HDU-ITMO Joint Institute
杭州电子科技大学圣光机联合学院

# COMPUTER NETWORKS

## LAB 2

## A SIMPLE NETWORK ROUTING

## Instructors


Elena Boldyreva, Associate Professor eaboldyreva@itmo.ru

Yuriy Boldyrev, Assistant Professor

PREREQUISITES

1. Computer Networking: A Top-Down Approach, 7th ed., J.F. Kurose and K.W. Ross (Chapter 1 and Chapter 2)

REPORT

After completion the task the students need to submit:

- **The format of the report file is *pdf*.**
- Upload reports for the GitLab repository (to your personal project)
- Do not forget to write your name and ID inside the report

**GOAL:** to study the simulation mode of Cisco Packet Tracer, ARP and ICMP protocols on the example of $ping$ and $tracert$ commands.

**PLAN:**

1. Building a network topology, configuring end nodes;
2. To configure the router;
3. Checking network operation in simulation mode;
4. Sending ping inside the network;
5. sending a ping request to an external network;
6. Sending ping request to a nonexistent IP address of the host;
7. Completing an individual task.

## THEORETICAL PART:

### The ARP Protocol

The address Resolution Protocol (ARP) is used to determine a physical address from an IP address. The ARP Protocol works in different ways depending on which link layer Protocol is running on the given network with the ability to broadcast access to all network nodes simultaneously.

The ARP Protocol allows you to dynamically determine the MAC address by IP address. A MAC address is a unique serial number assigned to each network device to identify it in the network, also called a physical or hardware address. The LAN Protocol supported in the lab is Ethernet. In Ethernet networks that use the TCP/IP stack, the network interface has a 48-bit physical address. Frames exchanged at the link layer must contain the hardware address of the network interface. However, TCP / IP uses its own addressing scheme: 32-bit IP addresses. The value of the receiver's IP address is not sufficient to send a datagram to this host. The Ethernet driver must know the MAC address of the destination interface in order to send data there. The task of ARP is to provide dynamic correspondence between 32-bit IP addresses and 48-bit MAC addresses used by various network technologies. The ARP Protocol works within a single subnet and automatically starts when it is necessary to convert an IP address to a hardware address.

A node that needs to map an IP address to a local address generates an ARP request, inserts it into a link-layer Protocol frame, specifying a known IP address in it, and sends the request broadcast. All local network nodes receive an ARP request and compare the IP address specified there with their own. If they match, the node generates an ARP response, in which it specifies its IP address and its local address, and sends it already directionally, since the sender specifies its local address in the ARP request.

In order to reduce the number of ARP requests sent, each device on the network that uses the ARP Protocol must have a special buffer memory. It stores address pairs (IP address, physical address) of devices on the network. Whenever the device receives an ARP response, it stores the corresponding pair in buffer memory. If the address is in the list of pairs, there is no need to send an ARP request. This buffer memory is called an ARP table.

An ARP table can contain both static and dynamic entries. Dynamic entries are added and deleted automatically, while static entries are entered manually.

Since most devices on the network support dynamic address resolution, the administrator usually does not need to manually specify ARP Protocol entries in the address table.

Each entry in the ARP table has its own lifetime. ARP table cleanup policies are dictated by the operating system used. When you add an entry, a timer is activated for it.

ARP messages are encapsulated in the frame data field when transmitted over the network. They do not contain an IP header. Unlike most protocols, ARP messages do not have a fixed header format. This is because the Protocol was designed so that it is applicable to address resolution in various networks.

ARP requests and responses use the same packet format. Since local addresses can have different lengths in different types of networks, the format of the ARP Protocol packet depends on the type of network. Figure shows the structure of the request and response package.

<table>
<tr><td colspan="2">Network Type</td><td>Protocol</td></tr>
<tr><td>HAL</td><td>PAL</td><td>Operation</td></tr>
<tr><td colspan="3">Source Hardware Address</td></tr>
<tr><td colspan="2">Source Hardware Address</td><td>Source IP</td></tr>
<tr><td colspan="2">Source IP</td><td>Destination Hardware Address</td></tr>
<tr><td colspan="3">Destination Hardware Address</td></tr>
<tr><td colspan="3">Destination IP</td></tr>
</table>

* Network Type – type of channel Protocol

For Ethernet-1.

• Protocol - the network layer Protocol

• HAL - length of the channel address

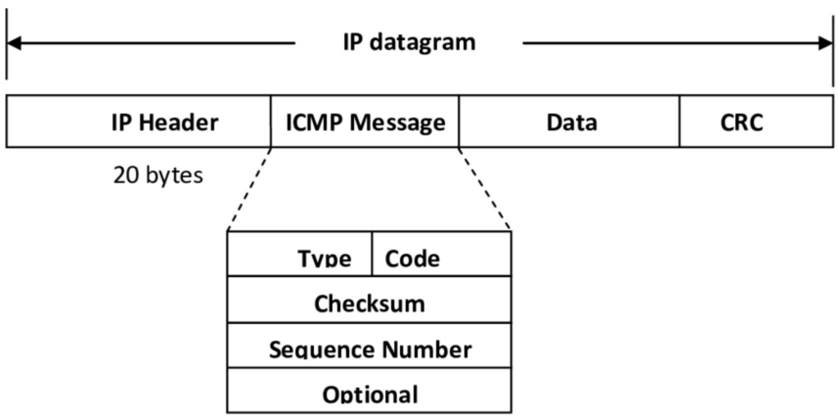
* PAL-length of the network address

• Operation - type of operation (1-request, 2-response)

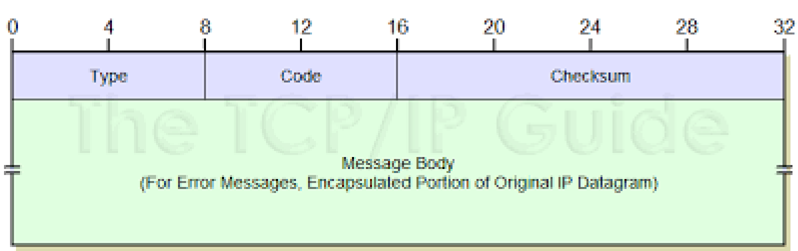
The node that sends the ARP request fills in all fields in the packet, except for the local address field that is being searched for. The value of this field is filled in by the host that recognized its IP address.

**The ICMP Protocol**

The ICMP Protocol is used for transmitting control and diagnostic messages. It is used to send messages about errors, as well as situations that require special attention. The Protocol belongs to the network layer of the TCP/IP model. ICMP messages are generated and processed by network (IP) and higher-level protocols (TCP or UDP). When some ICMP messages appear, error messages are generated and passed to user processes. ICMP messages are transmitted inside IP datagrams:




The format of the ICMP message is shown in figure. The ICMP header includes 8 bytes, but only the first 4 bytes are the same for all messages, and the rest of the message header and body fields are determined by the message type.

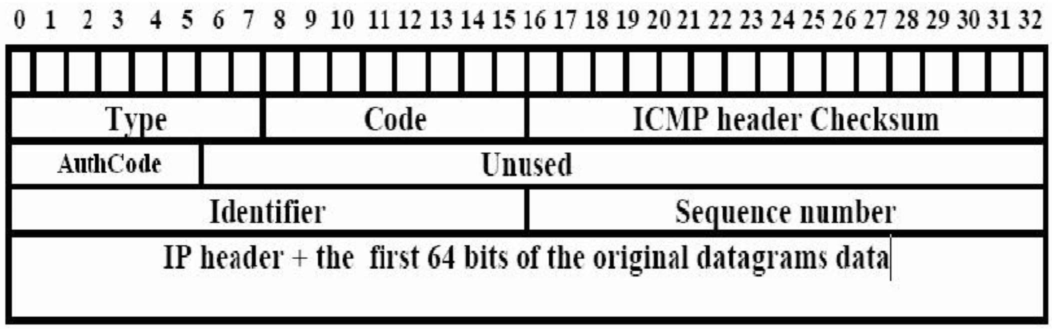



The checksum field covers the entire ICMP message.

The message type is determined by the value of the “Type” field in the header. Some types of ICMP messages have internal granularity (code), and the specific type of message is determined by both the message type and the message code. For more information about types and codes of ICMP messages, see the ICMP Protocol specification RFC 792. [Electronic resource]. URL: http://tools.ietf.org/html/rfc792.

**Ping program**

The *ping* program was designed to check the availability of a remote host. The program sends an ICMP echo request to the node and waits for the ICMP echo response to return. The hing program is usually the first diagnostic tool that begins identifying a problem in networks. In addition to availability, you can use ping to estimate how long it takes for a packet to return from a node, which gives you an idea of "how far away" the node is. In addition, Ping has options for recording the route and timestamp. The echo request and echo response messages have the same format:




- • Type – type of package
- 8-echo request
- 0 – response to the echo request
- * Code-decryption of the packet destination inside the type (in this case 0)
- * The checksum is calculated for the entire package
- * ID (identifier) – number of the message stream
- * Serial number – the number of the packet in the stream

Just as with other ICMP requests, the echo response must contain the ID and sequence number fields. In addition, any additional data sent by the computer must be echoed.

The ID field of the ICMP message sets the ID of the process sending the request. This allows the ping program to identify the returned response if multiple ping programs are running on the same host at the same time.

The sequence number starts at 0 and is incremented each time the next echo request is sent.

How the ping program works you studied in the lab 1

**Tracert program**

The tracert program allows you to view the route that IP datagrams travel from one host to another.

The tracert program does not require any special server applications. It uses standard functions of the ICMP and IP protocols. To understand how the program works, remember how to process the TTL field in the IP datagram header.

Each router that processes a datagram reduces the value of the TTL field in its header by one. When a datagram with a TTL of 1 is received, the router destroys it and sends an ICMP "time expired"message to the host that sent it. However, the datagram containing this ICMP message has the router's IP address as the source address.

This is what is used in the tracert program. An IP datagram is sent to the destination host with the TTL field set to one. The first router on the path of the datagram, destroys it (since TTL is 1) and sends an ICMP message about the expiration of time. This determines the first router in the route. Tracert then sends a datagram with a TTL field of 2, which allows you to get the IP address of the second router. Similar actions continue until the datagram reaches the destination host. When a response is received from this node, the tracing process is considered complete.

The first line, without a number, contains the name and IP address of the destination and indicates that the TTL value cannot be greater than 30.

The following output lines start with a printout of the TTL value (1, 2, 3, etc.) and contain the name (IP address) of the host or router and the time when the ICMP message was returned.

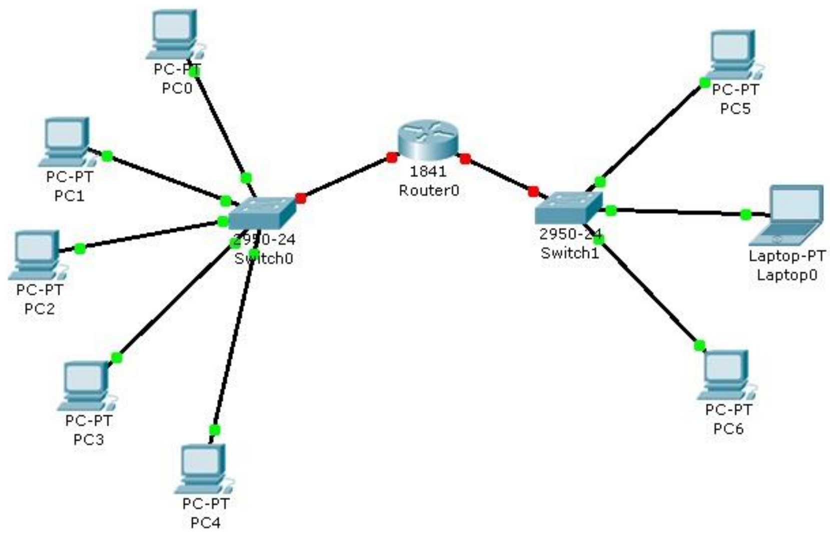
For each TTL value, 3 datagrams are sent. For each returned ICMP message, the return time is calculated and printed.

