

Bài 3: PHÂN TÍCH CÁC PROTOCOL THÔNG DỤNG CỦA TCP/IP

- **Mục tiêu thí nghiệm:**

- Giúp sinh viên làm quen với các giao thức thông dụng của TCP/IP:
 - ARP.
 - ICMP.
 - TELNET.
 - Phân tích quá trình thiết lập và kết thúc một kết nối TCP.
- Thực hành phân tích protocol bằng chương trình Wireshark.

- **Nội dung thí nghiệm:**

- Phân tích các giao thức ARP, DHCP, ICMP, TELNET.
- Tìm hiểu về quá trình thiết lập và giải tỏa một kết nối TCP.

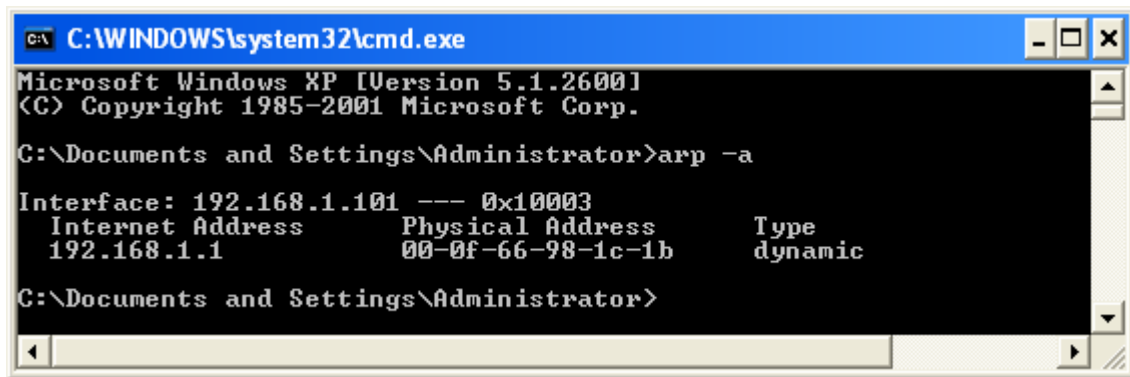
- **Thiết bị thí nghiệm:**

- 2 máy tính có card mạng có cài hệ điều hành WINXP, chương trình TFTP32, chương trình Wireshark.
- 1 đoạn dây cáp mạng (cáp chéo).

Phần 1: Cơ sở lý thuyết

1. ARP (Address Resolution Protocol)

Để các máy có thể trao đổi dữ liệu được với nhau thì phía gửi phải biết được thông tin về địa chỉ IP và địa chỉ MAC của máy nhận. Trong khi địa chỉ IP có thể có được thông qua một số phương pháp như DNS hay hệ thống tên thiết bị (devices name) thì địa chỉ MAC gần như là chưa được biết trước. TCP/IP định nghĩa một giao thức để thực hiện việc tìm địa chỉ MAC với địa chỉ IP đã biết, đó là ARP. Nói cách khác ARP cho phép ánh xạ một địa chỉ IP với một địa chỉ MAC tương ứng, thông tin này sau đó được lưu vào trong một cơ sở dữ liệu là bảng ARP (lưu trong RAM) để dùng sau này. Ta có thể xem bảng ARP ở hệ điều hành Windows bằng lệnh **arp -a** ở dấu nhắc DOS:



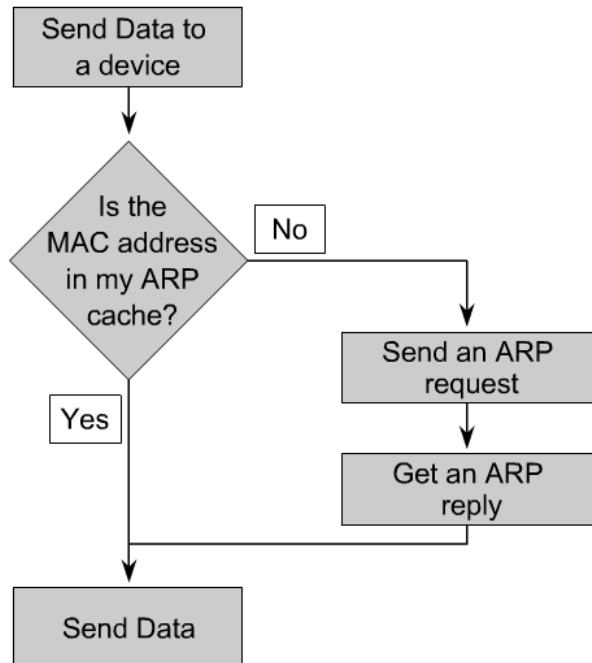
```
C:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>arp -a

Interface: 192.168.1.101 --- 0x10003
Internet Address      Physical Address      Type
192.168.1.1           00-0f-66-98-1c-1b    dynamic

C:\Documents and Settings\Administrator>
```

