CSG 1105 / 5130 - Applied Communications

Week 5 Tutorial

Objectives

- To learn router configuration at a basic level
- To reinforce switch configuration at a basic level
- To see how routers affect broadcast traffic on a network
- To see how to internetwork two different subnets
- Use Wireshark to analyse real network traffic

By the end of this workshop you should be able to

- Use Packet Tracer to configure switches and routers at basic level
- Explain a routers effects on a network and broadcast traffic
- Use Wireshark confidently to find out information about network traffic

Required Tools & Documents

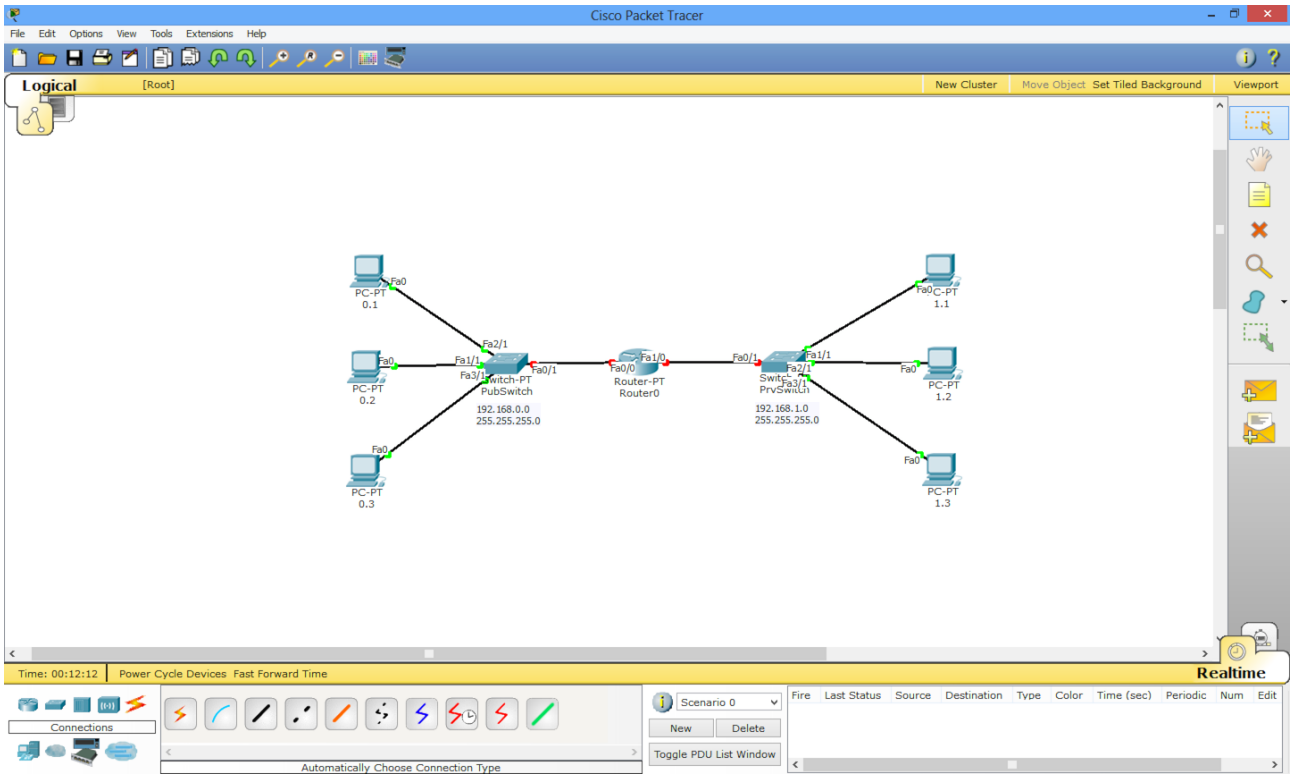
- Packet Tracer (Available in Unit Documents on Blackboard)
- Wireshark (Available in Unit Documents on Blackboard)
  - Note: Mac users also need XQuartz (included when downloaded from Blackboard)
- An internet connection
- Week 5 Tutorial Logical Topology PDF
- Packet Tracer Simulation Guide (Available in Unit Documents on Blackboard)

Optional Downloads

- Week 2 Tutorial Documentation
- A recording of this Tutorial from blackboard; discussions are held in class and clarifications are made too.

Task 1 - Packet Tracer Routing Basic

1. In Packet Tracer using the generic router and generic switches and Workstations build the network topology found in the PDF titled Week 5 Logical Network Topology. It should look like below:



PubSwitch		
PC0	192.168.0.1	255.255.255.0
PC1	192.168.0.2	255.255.255.0
PC2	192.168.0.3	255.255.255.0

PrvSwitch		
PC4	192.168.1.1	255.255.255.0
PC5	192.168.1.2	255.255.255.0
PC6	192.168.1.3	255.255.255.0

2. You will notice that the two networks connected via the router are of different subnets. Under normal circumstances they would not be able to communicate to each with their current subnet masks. This is because the 3rd set of 255 indicates that the communication can only come from the same subnet. A router is the solution to this problem and will allow us to internetwork our two separate subnets.
3. Assign your workstations the following IP Addresses and Subnet Masks, be sure to note that there is only 3 workstations in each subnet.