If a response to a datagram is not received within five seconds, an asterisk is printed and the next datagram is sent.

**PRACTICAL PART**

### 1. Creating a network topology

At the end of the first part, we created the following network topology, consisting of end nodes (PCs), switches, and a router:




*The workspace view*

Router 0 Router has two interfaces and connects the two subnets. Let's configure the end nodes.

### 2. Configuring the end nodes

On PC0-PC4 devices, set the specified IP addresses and subnet mask (table 1). the gateway IP address for all nodes is 192.168.3.1. the DNS server IP address is optional, because it will not be used in this work.

Table 1

| Host | IP-address | Subnet mask |
|---|---|---|
| PC0 | 192.168.3.3 | 255.255.255.0 |
| PC1 | 192.168.3.4 | 255.255.255.0 |
| PC2 | 192.168.3.5 | 255.255.255.0 |
| PC3 | 192.168.3.6 | 255.255.255.0 |

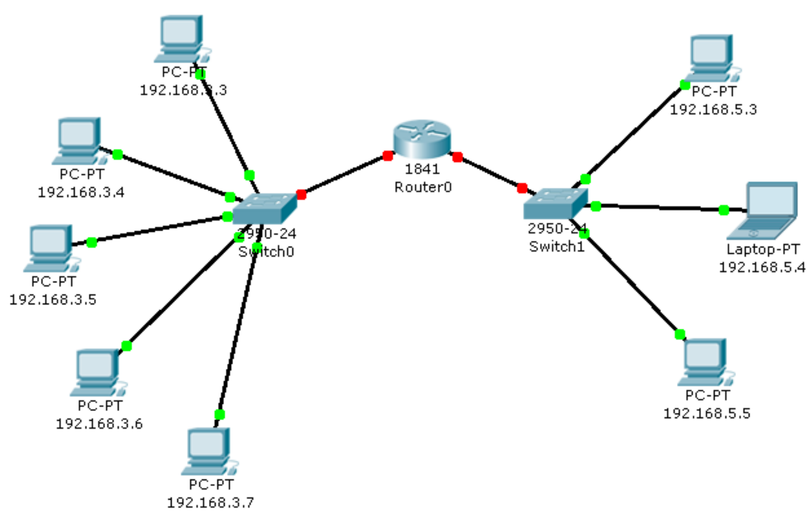
| | | |
|---|---|---|
| PC4 | 192.168.3.7 | 255.255.255.0 |

On PC 5, Laptop 0, and PC6 devices, set the specified IP addresses and subnet mask (table 2). the gateway IP address for all nodes is 192.168.5.1. the DNS server IP address is optional.

Table 2

| Host | IP-address | Subnet mask |
|---|---|---|
| PC5 | 192.168.5.3 | 255.255.255.0 |
| Laptop0 | 192.168.5.4 | 255.255.255.0 |
| PC6 | 192.168.5.5 | 255.255.255.0 |

Rename each node by its own IP address, and you will get the following:




*The workspace view*

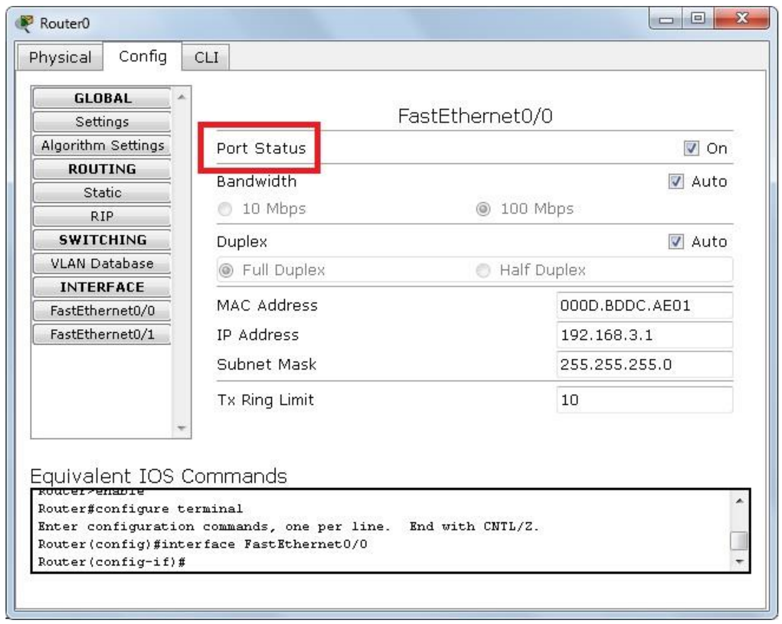
### 3. Configuring the router

When configuring end nodes, it was already mentioned that the router in this network topology has two interfaces. Configuring the fastethernet0/0 interface:

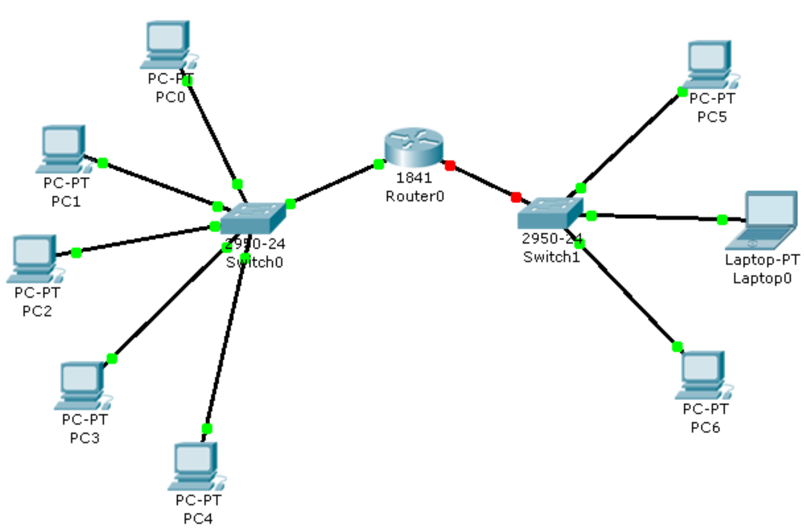
1) One click on the device (router);
2) Select the “Config " tab”;

3) Find the fastethernet0/0 interface, set the desired IP address and subnet mask.

**Important: the router interface is disabled by default; you must enable it by clicking the mouse next to”On".**

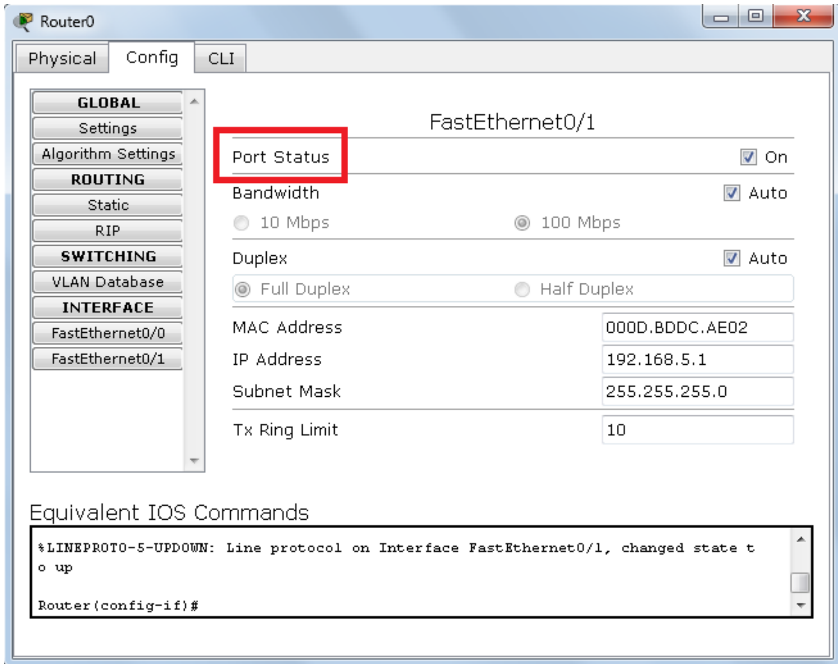



4) Close the window and look at the entire network topology. Green status indicators on the link between Router0 and Switch0 indicate that the interface is connected correctly



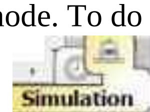

*The workspace view*

Similarly, we configure the FastEthernet0/1 interface.




You can add labels to the router interfaces using the Place Note tool in the Common Tools panel . You need to click on the tool, then click in the right place on the workspace.

## 4. Simulation mode Cisco Packet Tracer

Make sure that you are in simulation mode. To do this, click on the simulation icon in the lower-right corner of the simulator workspace. 


The event window opens, where you will see a list of events, control buttons, and the specified filters. By default, the packets of all possible protocols are filtered, i.e. they will be displayed. you need to correct and limit this list to the protocols under investigation.

Control buttons:

• Back – to-back

* Auto Capture / Play-automatically capture packets from source to receiver and back

* Capture/Forward-capture packets only from one device to another

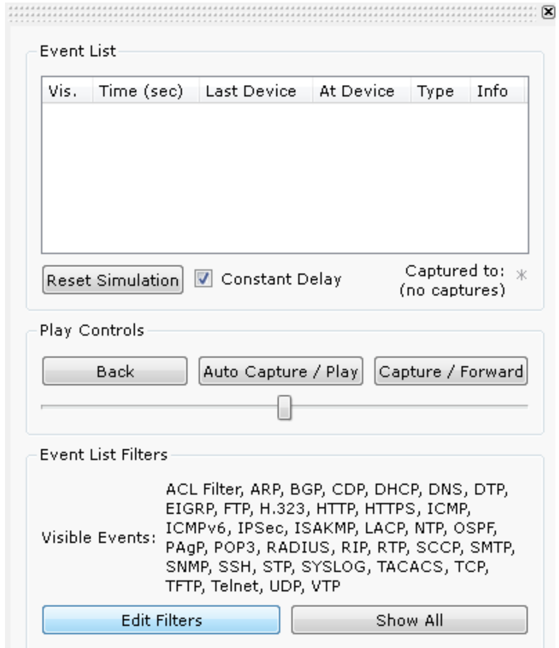
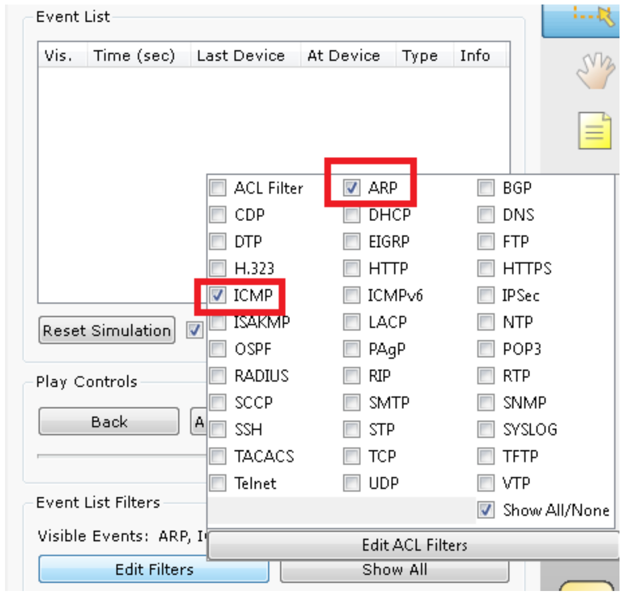



*Figure – Event List*

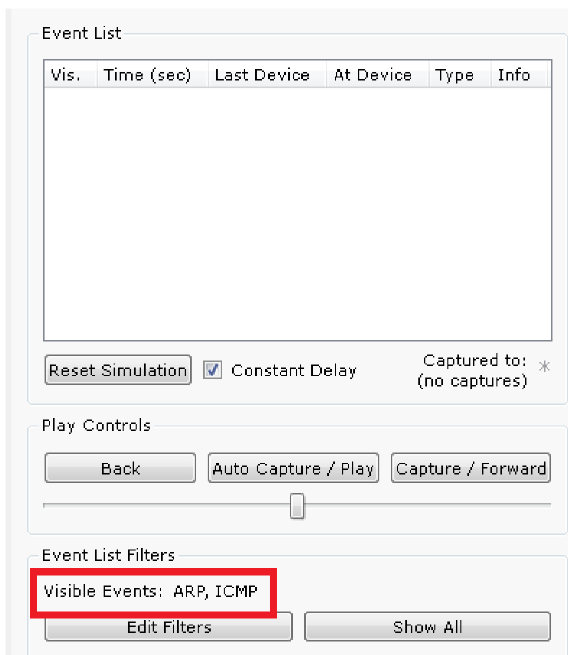
In this part, we are interested in two types of ARP and ICMP packets.