Hoạt động của ARP có thể được tóm tắt như sau: khi một máy A muốn gửi dữ liệu đến máy B (đã biết địa chỉ IP), nó sẽ tra địa chỉ IP này trong bảng ARP để tìm địa chỉ MAC. Nếu trong bảng ARP chưa có địa chỉ này thì máy gửi sẽ thực hiện gửi một gói ARP request với địa chỉ IP nguồn và đích tương ứng là của máy A và B, địa chỉ MAC nguồn là của máy A, địa chỉ MAC đích là địa chỉ quảng bá (ff-ff-ff-ff-ff-ff). Do địa chỉ MAC đích là địa chỉ quảng bá nên tất cả thiết bị mạng trên phần mạng đó sẽ nhận gói ARP request này, các máy mở gói và kiểm tra địa chỉ IP đích, máy B kiểm tra thấy địa chỉ IP đích chính là địa chỉ IP của nó, gói ARP request yêu cầu địa chỉ MAC của máy B, do đó máy B sẽ trả lời yêu cầu này bằng một gói ARP reply; tất cả các máy khác có địa chỉ IP không giống với địa chỉ IP đích trong gói ARP request sẽ hủy gói. Trước khi gửi gói ARP reply, máy nhận sẽ trích địa chỉ IP và MAC nguồn trong gói ARP request và lưu vào bảng ARP. Gói ARP reply có địa chỉ IP và MAC nguồn và đích tương ứng của máy B và A, trong phần dữ liệu của gói ARP reply bao gồm địa chỉ IP và MAC của cả máy A và B.



Trên đây chỉ trình bày quá trình ARP giữa hai máy trong cùng một phần mạng, sinh viên tự tìm hiểu quá trình ARP giữa các máy nằm ở các mạng khác nhau, proxy ARP, gratuitous ARP và trình bày trong phần chuẩn bị.

2. ICMP (Internet Control Message Protocol)

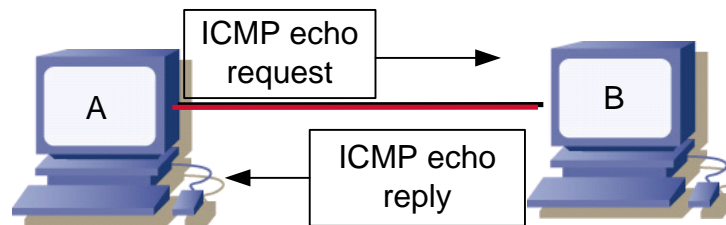
Trong mô hình TCP/IP thì IP cung cấp phương pháp truyền không đáng tin cậy, không kết nối (connectionless), nó được thiết kế để tận dụng tối đa tài nguyên mạng. Tuy nhiên, IP không có cơ chế báo lỗi hay sửa lỗi, như vậy, chuyện gì sẽ xảy ra nếu như có sự cố, chẳng hạn router loại bỏ gói khi nó không tìm thấy đường đến đích? ICMP được thiết kế để hoàn tất 2 nhiệm vụ: báo lỗi và query.

Để phục vụ nhiệm vụ này, ICMP có 2 loại gói: gói báo lỗi và gói query, mỗi loại có nhiều thông điệp mang ý nghĩa khác nhau:

Loại gói	Mã thông điệp	Thông điệp
Báo lỗi	3	Destination unreachable
	4	Source quence
	11	Time exceeded
	12	Parameter problem
	5	Redirection
Query	8 hay 0	Echo request or reply

	13 hay 14	Timestamp request or reply
	17 hay 18	Address mask request or reply
	10 hay 9	Router solicitation or advertisement

