Stop capturing data by clicking the **Stop Capture** icon.

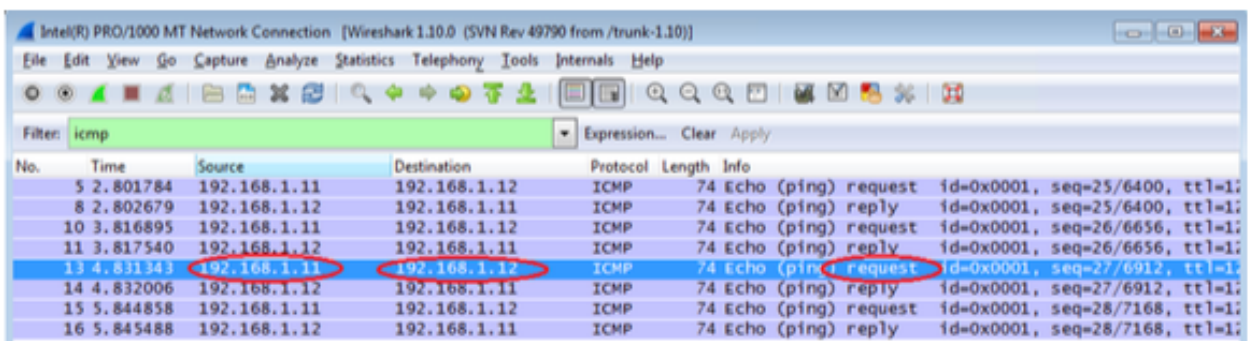
What is the ping request useful for?

### Step 3: Examine the captured data.

In Step 3, examine the data that was generated by the ping requests of your team member's PC. Wireshark data is displayed in three sections: 1) The top section displays the list of PDU frames captured with a summary of the IP packet information listed, 2) the middle section lists PDU information for the frame selected in the top part of the screen and separates a captured PDU frame by its protocol layers, and 3) the bottom section displays the raw data of each layer. The raw data is displayed in both hexadecimal and decimal form.

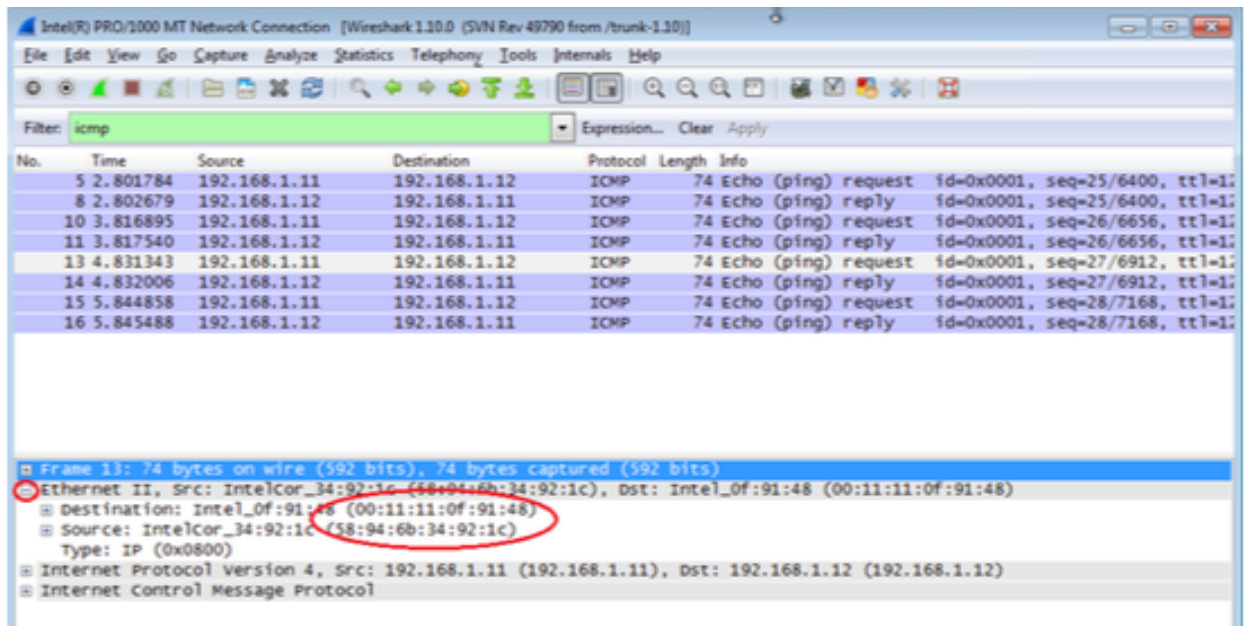


- a) Click the first ICMP request PDU frames in the top section of Wireshark. Notice that the Source column has your PC's IP address, and the Destination contains the IP address of the machine you pinged.



- b) With this PDU frame still selected in the top section, navigate to the middle section. Click the plus sign to the left of the Ethernet II row to view the Destination and Source MAC addresses.





Does the Source MAC address match your PC's interface?

Does the Destination MAC address in Wireshark match your team member's MAC address (or your gateway if you pinged the gateway)?

How is the MAC address of the pinged machine obtained by your PC?