Therefore, you need to set a filter only for messages of the specified type:

1) Click on the "Edit Filters" button”
2) Remove the label from " Show All/None”
3) Select ARP and ICMP




4) Make sure that the specified filtering protocols are assigned

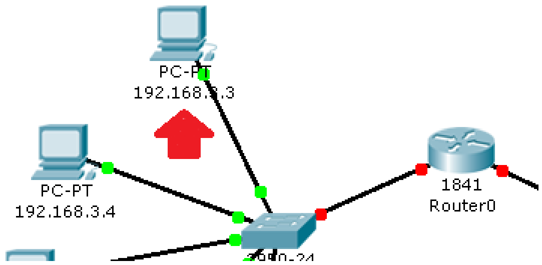



5. Verification of network operation in the simulation mode

Send a test ping request from the destination node with the IP address 192.168.3.3 to the host with the IP address 192.168.3.5.

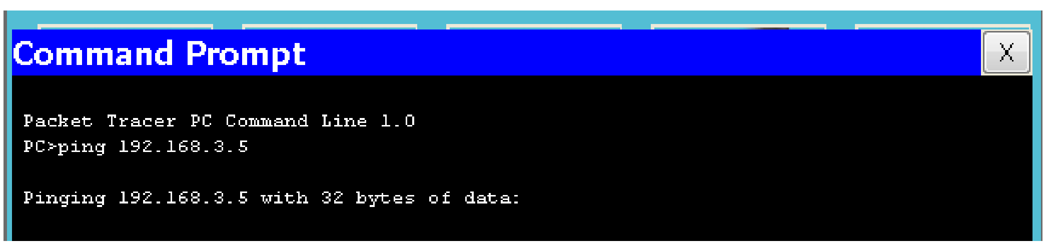
**Important: both nodes are located within the same network segment**

1) One click on the selected device



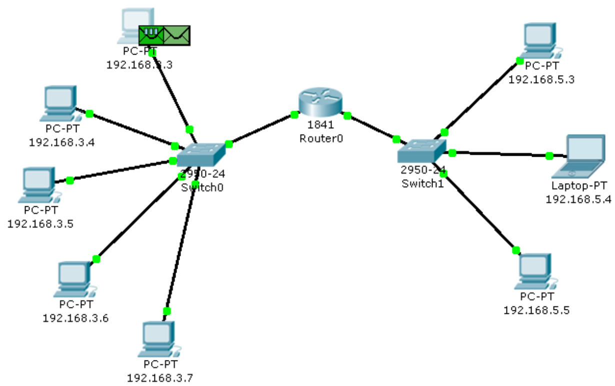

*Choise of PC-PT 192.168.3.3*

2) Select the Desktop tab, which contains simulators of some programs available on your computer
3) Select "Command Prompt", a program that simulates the computer's command line.
4) Use the ping utility to send a ping request. (Don't forget to press “Enter”).



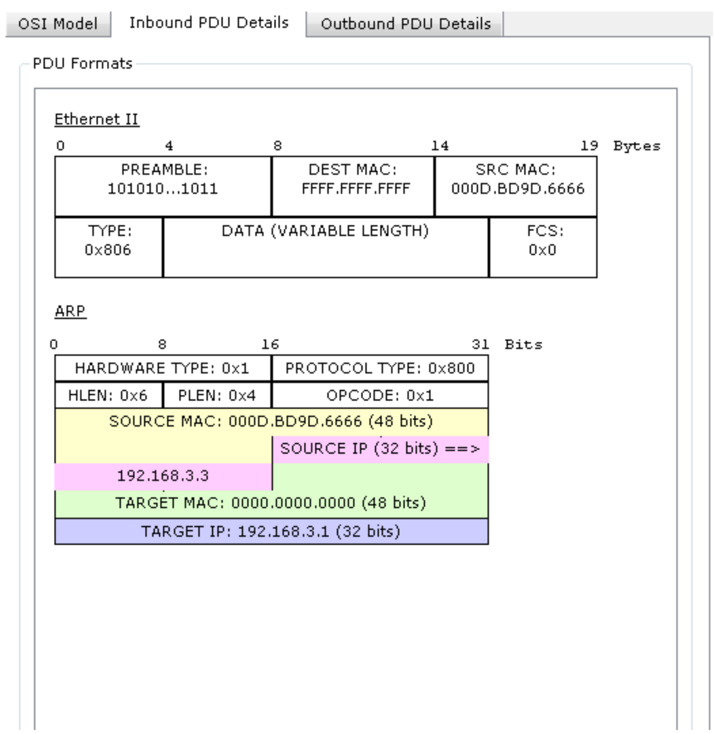

*Host 192.168.3.3command promt*

Two ARP and ICMP Protocol packets are generated on the source device. an ARP request is always generated when a host tries to communicate with another host.




*The workspace view*

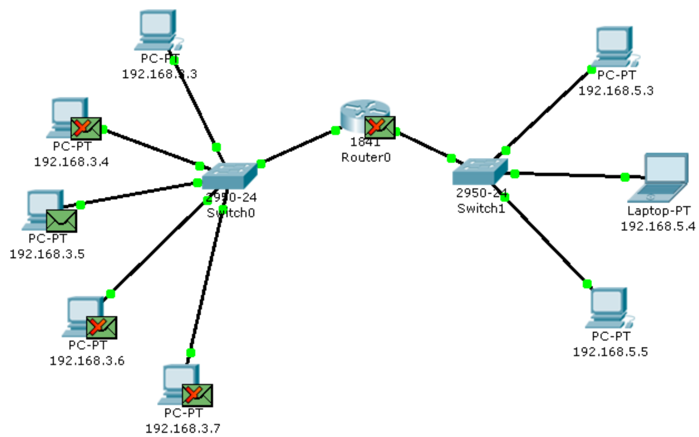
Click on the “Auto Capture/play” or “Capture/Forward " button, the latter will allow you to control the movement of packets from device to device yourself. We see that the ARP Protocol packet is sent first, since the ARP table of the host 192.168.3.3 is empty, and it still "does not know" who to send the ping request to. Make a single click on the package itself (the envelope) and see what levels of the OSI model are involved. Go to the “Inbound PDU Details " tab, which contains the packet structure.




*Format of the ARP request packet*

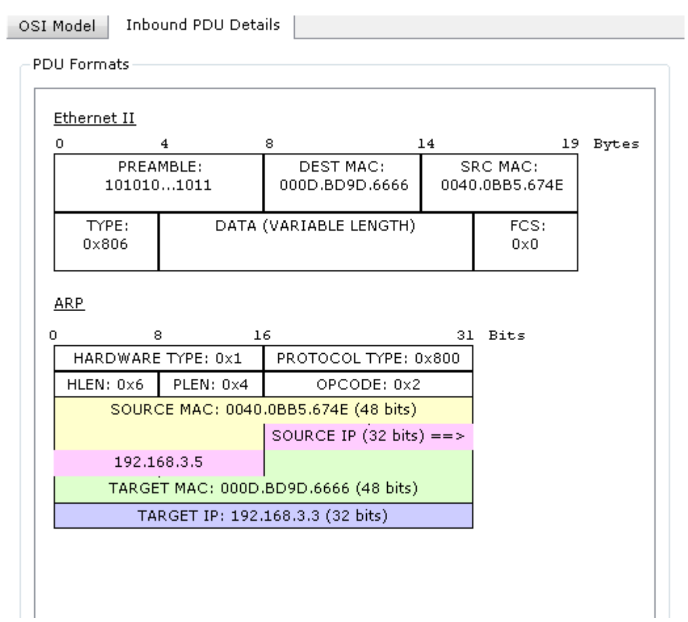
Node 192.168.3.3 has built a request and sends it as a broadcast message to all hosts on the subnet. In addition to the destination IP address, the request contains the sender's IP address and MAC address so that the receiving party can respond.

When viewing the packet flow, make sure that only host 192.168.3.5 responds to the ARP request. Each host in the subnet receives the request and checks its IP address for compliance. If it does not match the specified address in the request, the request is ignored.




*The workspace view*

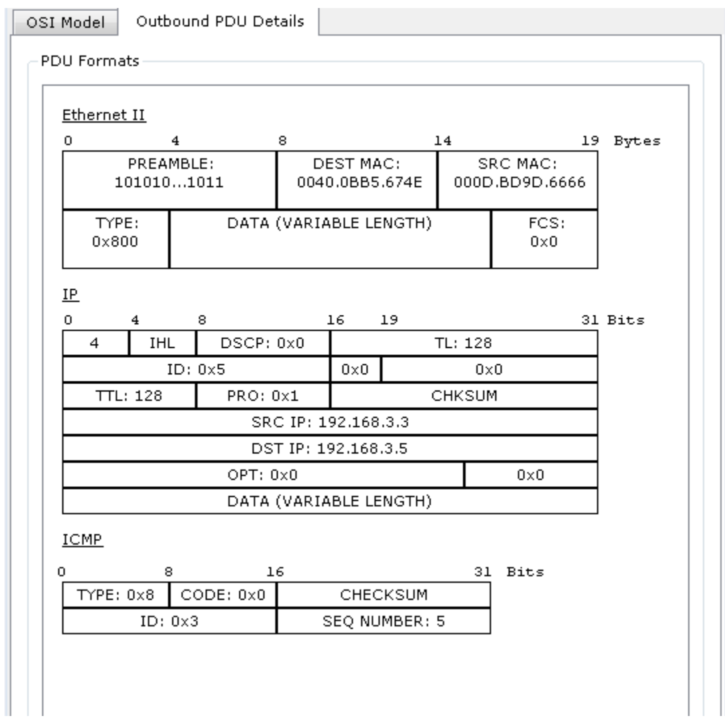
View the contents of the ARP response packet sent to host 192.168.3.3.




*Format of the ARP request packet*

Node 192.168.3.5. sent an ARP response directly to the sender using its MAC address, specifying its own MAC address in the “Target MAC " field.

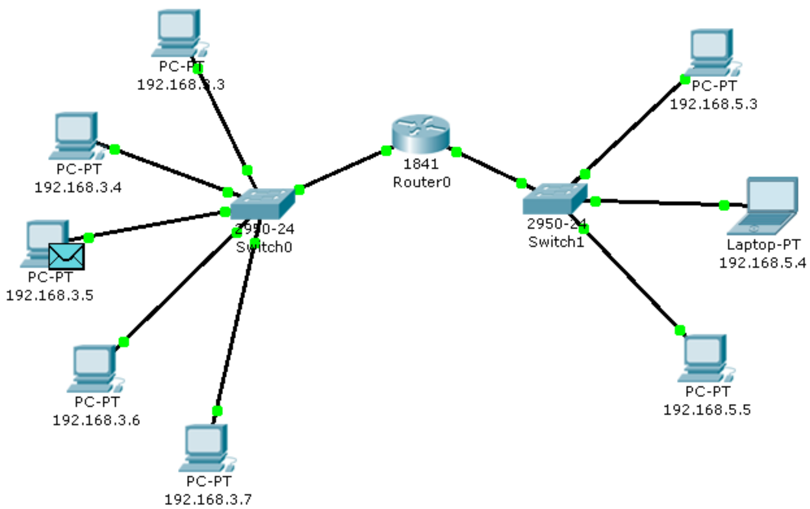
Next, an ICMP ping request message is sent. View the contents of the package by clicking on the package (envelope).