Trong phạm vi bài thí nghiệm này ta chỉ xét đến các thông điệp echo request và reply của ICMP mà thôi. Thông điệp echo request và reply được thiết kế cho mục đích phát hiện và chuẩn đoán lỗi. Hoạt động của cặp thông điệp này hết sức đơn giản: người dùng hoặc người quản trị gửi một thông điệp echo request từ một hệ thống, hệ thống nhận được thông điệp echo request sẽ gửi phúc đáp bằng một thông điệp echo reply cho hệ thống gửi. Cặp thông điệp này có thể cho biết hai hệ thống có thể liên lạc được với nhau ở lớp 3 hay không, đồng thời cũng cho biết các thiết bị trung gian (router, switch) đã nhận, xử lý và chuyển được thông điệp IP. Nếu vì một lý do nào đó mà máy đích không nhận được thông điệp echo request thì tại thiết bị cuối cùng nhận được thông điệp echo request sẽ phúc đáp bằng một thông điệp lỗi cho biết lỗi là gì.

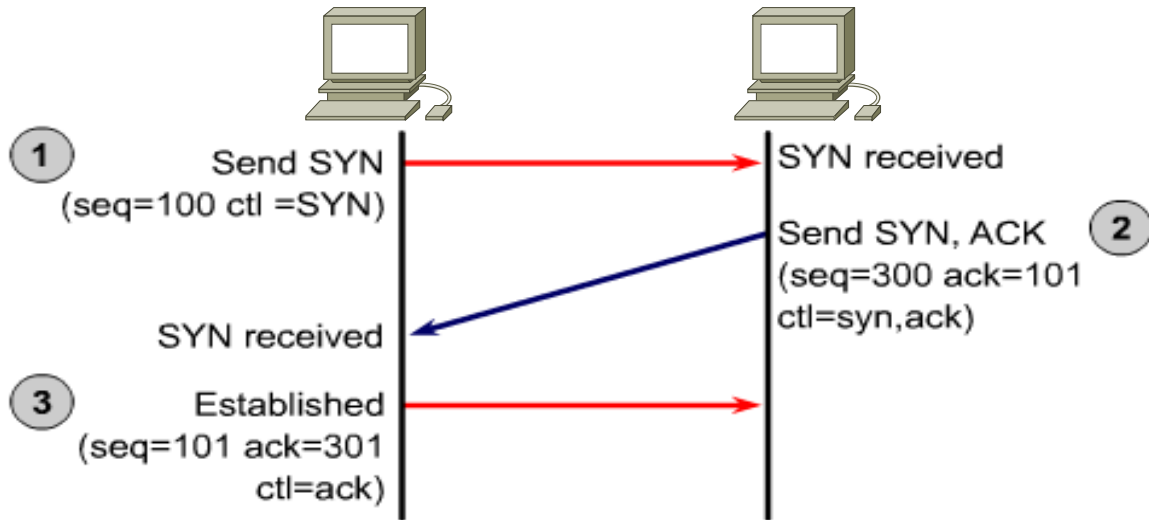


3. Quá trình thiết lập và giải tỏa một kết nối TCP

TCP là một giao thức ở lớp 4, có chức năng đảm bảo sự chuyển vận đáng tin cậy của dữ liệu qua môi trường mạng, ngoài ra, TCP còn được thiết kế với cả chức năng kiểm soát luồng và kiểm soát lỗi. Chi tiết về TCP đã được đề cập nhiều trong các giáo trình truyền số liệu và hệ thống viễn thông, ở đây chỉ tóm tắt quá trình thiết lập và giải tỏa một kết nối TCP.

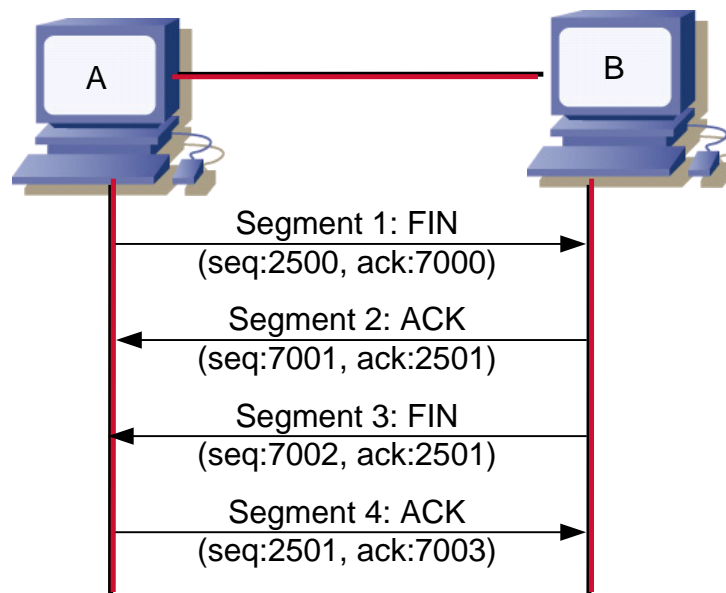
_ **Quá trình thiết lập một kết nối TCP:** còn được gọi là quá trình bắt tay ba chiều (three-way-handshake), được tiến hành trước khi dữ liệu có thể được chuyển giữa các thiết bị nhằm đồng bộ các thông số của kết nối. Quá trình này bao gồm ba bước như sau:

- **Bước 1:** client khởi tạo kết nối với server bằng cách gửi một gói TCP với cờ SYN được bật, thông báo cho server biết số thứ tự x của gói nhằm đồng bộ về thông số với server.
- **Bước 2:** server nhận được gói này lưu lại số thứ tự x , và trả lời bằng một gói có thứ tự $x+1$, trong đó chứa số thứ tự y của nó với cờ SYN và ACK được bật. Việc trả lời bằng gói có số thứ tự là $x+1$ nhằm mục đích thông báo cho client biết được máy nhận đã nhận được tất cả dữ liệu cho đến số thứ tự là x và mong chờ gói có số thứ tự là $x+1$.
- **Bước 3:** sau khi nhận được gói này, client phúc đáp bằng một gói TCP có cờ ACK được bật và có số thứ tự là $y+1$. Sau bước này thì dữ liệu có thể được chuyển giữa client và server



_ **Quá trình giải tỏa một kết nối TCP:** Quá trình giải tỏa một kết nối TCP bao gồm bốn bước (four-way handshake) được tóm tắt như sau:

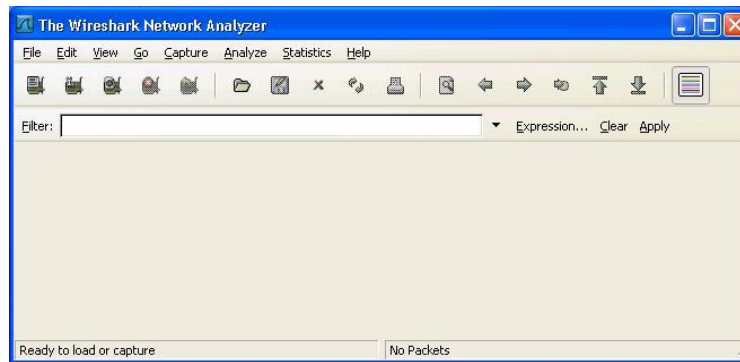
- **Bước 1:** client khi muốn kết thúc kết nối sẽ gửi một gói TCP với cờ FIN được bật nhằm thông báo cho server việc giải tỏa kết nối.
- **Bước 2:** server trả lời client bằng một gói TCP có cờ ACK được bật nhằm xác nhận đã nhận được gói trước đó của client.
- **Bước 3:** server gửi tiếp một gói có cờ FIN được bật nhằm thông báo cho client biết việc giải tỏa kết nối.
- **Bước 4:** client trả lời server bằng một gói có cờ ACK được bật để xác nhận đã nhận được gói FIN của server, sau gói này, cả client và server đều giải tỏa kết nối.



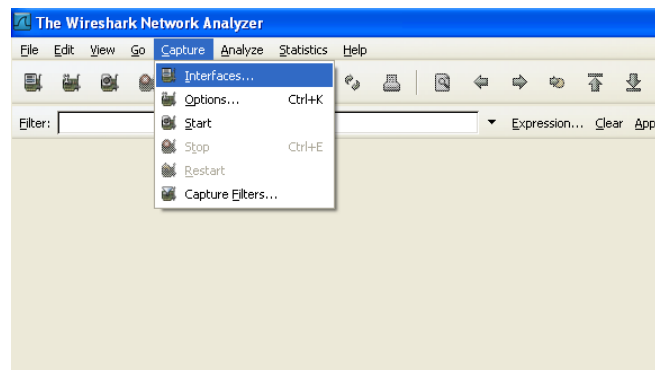
4. Dùng chương trình Wireshark để phân tích giao thức mạng