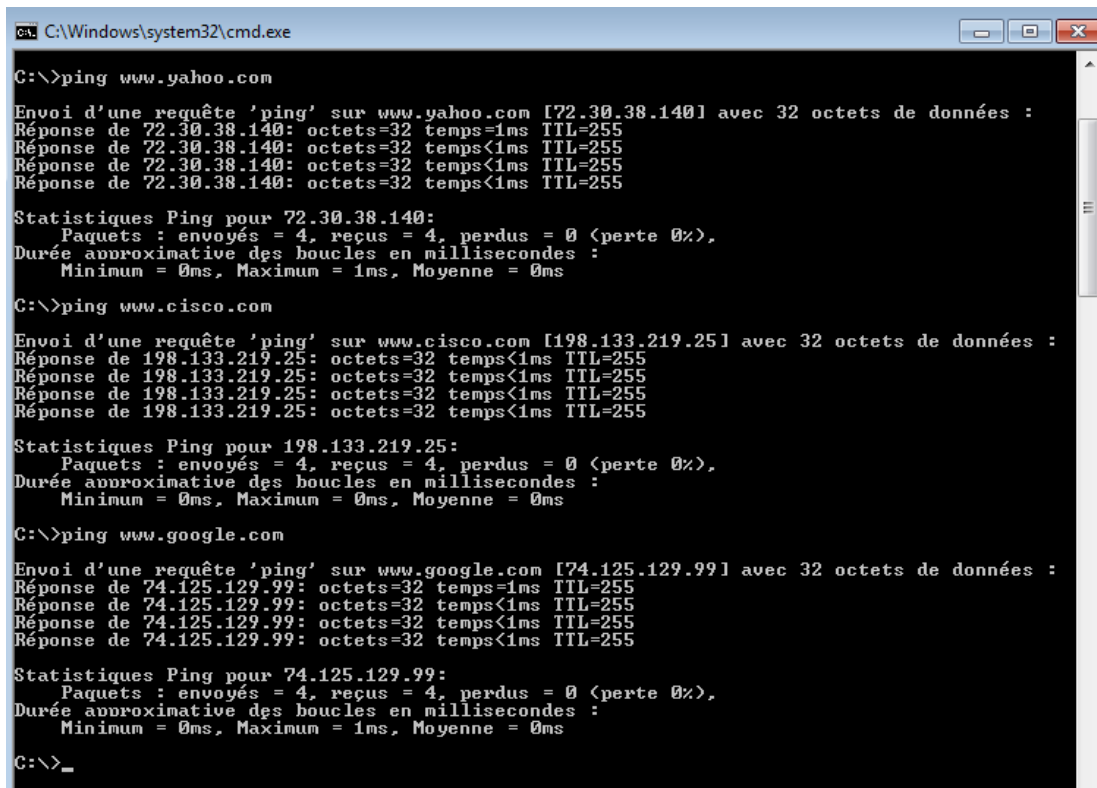
**Note:** In the preceding example of a captured ICMP request, ICMP data is encapsulated inside an IPv4 packet PDU (IPv4 header) which is then encapsulated in an Ethernet II frame PDU (Ethernet II header) for transmission on the LAN.

## Part 2: Ping a remote machine

In Part 2, you will ping remote hosts (hosts not on the LAN) and examine the generated data from those pings. You will then determine what is different about this data from the data examined in Part 1.

### Step 1: Start capturing data on the interface.

- a) With the capture active, ping the following three website URLs:
  - www.yahoo.com
  - www.cisco.com
  - www.google.com



```
C:\Windows\system32\cmd.exe

C:\>ping www.yahoo.com

Envoi d'une requête 'ping' sur www.yahoo.com [72.30.38.140] avec 32 octets de données :
Réponse de 72.30.38.140: octets=32 temps=1ms TTL=255
Réponse de 72.30.38.140: octets=32 temps<1ms TTL=255
Réponse de 72.30.38.140: octets=32 temps<1ms TTL=255
Réponse de 72.30.38.140: octets=32 temps<1ms TTL=255

Statistiques Ping pour 72.30.38.140:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 1ms, Moyenne = 0ms

C:\>ping www.cisco.com

Envoi d'une requête 'ping' sur www.cisco.com [198.133.219.25] avec 32 octets de données :
Réponse de 198.133.219.25: octets=32 temps<1ms TTL=255
Réponse de 198.133.219.25: octets=32 temps<1ms TTL=255
Réponse de 198.133.219.25: octets=32 temps<1ms TTL=255
Réponse de 198.133.219.25: octets=32 temps<1ms TTL=255

Statistiques Ping pour 198.133.219.25:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 0ms, Moyenne = 0ms

C:\>ping www.google.com

Envoi d'une requête 'ping' sur www.google.com [74.125.129.99] avec 32 octets de données :
Réponse de 74.125.129.99: octets=32 temps=1ms TTL=255
Réponse de 74.125.129.99: octets=32 temps<1ms TTL=255
Réponse de 74.125.129.99: octets=32 temps<1ms TTL=255
Réponse de 74.125.129.99: octets=32 temps<1ms TTL=255

Statistiques Ping pour 74.125.129.99:
    Paquets : envoyés = 4, reçus = 4, perdus = 0 (perte 0%),
    Durée approximative des boucles en millisecondes :
        Minimum = 0ms, Maximum = 1ms, Moyenne = 0ms

C:\>_
```

Note the IP address of each URL.  
How did your computer find these IP addresses?

- b) You can stop capturing data by clicking the Stop Capture icon

## Step 2: Examining and analyzing the data from the remote hosts.

- a) Review the captured data in Wireshark, examine the IP and MAC addresses of the three locations that you pinged. List the destination IP and MAC addresses for all three locations.
- b) How does this information differ from the local ping information you received in Part 1? What is the main difference between a local and a remote communication?