*Format of the ICMP echo request packet*

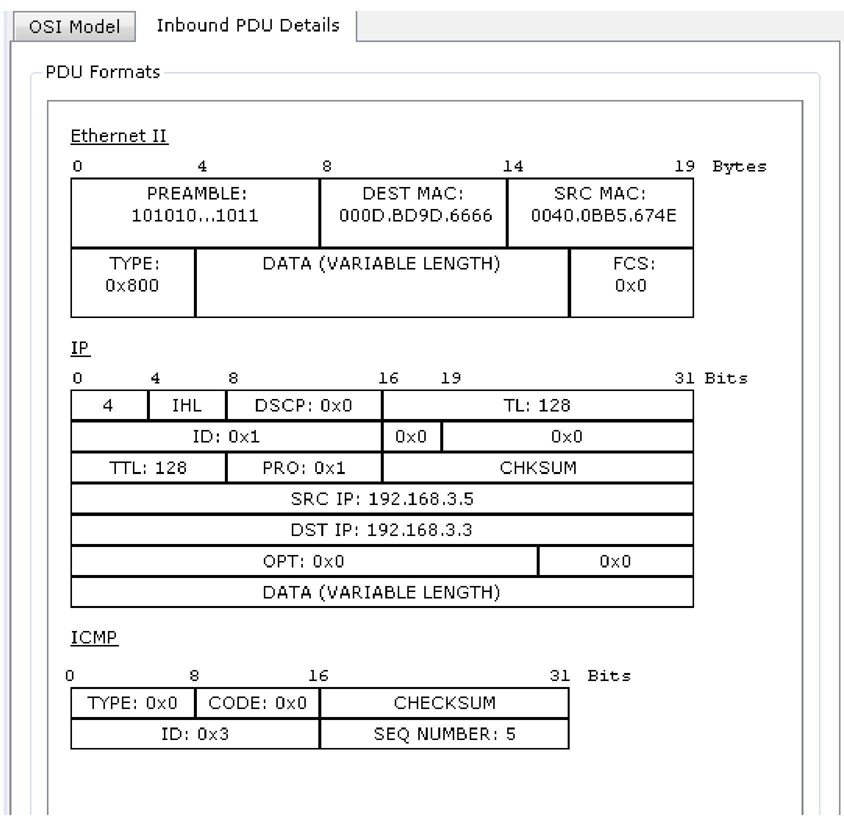
The physical addresses of the nodes are known. The source IP address is 192.168.3.3. the destination IP address is 192.168.3.5. the ICMP message Type is 8 (echo request).

The request is made to the host 192.168.3.5 via the switch:




*The workspace view*

View the contents of the ping response packet sent to host 192.168.3.3.



*Format of the ICMP echo request packet*

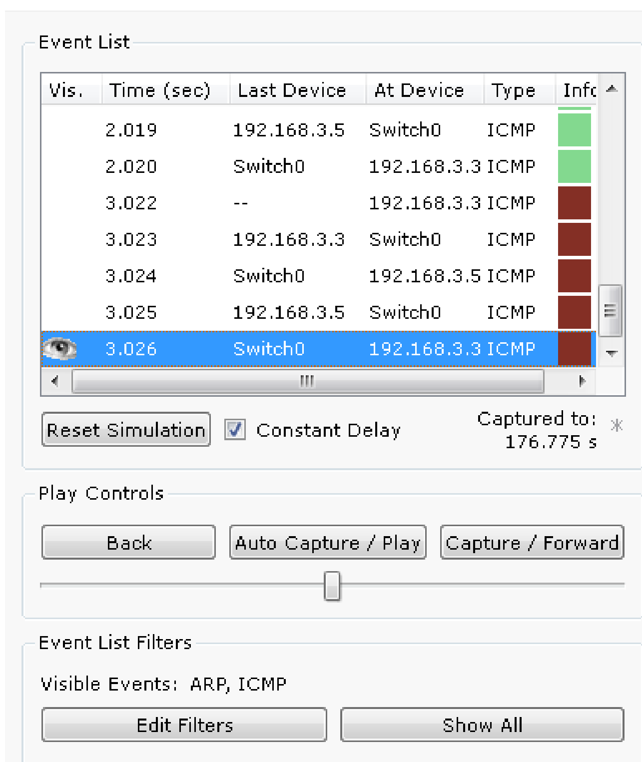
IP source address 192.168.3.5. IP destination address – 192.168.3.3. The ICMP message type 0 (echo reply).

See the ping response in the command line of the host 192.168.3.3.

```
PC>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Reply from 192.168.3.5: bytes=32 time=8ms TTL=128
Reply from 192.168.3.5: bytes=32 time=4ms TTL=128
Reply from 192.168.3.5: bytes=32 time=4ms TTL=128
Reply from 192.168.3.5: bytes=32 time=4ms TTL=128

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 4ms, Maximum = 8ms, Average = 5ms
```

*Ping output*

The event window also shows the ARP and ICMP request routes: which devices the packets passed through.



*The event window of the simulation mode*

You can delete the simulation script using the “Reset Simulation "button or use the "Delete" button in the User Created Packet Window.

Now the ARP tables of hosts 192.168.3.3 and 192.168.3.5 are not empty, they contain a single entry. To view the contents of the ARP table, run the command

“arp-a " on the command line.

Contents of the ARP table of node 192.168.3.3:

```
PC>arp -a
  Internet Address      Physical Address      Type
  192.168.3.1           000d.bddc.ae01        dynamic
  192.168.3.5           0040.0bb5.674e        dynamic
```

*ARP table of node 192.168.3.3 on the command promt*

You can use another method: click on the "Inspect" button, click on the selected device, select "ARP table" and view the node's ARP table entries.

ARP Table for 192.168.3.5

| IP Address | Hardware Address | Interface |
|---|---|---|
| 192.168.3.3 | 000D.BD9D.6666 | FastEthernet |

*ARP table of node 192.168.3.5, shown using the "Inspect" tool*

If you set the ping request to host 192.168.3.5 again, only one ICMP message packet will be generated at once, because the corresponding local address is already stored in the source computer's ARP table.

Try sending the ping request again.

To delete all entries in the ARP table, use the “arp –d " command”.

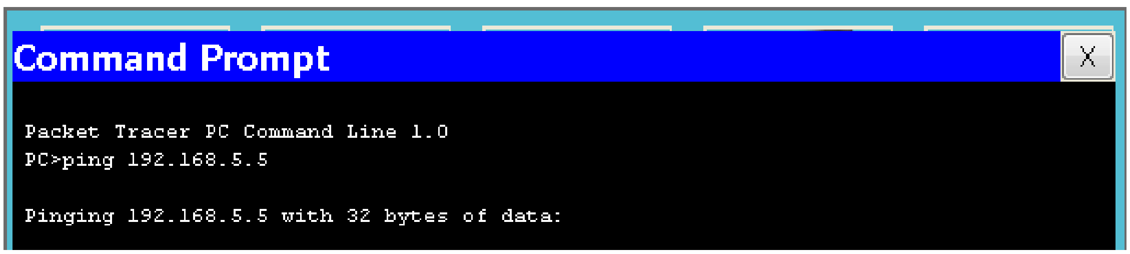
6. Sending a ping request to an external network

Send a test ping request from the destination node with the IP address 192.168.3.4 to the host with the IP address 192.168.5.5.

**Important: one node tries to transmit a packet to another node that is on different networks with it.**

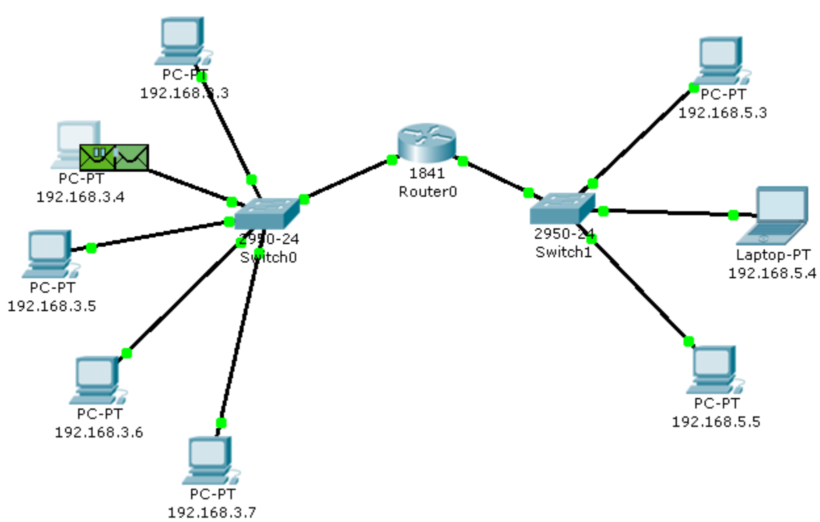
In point 5 of the laboratory work, the case of sending an ARP request inside a local network was considered. In this case, the ARP Protocol directly determined the MAC address of the receiving node of the request. Now consider a situation where the source node and the destination node are located in different networks. The ARP Protocol operates within a network segment, so in this case it will be used to determine the MAC address of the router. This way, the packet will be passed to the router for further retransmission.

Open "Command Promt", which simulates the command line, on the computer 192.168.3.4 and send a ping request to the host 192.168.5.5.



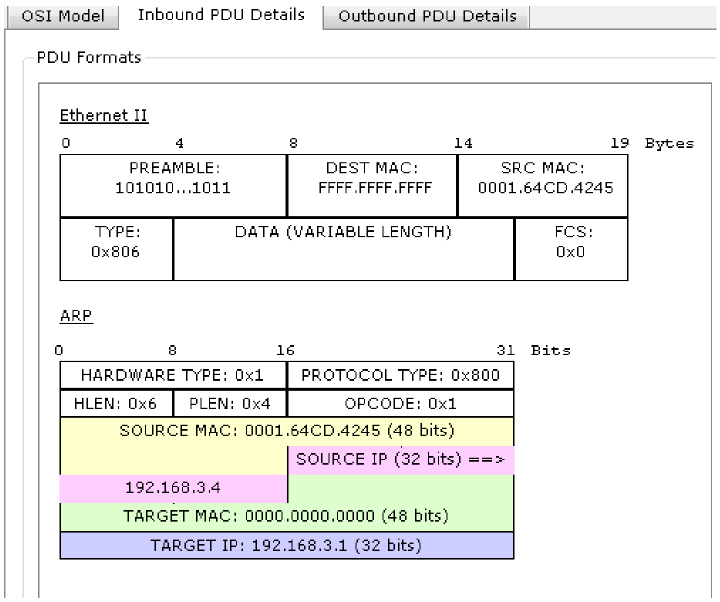

*Host 192.168.3.4 commend promt*

In this case, an ARP request is initiated to the router, which forwards packets to the destination network. Two ARP and ICMP Protocol packets are generated on the source node.



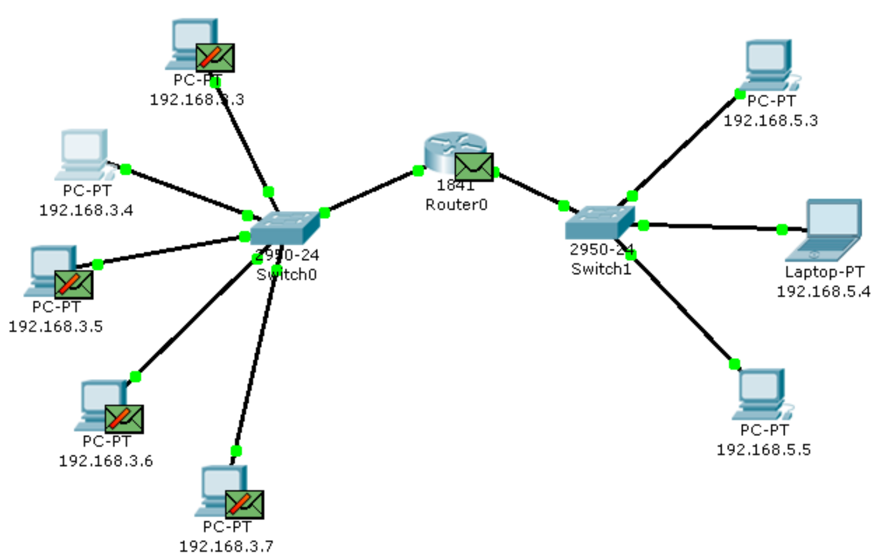
*The workspace view*

The format of the ARP request packet contains the same information as for resolving the local address of the device, and is broadcast to all nodes in the subnet.