Wireshark là một chương trình giúp phân tích giao thức mạng, được cung cấp miễn phí tại địa chỉ <http://www.wireshark.org/>

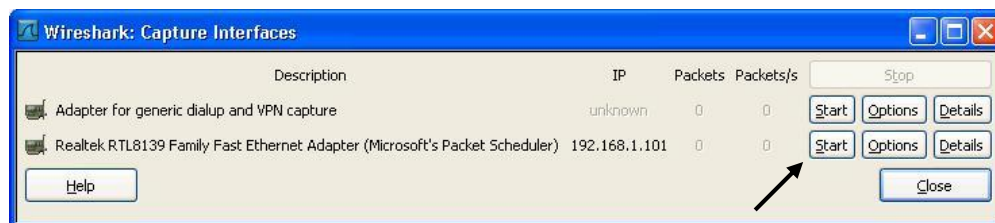
Sau khi cài đặt, chạy chương trình Wireshark, giao diện chương trình như sau:



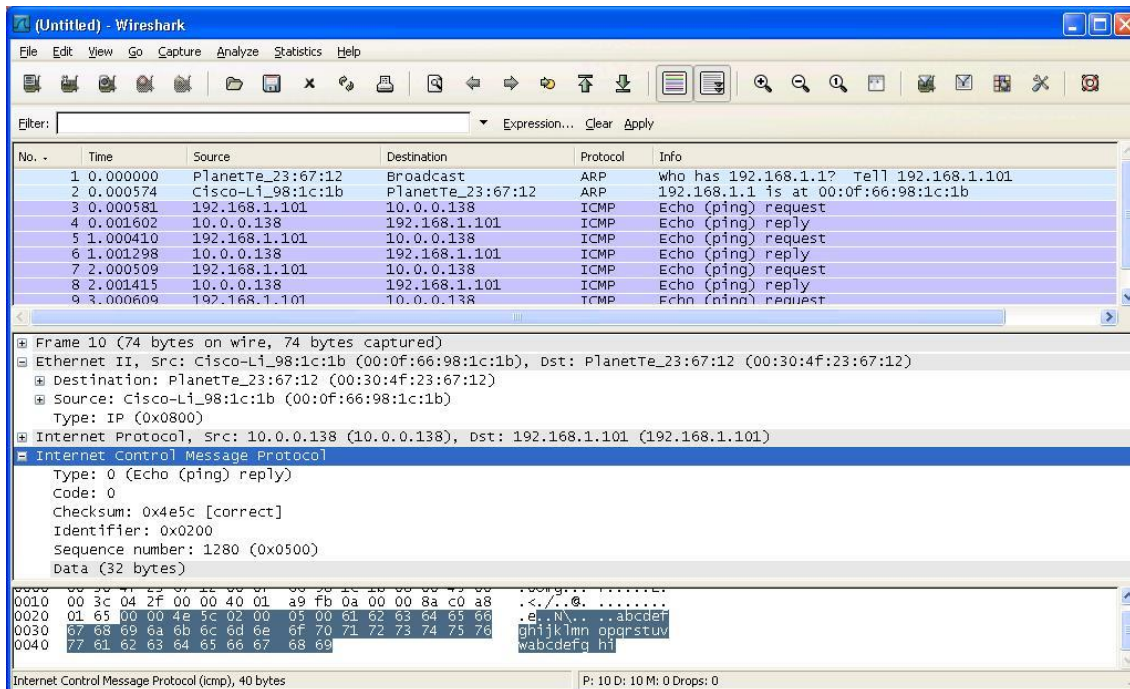
Để bắt đầu bắt gói để phân tích, từ menu **Capture**, chọn **Interfaces**



Cửa sổ mới hiện ra cho phép chọn cổng để bắt đầu bắt gói, ta chọn card mạng đang chạy của máy rồi bấm nút **start** để bắt đầu bắt gói:



Sau khi đã bắt gói xong, ta dừng quá trình bắt gói bằng cách từ menu **Capture** chọn **Stop**, giao diện chương trình sau khi bắt gói như sau:



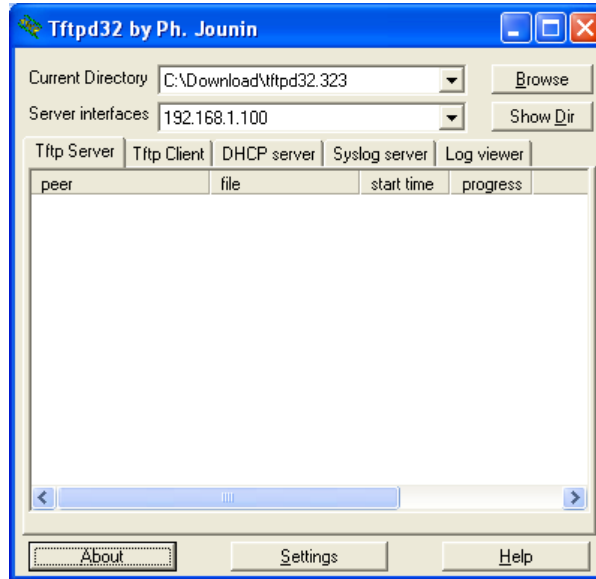
Giao diện chương trình gồm có 3 phần:

- Phần trên cùng cho người dùng thấy thông tin tóm tắt của các gói đã bắt được theo thứ tự thời gian.
- Khi ta chọn vào một gói ở phần trên, phần giữa giúp người dùng phân tích toàn bộ thông tin chi tiết của gói hiện tại, bao gồm tất cả thông tin đóng gói ở các lớp và thông tin về các trường trong header ở mỗi lớp.
- Phần thứ ba cho biết giá trị của các trường của gói hiện tại dưới dạng số hex và mã ASCII.

5. Sử dụng chương trình TFTP32 làm DHCP server

Chương trình TFTP32 là một phần mềm cho phép dựng TFTP server, TFTP client, DHCP server và Syslog server. TFTP32 được cung cấp miễn phí tại địa chỉ <http://tftpd32.jounin.net/>

Trong bài này ta chỉ sử dụng TFTP32 để dựng DHCP server. Sau khi cài đặt, khởi động chương trình, giao diện chính của chương trình như sau:



Trong mục Settings, ta chọn tab DHCP server. Ý nghĩa của các trường chủ yếu như sau:

- **IP pool starting address:** địa chỉ IP bắt đầu để cấp phát cho các client trong mạng.
- **Size of pool:** số lượng địa chỉ IP cung cấp cho các máy.
- **WINS/DNS server:** địa chỉ của WINS hay DNS server cung cấp cho client.
- **Default router:** địa chỉ của default gateway.
- **Mask:** subnetmask cung cấp cho client.

Sau khi điền các thông số, ta bấm nút **save** để kích hoạt cho DHCP server làm việc.

Phần 2: Câu hỏi chuẩn bị

Câu 1: Hãy trình bày quá trình đóng gói (encapsulation) và gỡ gói (de-encapsulation) của dữ liệu khi gửi qua mạng.

Câu 2: Hãy so sánh các phương thức truyền unicast, broadcast và multicast.

Câu 3: Trình bày vắn tắt quá trình ARP giữa các máy nằm ở các mạng khác nhau, proxy ARP, gratuitous ARP.

Câu 4: Hãy trình bày các trường trong khung Ethernet, gói IP và TCP.

Câu 5: Hãy so sánh giữa TCP và UDP.

Phần 3: Thí nghiệm

SV thực hiện thí nghiệm và trả lời các câu hỏi trong phần thí nghiệm, sau khi hoàn thành xong phần thí nghiệm, sinh viên nộp lại câu trả lời cho giáo viên hướng dẫn thí nghiệm.

Ngày thí nghiệm:.....

Nhóm:

1/.....

2/.....

3/.....

4/.....

1. Dùng Wireshark để phân tích quá trình ARP và ICMP

Mô hình kết nối: kết nối hai máy, gán IP cho hai máy như mô hình sau:



Chạy chương trình Wireshark, bắt đầu cho bắt gói trên cả hai máy.

Từ dấu nhắc DOS xóa bảng ARP của cả hai máy bằng lệnh **arp -d**, kiểm tra lại rằng bảng ARP của hai máy là trống bằng lệnh **arp -a**.

Thực hiện ping từ máy A đến máy B bằng cách từ dấu nhắc DOS của máy A gõ lệnh **ping 192.168.1.2**. Quá trình ping có thành công không?..... (Nếu quá trình ping không thành công, sinh viên liên hệ với giáo viên đứng lớp nhờ giúp đỡ).