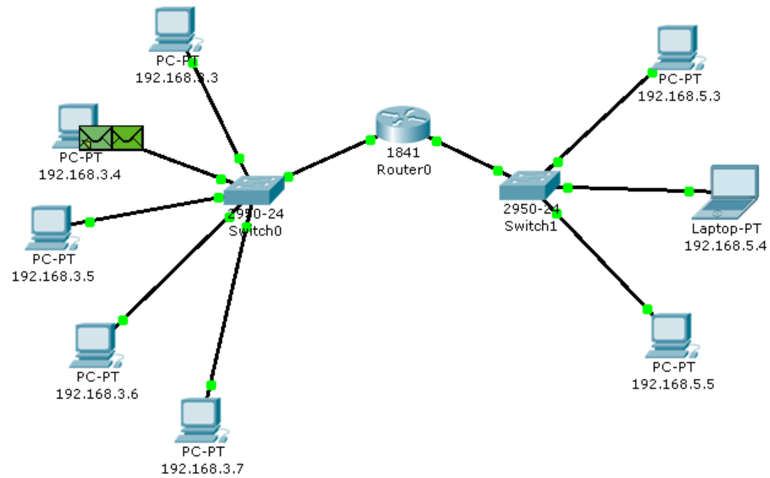

*Format of the ARP request packet*

All nodes ignore the packet, except the router that the packet was intended for.




*The workspace view*

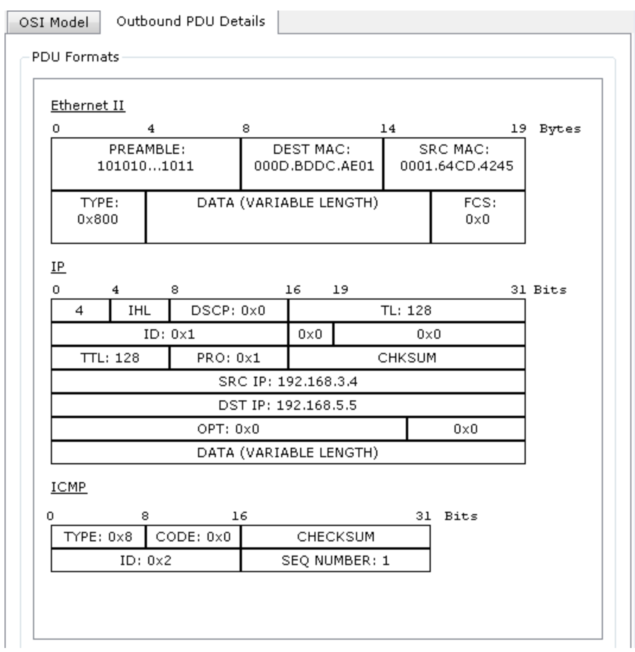
The router generates an ARP response specifying its physical address and sends it to node 192.168.3.4.




*The workspace view*

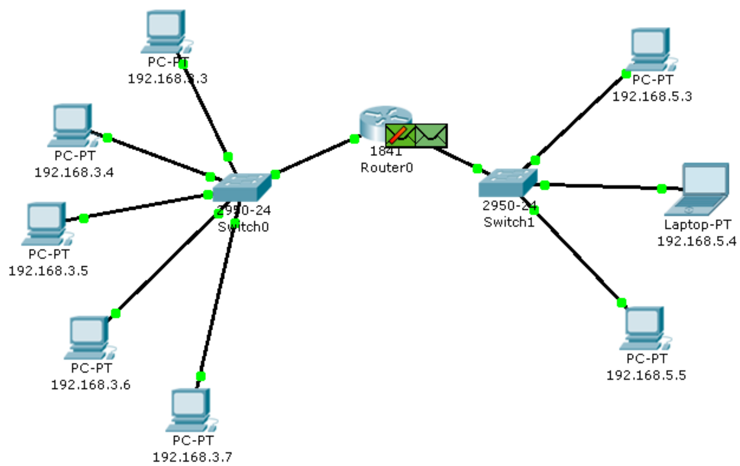
After receiving an ARP response, host 192.168.3.4 sends an ICMP ping request message through the router to the destination network.

View the contents of the package by clicking on the package (envelope).



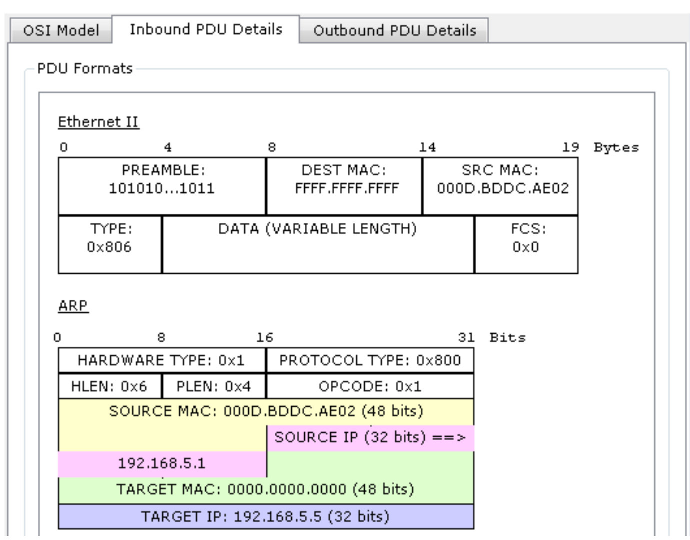
*Format of the ICMP echo request packet*

The source IP address is 192.168.3.4. the destination IP address is 192.168.5.5. the ICMP message Type is 8 (echo request). When a request arrives at the destination network, the router determines the Mac address of the recipient, if there is no Mac address in the router's ARP table. This again solves the problem of resolving the local address.



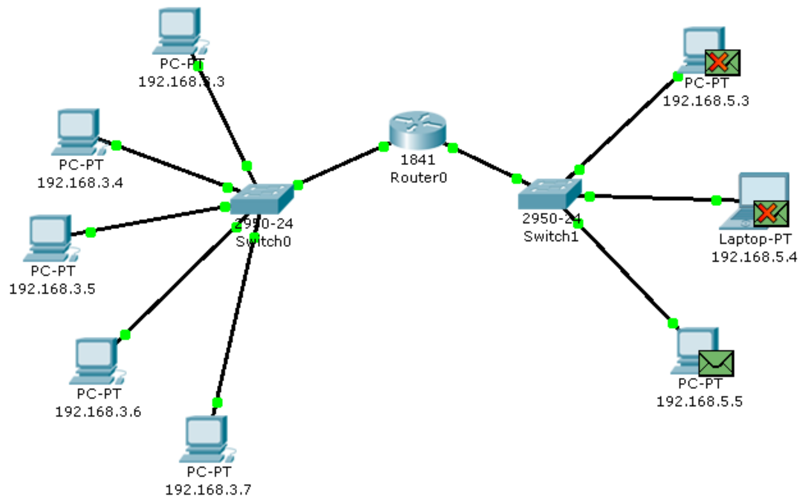

*The workspace view*

The router must first find out the physical address of the recipient before it can send a ping request to the destination, so the ping request packet that arrived at the router is rejected. A new ARP request is sent as a broadcast message from the router, containing its IP address and MAC address. The destination IP address is node 192.168.5.5.



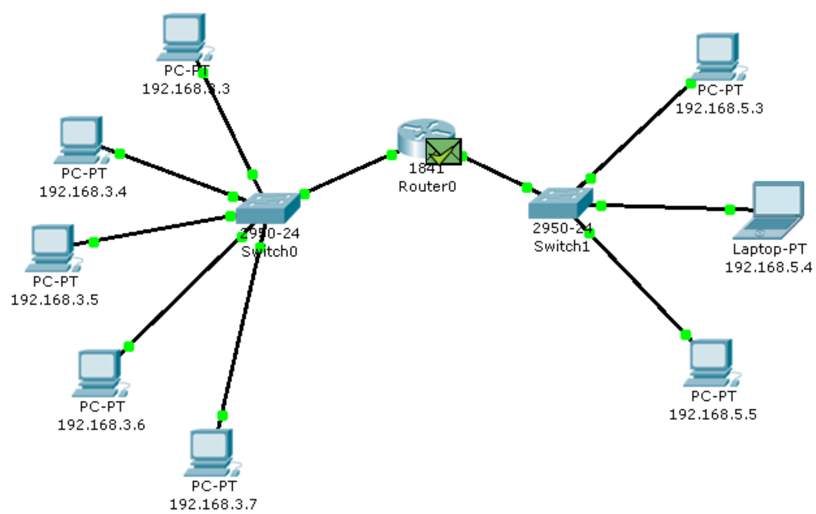
*Format of the ARP request packet*

Subnet nodes that do not receive the packet ignore it.



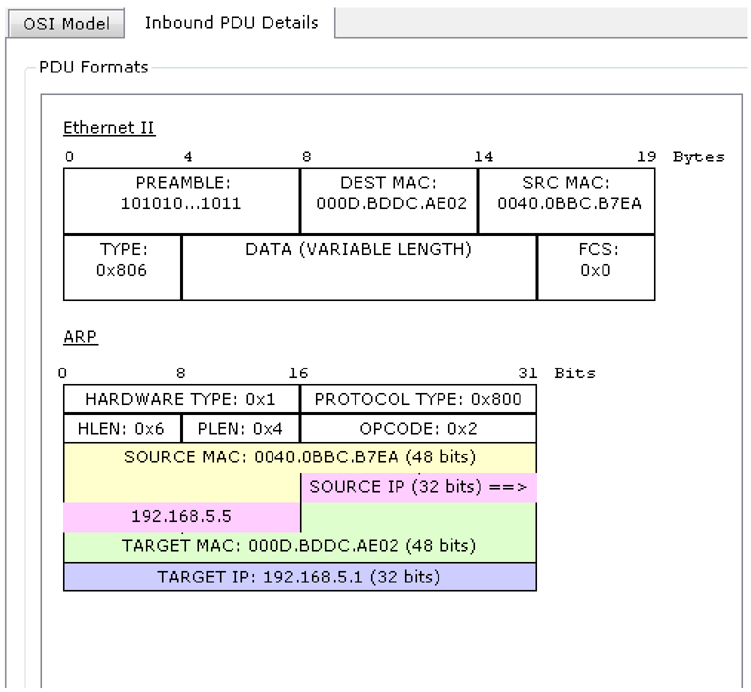

*The workspace view*

Node 192.168.5.5. generates an ARP response and sends it back to the router, specifying its MAC address, as evidenced by the contents of the packet.



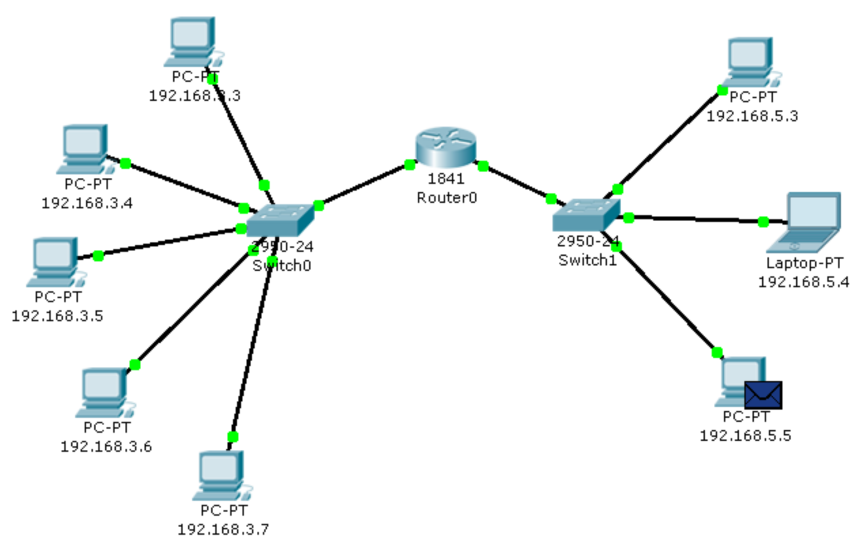

*The workspace view*

After the router determines the Mac address of the recipient of an incoming ping request, it sends an ICMP response to the router of the sender's host. (In this case, it is the same router Router0).



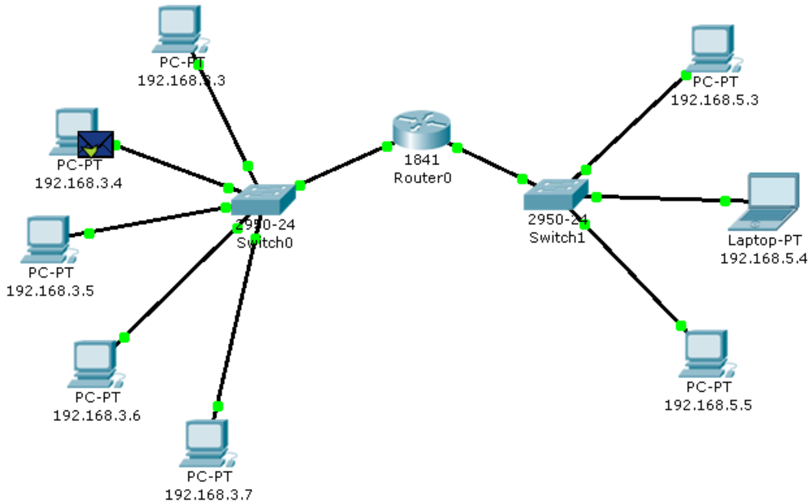

*Format of the ARP response packet*

Node 192.168.3.4. tries again to send a ping request to the external network to node 192.168.5.5. Its route must lie through Switch 0, router Router0, switch1 and reach the destination node. **FOLLOW THE ROUTE OF THE PACKAGE YOURSELF.**



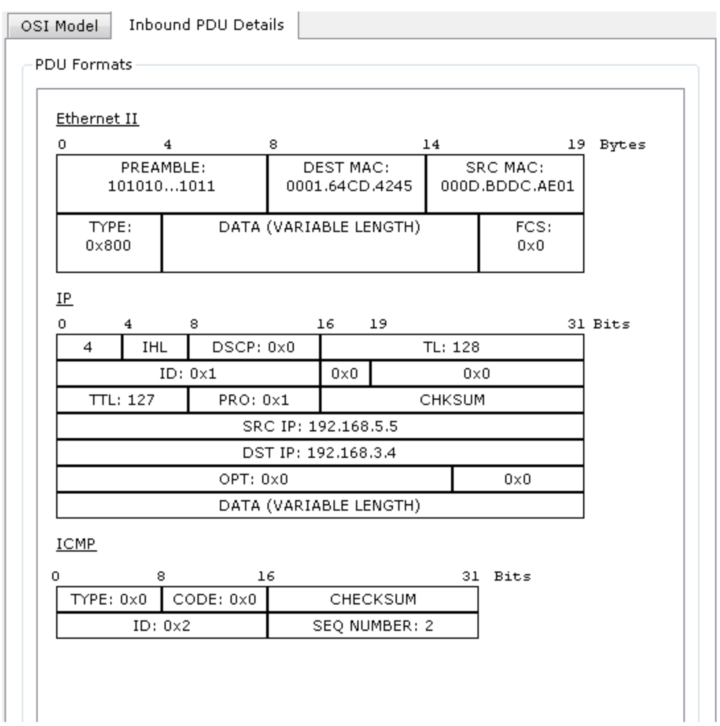

*The workspace view*

The node generates a ping response that is sent back to the node 192.168.3.4.




*The workspace view*

View the contents of the ping response packet sent to host 192.168.3.4.




*ICMP echo response packet format*

IP source address 192.168.5.5. IP destination address – 192.168.3.4. The ICMP message type

0 (echo reply).

See the ping response in the command promt of the host 192.168.3.4.

```
Command Prompt

Packet Tracer PC Command Line 1.0
PC>ping 192.168.5.5

Pinging 192.168.5.5 with 32 bytes of data:

Request timed out.
Reply from 192.168.5.5: bytes=32 time=8ms TTL=127
Reply from 192.168.5.5: bytes=32 time=8ms TTL=127
Reply from 192.168.5.5: bytes=32 time=8ms TTL=127

Ping statistics for 192.168.5.5:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
Approximate round trip times in milli-seconds:
    Minimum = 8ms, Maximum = 8ms, Average = 8ms
```

*Ping output*

You can view the packet route using the tracert command. Run this command, for example, in the command line of the computer 192.168.3.5:

```
PC>tracert 192.168.5.4

Tracing route to 192.168.5.4 over a maximum of 30 hops:

  1   4 ms     4 ms     4 ms     192.168.3.1
  2   8 ms     8 ms     8 ms     192.168.5.4

Trace complete.
```

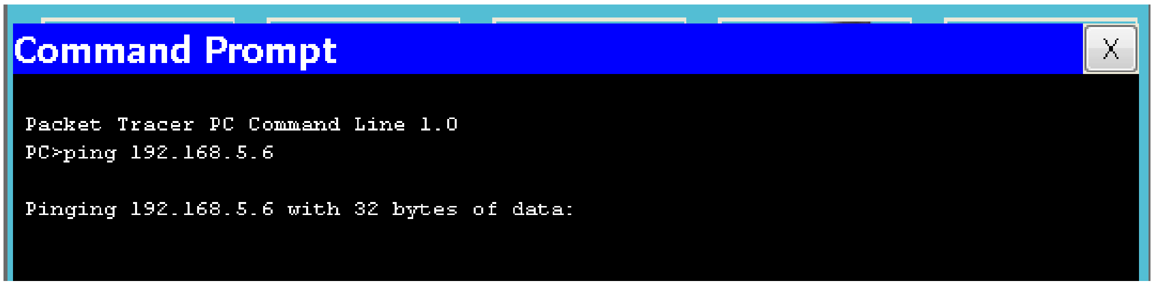
*Tracert outline*

There is one intermediate router on the packet path to host 192.168.5.4.

7. Sending a ping request to a non-existent host

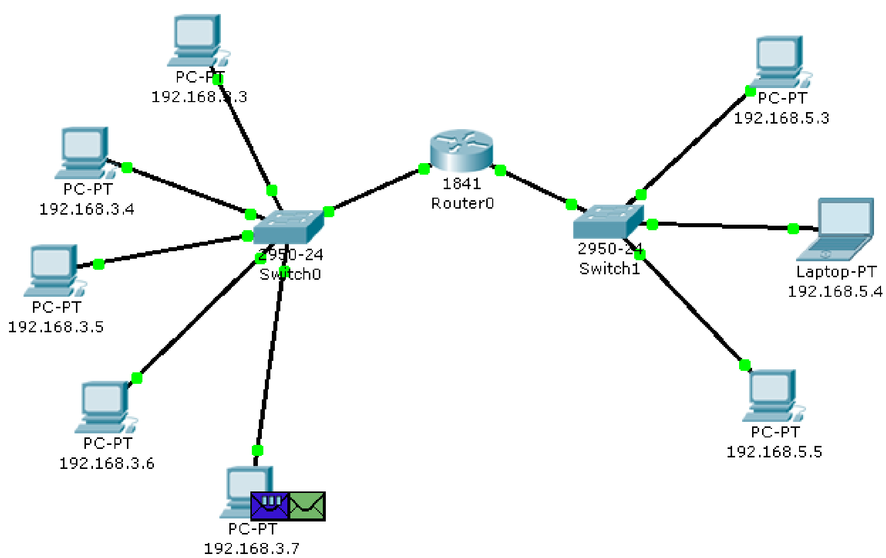
Send a ping request to a non-existent address in the network 192.168.5.0/24.

Open the "Command Promt" program on node 192.168.3.7 and try to send a ping request to a non-existent host with the IP address 192.168.5.6.



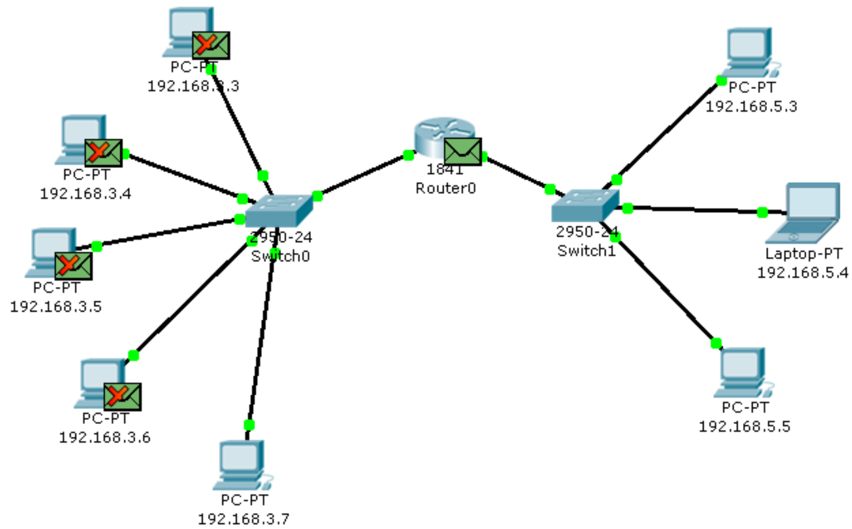

Host command line 192.168.3.7

The ARP table on the source node does not contain a corresponding entry about the MAC address of the node 192.168.5.6, so an ARP request is generated.



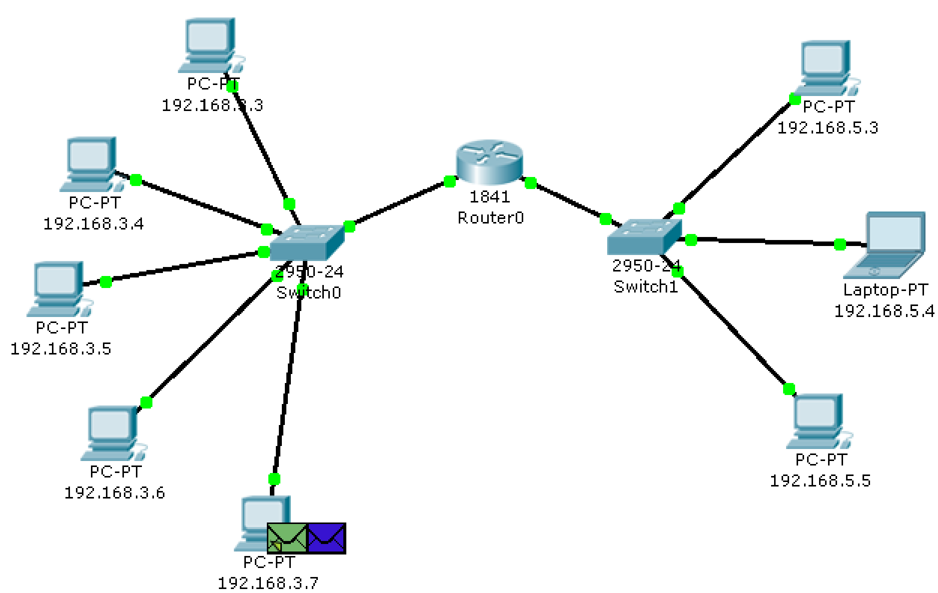

The workspace view

All nodes ignore the packet, except the router that the packet was intended for.