Sau khi thực hiện xong lện ping, dừng quá trình bắt gói trên cả hai máy.

Xem bảng ARP trên cả hai máy bằng lệnh **arp -a** tại dấu nhắc DOS. Ghi lại bảng ARP của hai máy:

.....

Xem địa chỉ MAC và địa chỉ IP của hai máy bằng lệnh **ipconfig /all** tại dấu nhắc DOS. Nhận xét về sự tương quan giữa bảng ARP và địa chỉ của các máy.

.....

Phân tích gói ARP request và ARP reply, điền vào bảng sau:

Gói ARP request:

Layer 2 Dest address _____ Layer 2 Src Address _____
 Layer 2 code for encapsulated data _____
 Hardware Type _____ Layer 3 Protocol Type _____
 Hardware Addr Length _____ Layer 3 Addr Length _____
 Arp Operation Code and Name _____
 Sender Hardware address _____
 Sender IP address _____.
 Target Hardware Address _____
 Target IP Address _____.

Gói ARP reply:

Layer 2 Dest address _____ Layer 2 Src Address _____
 Layer 2 code for encapsulated data _____
 Hardware Type _____ Layer 3 Protocol Type _____
 Hardware Addr Length _____ Layer 3 Addr Length _____
 Arp Operation Code and Name _____
 Sender Hardware address _____
 Sender IP address _____.
 Target Hardware Address _____
 Target IP Address _____.

Phân tích quá trình gửi và nhận gói giữa hai máy thông qua các gói bắt được.

.....

Phân tích các trường lớp 2 và lớp 3 của gói ICMP echo request và ICMP echo reply. Dữ liệu trong gói ICMP echo request và reply là gì? Có giống nhau hay không? Mục đích của dữ liệu này là gì?

.....

2. Phân tích quá trình thiết lập và kết thúc một kết nối TCP

Mô hình kết nối: thực hiện mô hình kết nối sau



Trên máy A, kích hoạt chức năng Telnet: chọn **Start>Run**, trong cửa sổ mới gõ vào lệnh **services.msc** rồi nhấn **Ok**. Trong cửa sổ mới hiện ra, click phải vào dòng “**Telnet**”, chọn **Properties**, ở tab **General**, chọn **Startup type** là **Manual**, rồi bấm vào nút **Start**. Chờ cho quá trình kích hoạt telnet thành công.

Chạy chương trình Wireshark, bắt đầu cho bắt gói trên cả hai máy.

Từ máy B, thực hiện telnet tới máy A bằng cách ở dấu nhắc DOS, dùng lệnh **telnet 192.168.1.1**.

Sau khi telnet thành công, gõ một lệnh DOS bất kỳ ở dấu nhắc trong cửa sổ telnet (sinh viên có thể dùng lệnh **help**). Sau đó, thoát khỏi kết nối telnet bằng lệnh **exit**. Dừng quá trình bắt gói.

Chọn vào một gói của kết nối telnet, chọn menu **Statistics>Flow graph**, trong gửả sổ mới hiện ra, sửa phần **Choose flow type** thành **TCP type**. Trả lời các câu hỏi sau: (sinh viên có thể dùng các thông tin chi tiết về các trường của các gói trong giao diện chính của chương trình để trả lời)

Dựa vào các gói Wireshark bắt được, phân tích quá trình thiết lập kết nối của một kết nối TCP (ở đây là telnet):

.....

.....

.....

.....

.....

Dựa vào các gói Wireshark bắt được, phân tích quá trình gửi dữ liệu của một kết nối TCP (ở đây là telnet):

.....

.....

.....

.....

.....

Dựa vào các gói Wireshark bắt được, phân tích quá trình giải tỏa kết nối của một kết nối TCP (ở đây là telnet):

[illegible]

Chọn vào một gói của kết nối telnet, chọn menu **Analyze>Follow TCP stream**, Follow TCP stream là chức năng của Wireshark, dừng lại thông tin trao đổi của kết nối TCP dựa vào dữ liệu nhận được trong các gói.

Hãy nhận xét về thông tin nhận được từ việc dựng lại kết nối telnet vừa thực hiện với thông tin nhận được từ kết nối thật.

[illegible]

Rút ra kết luận về hoạt động chuyển dữ liệu của telnet, tại sao telnet được gọi là một “terminal emulator”?

[illegible]