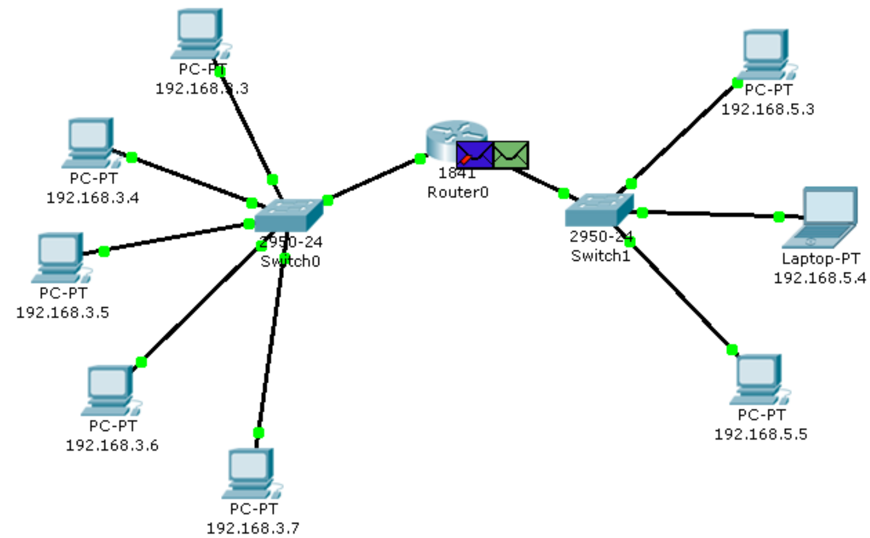

The workspace view

Node 192.168.3.7 receives an ARP response with the MAC address of the router. Now, knowing its hardware address, the host sends a ping request to node 192.168.5.6.



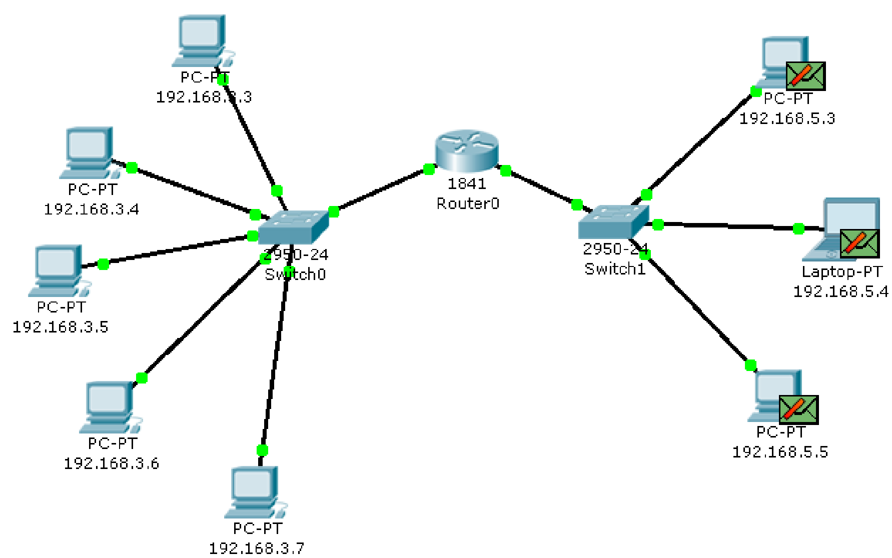

The workspace view

The router destroys the incoming packet, because it cannot redirect it to the specified address, because it "does not know" the corresponding MAC address. In this regard, the router generates an ARP request at the address 192.168.5.6.




The workspace view

All nodes in the subnet ignore the packet because the IP address in the request does not match their own. However, no response is received from anyone by router.

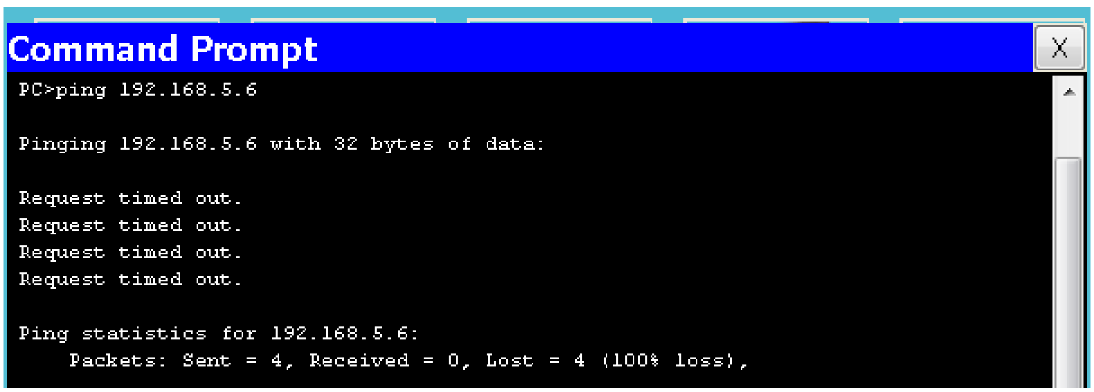



The workspace view

The procedure for passing packets is repeated throughout the simulation scenario: the router still "does not know" the MAC address of the IP address 192.168.5.6 specified in the ping request and continues to send ARP requests. None of the subnet nodes respond to these requests. Without receiving a response, the router itself is "silent", without notifying the host source of the ping request about the error.

Note: in fact, in this case, the router should send an ICMP message "host unreachable»: type 3 message with code 1. However the experiment carried out with the theory went.

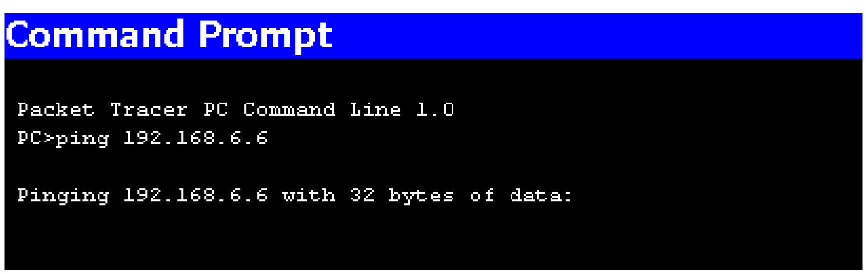
Let's look at the response to the ping request in the command line of the source node 192.168.3.7: "timeout exceeded".




Ping output

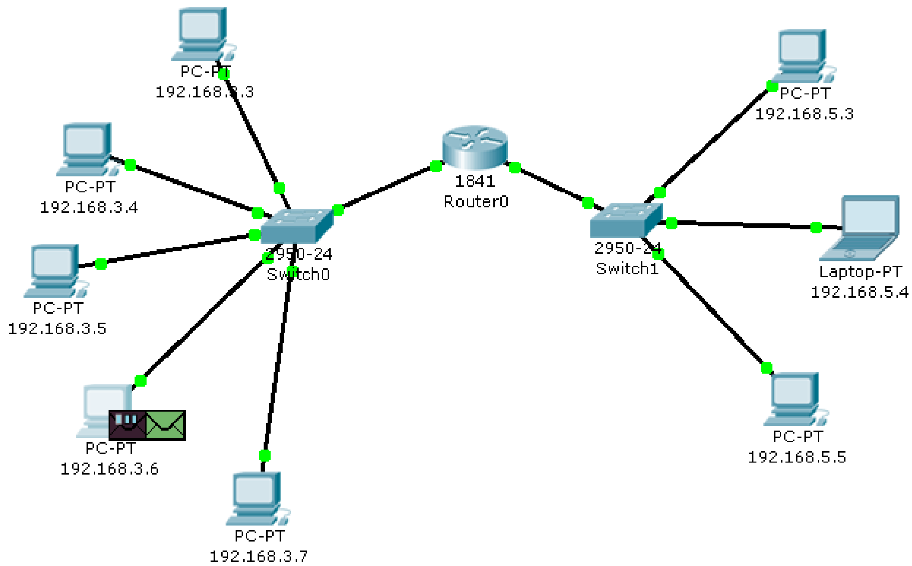
Let's try sending a ping request containing the host's IP address to a network that doesn't have a route.

Open the "Command Promt" program on node 192.168.3.6 and try to send a ping request to a non-existent host with the IP address 192.168.6.6.



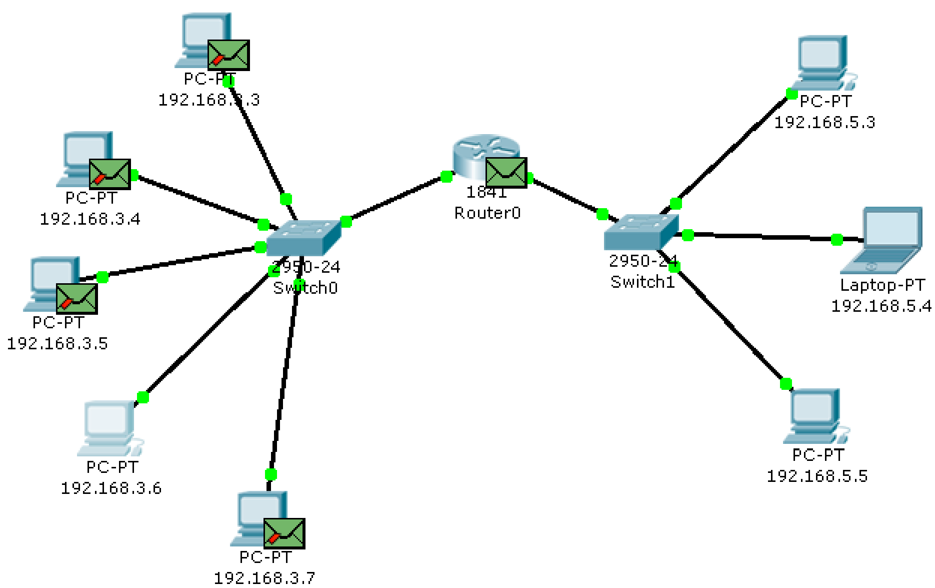

Commant prompt of the node 192.168.3.6

Since the source node's ARP table does not have a corresponding entry, an ARP request is generated for the specified node with the IP address 192.168.6.6.



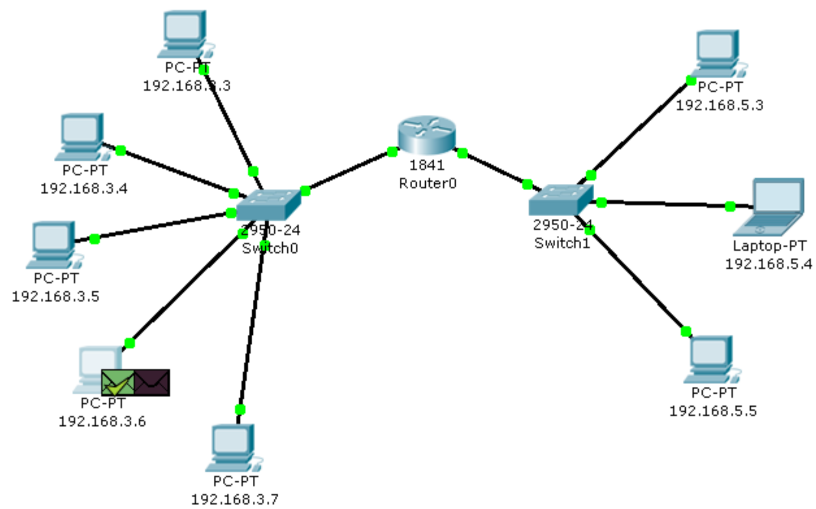

The workspace view

All nodes ignore the packet, except the router that the packet was intended for.



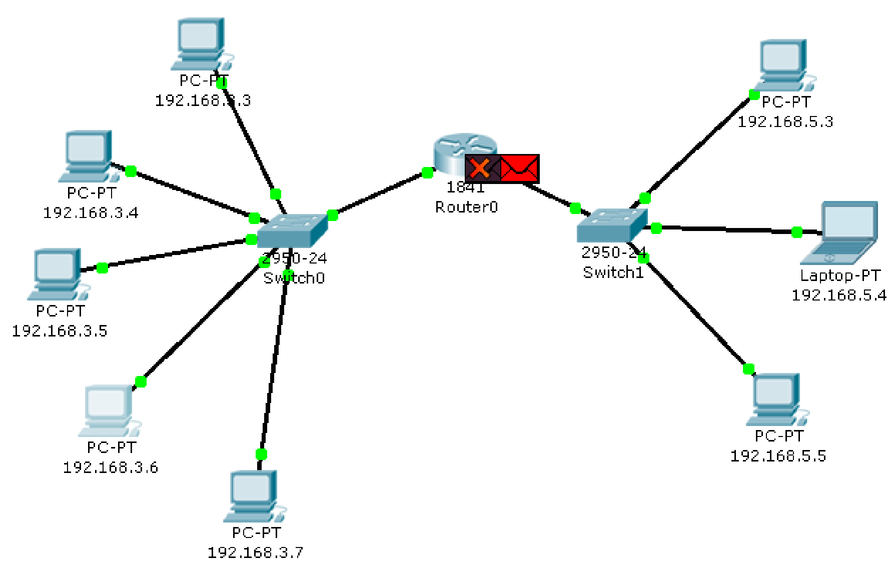

The workspace view

Node 192.168.3.6 receives an ARP response with the MAC address of the router. Now, knowing its hardware address, the host sends a ping request.



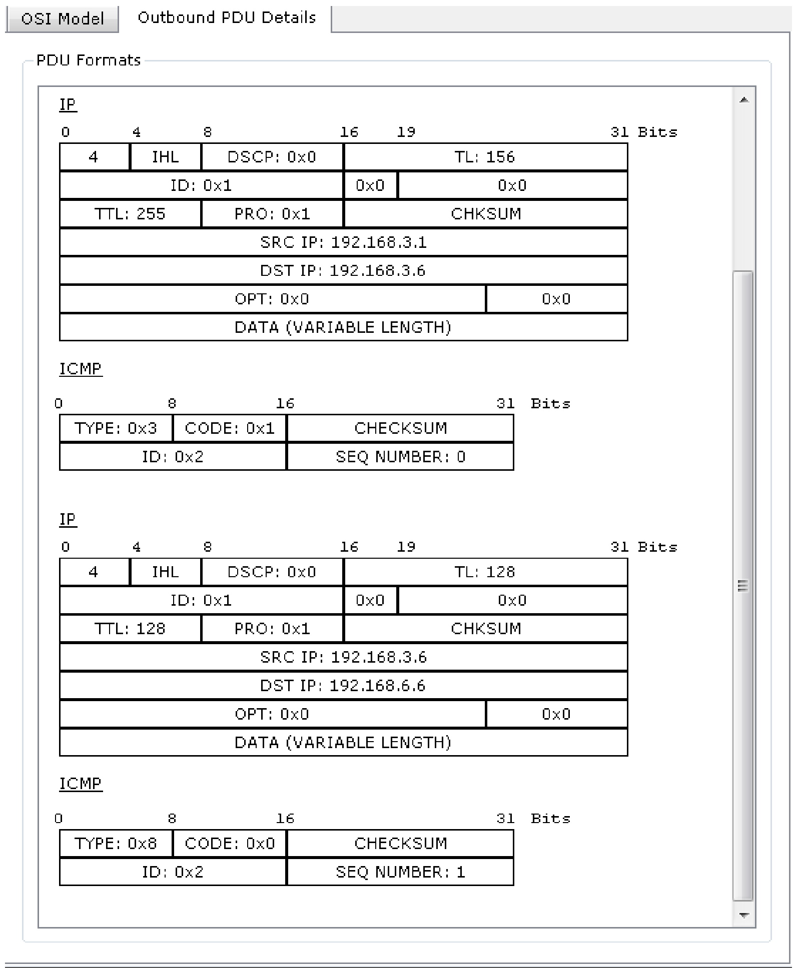

The workspace view

When a ping request reaches the router, it cannot redirect it to any of its interfaces, because the IP addresses of its interfaces do not match the address specified in the ping request. Accordingly, this packet is destroyed and a new ICMP message is generated.




The workspace view

Let's look at the contents of the packet generated by the router.




The packet format of an ICMP “host unreachable”

IP source address – 192.168.3.1. IP destination address – 192.168.3.6. The ICMP message type 3 code 1 means "host unreachable". This packet arrives at node 192.168.3.6.

Result of a ping request in the command line of node 192.168.3.6: "destination host unreachable".

```
Command Prompt

Packet Tracer PC Command Line 1.0
PC>ping 192.168.6.6

Pinging 192.168.6.6 with 32 bytes of data:

Reply from 192.168.3.1: Destination host unreachable.
Reply from 192.168.3.1: Destination host unreachable.
Reply from 192.168.3.1: Destination host unreachable.
Reply from 192.168.3.1: Destination host unreachable.

Ping statistics for 192.168.6.6:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
```

Ping output

Thus, the router "responded" to a ping request for which it did not have a corresponding route with a new ICMP message "host unreachable".

Note: did the router respond correctly in this situation by sending an ICMP "host unreachable" message to the source host? To answer this question, refer to the ICMP Protocol specification RFC 792 and learn about other types of ICMP messages. [Electronic source]. URL: http://tools.ietf.org/html/rfc792.

## 8. Individual task

According to the option, filter ARP and ICMP messages for the specified source-receiver pairs. Each option provides 2 ping request options: inside the network and to the external network. Use the tracert command to view the route of a packet addressed to the external network.

In the report for each test, specify the packet routes, their contents, and explain the results.

REPORT

After completion the task the students need to submit:

- **The format of the report file is *pdf*.**
- Upload reports for the GitLab repository (to your personal project)
- Do not forget to write your name and ID inside the report

Options for individual tasks (table 3).

Table 3

| # | Sender | Receiver |
|---|---|---|
| 1 | 192.168.3.3<br>192.168.3.4 | 192.168.3.4<br>192.168.3.6 |
| 2 | 192.168.3.4<br>192.168.3.5 | 192.168.3.7<br>192.168.5.3 |
| 3 | 192.168.3.5<br>192.168.3.6 | 192.168.3.6<br>192.168.3.7 |
| 4 | 192.168.3.6<br>192.168.3.7 | 192.168.5.4<br>192.168.3.4 |
| 5 | 192.168.3.3<br>192.168.3.7 | 192.168.3.7<br>192.168.5.5 |
| 6 | 192.168.5.3<br>192.168.3.6 | 192.168.5.4<br>192.168.3.4 |
| 7 | 192.168.3.3<br>192.168.3.5 | 192.168.5.3<br>192.168.3.7 |
| 8 | 192.168.3.3<br>192.168.3.4 | 192.168.5.4<br>192.168.3.5 |
| 9 | 192.168.3.4<br>192.168.3.5 | 192.168.5.3<br>192.168.3.4 |
| 10 | 192.168.5.4<br>192.168.3.6 | 192.168.5.5<br>192.168.3.3 |
| 11 | 192.168.3.4<br>192.168.3.7 | 192.168.5.3<br>192.168.5.4 |

| | | |
|---|---|---|
| 12 | 192.168.3.5<br>192.168.3.6 | 192.168.5.5<br>192.168.3.7 |
| 13 | 192.168.3.5<br>192.168.3.7 | 192.168.5.4<br>192.168.3.3 |
| 14 | 192.168.3.6<br>192.168.3.7 | 192.168.5.3<br>192.168.5.5 |

| No. In HDU | Surname, First name in English | Option/variant |
|---|---|---|
| 19321101 | **Huo Yiming** | 1 |
| 19321103 | **Lin Xiaochen** | 2 |
| 19321104 | **Sun Sijie** | 3 |
| 19321105 | **Wang Yining** | 4 |
| 19321106 | **Ying Yi** | 5 |
| 19321107 | **An Yifei** | 6 |
| 19321108 | **Fei Cheng** | 7 |
| 19321109 | **Feng Daoyuan** | 8 |
| 19321110 | **Gu Chengxian** | 9 |
| 19321111 | **Hu Jinghao** | 10 |
| 19321112 | **Huang Gehang** | 11 |
| 19321113 | **Huang Junhao** | 12 |
| 19321114 | **Jiang Xiaoyu** | 13 |
| 19321115 | **Li Sijie** | 14 |

| | | |
|---|---|---|
| 19321116 | **Lin Rui** | 1 |
| 19321118 | **Peng Yin** | 2 |
| 19321119 | **Shen Jiahui** | 3 |
| 19321120 | **Song Mingxin** | 4 |
| 19321121 | **Wang Qian** | 5 |
| 19321122 | **Wu Xiyuan** | 6 |
| 19321123 | **Xiao Shitong** | 7 |
| 19321124 | **Xie Zhixuan** | 8 |
| 19321125 | **Xu Tian** | 9 |
| 19321126 | **Yan Qiaosen** | 10 |
| 19321127 | **Yang Chengzhi** | 11 |
| 19321128 | **Ye Hongyi** | 12 |
| 19321130 | **Yu Junlong** | 13 |
| 19321131 | **Yuan Chengpeng** | 14 |
| 19321132 | **Zhou Zhengyu** | 1 |
| 19321133 | **Zhu Haibin** | 2 |
| 19321134 | **Zhu Hangtao** | 3 |
| 19321201 | **Cao Yunxi** | 4 |
| 19321202 | **Chen Jiao** | 5 |
| 19321203 | **Lin Yanjun** | 6 |
| 19321204 | **Niu Jia** | 7 |
| 19321205 | **Su Ruiyuan** | 8 |
| 19321206 | **Zheng Liangyin** | 9 |

| | | |
|---|---|---|
| 19321207 | **Chen Jiawei** | 10 |
| 19321208 | **Chen Jiarui** | 11 |
| 19321209 | **Chen Zhaowei** | 12 |
| 19321210 | **He Jinglin** | 13 |
| 19321211 | **Li Jizhou** | 14 |
| 19321212 | **Li Yitong** | 1 |
| 19321213 | **Lin Yizhang** | 2 |
| 19321214 | **Liu Fuchuan** | 3 |
| 19321215 | **Lu Bin** | 4 |
| 19321217 | **Meng Chenxu** | 5 |
| 19321218 | **Ran Ziqi** | 6 |
| 19321219 | **Wang Canxiong** | 7 |
| 19321220 | **Wang Chaoyang** | 8 |
| 19321221 | **Wei Qingyun** | 9 |
| 19321224 | **Yan Xiangbo** | 10 |
| 19321225 | **Yang Ao** | 11 |
| 19321226 | **Ye Haoyang** | 12 |
| 19321227 | **Yu Hao** | 13 |
| 19321228 | **Yu Hao** | 14 |
| 19321229 | **Yu Chuyang** | 1 |
| 19321230 | **Zhang Jinchuan** | 2 |
| 19321231 | **Zhao Zihang** | 3 |
| 19321232 | **Zhou Hao** | 4 |

| | | |
|---|---|---|
| 19321233 | **Zhou Zhongrui** | 5 |
| 19321301 | **Shen Mengxin** | 6 |
| 19321302 | **Xu Yuansui** | 7 |
| 19321303 | **Yuan Jing** | 8 |
| 19321304 | **Zhang Weixi** | 9 |
| 19321305 | **Zhou Yuliu** | 10 |
| 19321306 | **Bai Yusong** | 11 |
| 19321307 | **Chen Hao** | 12 |
| 19321308 | **Chen Ziyuan** | 13 |
| 19321309 | **Gu Jiajun** | 14 |
| 19321310 | **Guo Xiaolei** | 1 |
| 19321311 | **Han Weiwei** | 2 |
| 19321312 | **He Jian** | 3 |
| 19321314 | **Jing Xuan** | 4 |
| 19321315 | **Li Haowei** | 5 |
| 19321316 | **Li Weitao** | 6 |
| 19321317 | **Li Xinjiang** | 7 |
| 19321318 | **Li Zhenpeng** | 8 |
| 19321319 | **Luo Weizhi** | 9 |
| 19321320 | **Meng Weijie** | 10 |
| 19321321 | **Peng Haonan** | 11 |
| 19321322 | **Shen Yu** | 12 |
| 19321323 | **Sun Yichen** | 13 |

| | | |
|---|---|---|
| 19321324 | **Tan Yangke** | 14 |
| 19321325 | **Tang Yuxuan** | 1 |
| 19321326 | **Wang Chen** | 2 |
| 19321327 | **Wang Ting** | 3 |
| 19321328 | **Yang Han** | 4 |
| 19321329 | **Zhang Jianwei** | 5 |
| 19321330 | **Zhang Leshan** | 6 |
| 19321331 | **Zhang Tianyu** | 7 |
| 19321332 | **Zheng Lei** | 8 |
| 19321333 | **Zhou Hanyu** | 9 |