4. You'll notice that they will show green status lights between their own networks, but the connection between the switches and the router remain red. This is because although a switch can be configured, it will work without any configuration at all. A router however has to be configured and have the ports enabled for them to function.
5. To reinforce our learning from previous weeks, we will setup our switches first and then look into the router. Proceed to configure the following points on your switches. Keep in mind you can always use `TAB` and `?` to help you with the commands. I will provide the initial commands to bring you into privileged mode and will also provide the first word of the configuration command. You can also look back at Week 2 for more in-depth help of these steps.
    1. Enable privileged mode - `enable`
    2. Enter configuration mode through terminal - `configure terminal`
    3. Set the hostname - `hostname …`
    4. Set the time zone - `clock …`
    5. Set the banner - `banner …`
    6. Exit configuration mode - `exit`
    7. Save the running config to NVRAM - `write memory`
6. That last step is a new concept. When you configure a Cisco device all the configuration is only stored in it's RAM, it's volatile memory. Volatile memory is cleared whenever power is cut from the device, so if it is restarted or a blackout occurs we would need to reconfigure the device. NV RAM is **non**-volatile memory. This is memory which is not cleared when power is cut from the device, this is a crucial step in configuring Cisco devices.
7. Now configure your other switch with it's information, then we'll move onto the router.
8. Configuring a router requires more steps than a switch, but the language used, and shortcuts mentioned in step 5 are still valid. Let's get started on configuring the router.
9. Open your router and go to the `CLI` tab and you should notice that it immediately has a prompt for us:
    ```
    Continue with configuration dialog? [yes/no]:
    ```
10. Typing '`yes`' or '`y`' at this point will allow you to answer a series of questions to aid in setting up the router with basic configuration for a single port. For the purpose of reinforcing learning and this tutorial **we will answer '`no`' or '`n`'** to configure the device manually.
11. You should be able to press Enter (Return) to get started, do so and you should see a familiar command line appear, as below:
    ```
    Router>
    ```
12. At this point the commands to get to the part we need are the same as the Switch, we want to enable privileged mode and enter configuration mode through terminal. These steps are identical to those in step 5. (More over page).

13. Firstly, let's prevent our router from trying to 'translate' our typo's into something, enter the following command:

```
BorderPatrol#no ip domain-lookup
```

14. Then, let's set up the hostname, time zone and banner once more, using the same commands as above. I will call my router 'BorderPatrol'. Just like with the switches, once you assign a hostname the prompt should change to that, for example:

```
BorderPatrol#
```

15. Once we have setup the house keeping side of things we can get into configuring the ports to enable network connectivity.

    You may have noticed when configuring the IP Address and Subnet Masks there is also a text box to type in the Default Gateway. We have been leaving this blank up until now, the reason behind this is that the default gateway is the address of the port that allows the subnet traffic to leave the subnet and travel into other networks. A good example of this is your modem at home. It's IP Address that you browse to to configure the modem is the default gateway, it is the port that has access to the network that isn't your house. We will now configure the default gateway for our subnets.

    I've named my router Border Patrol because a router is always between two or more different subnets, as a border between them.

16. The default gateway is the port on the router that the switch connects to. We must assign this port an IP address and subnet mask in the same range as the network that is connected to it. So for the left port, it will be on the 192.168.0.0 with 255.255.255.0 subnet, and the right port it will be on the 192.168.1.0 with 255.255.255.0 subnet.
17. To configure the ports you need to know the name of the interface you wish to configure. On a Cisco switch a standard ethernet port is called FastEthernet and then has the module number and port number. Cisco devices are modular, this means the physical configuration can be modified if they are powered down. By default, the generic routers we have used have single port ethernet modules in slots 0 and 1. Let's configure port 0 for FastEthernet in slot 0 (this should be your PubSwitch connection). Remember, you can always use `TAB` to auto complete the word and `?` to find out what is next.

```
BorderPatrol(config)#interface FastEthernet 0/0
```

18. You'll notice our prompt has changed once again to say config-if. 'if' is short for interface in Cisco devices. This means we are in interface configuration mode. Now we'll need to set it's IP address and subnet mask as below (copy these numbers exactly):

```
BorderPatrol(config-if)#ip address 192.168.0.250 255.255.255.0
```

19. If you ever accidentally type the wrong values, you can simply type 'no' before the command (without the numbers) to reverse this.
20. Let's make use of this 'no' technique right now. All the ports on our router are currently shutdown. We'll reverse this on this port using the following:

```
BorderPatrol(config-if)#no shutdown
```

21. You should get the following messages indicating that two things have happened, firstly, that the port is now powered (line up) and it has a valid IP address assigned (protocol up):

```
%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to
up
%LINEPROTO-5-UPDOWN: Line protocol on Interface
FastEthernet0/0, hanged state to up
```

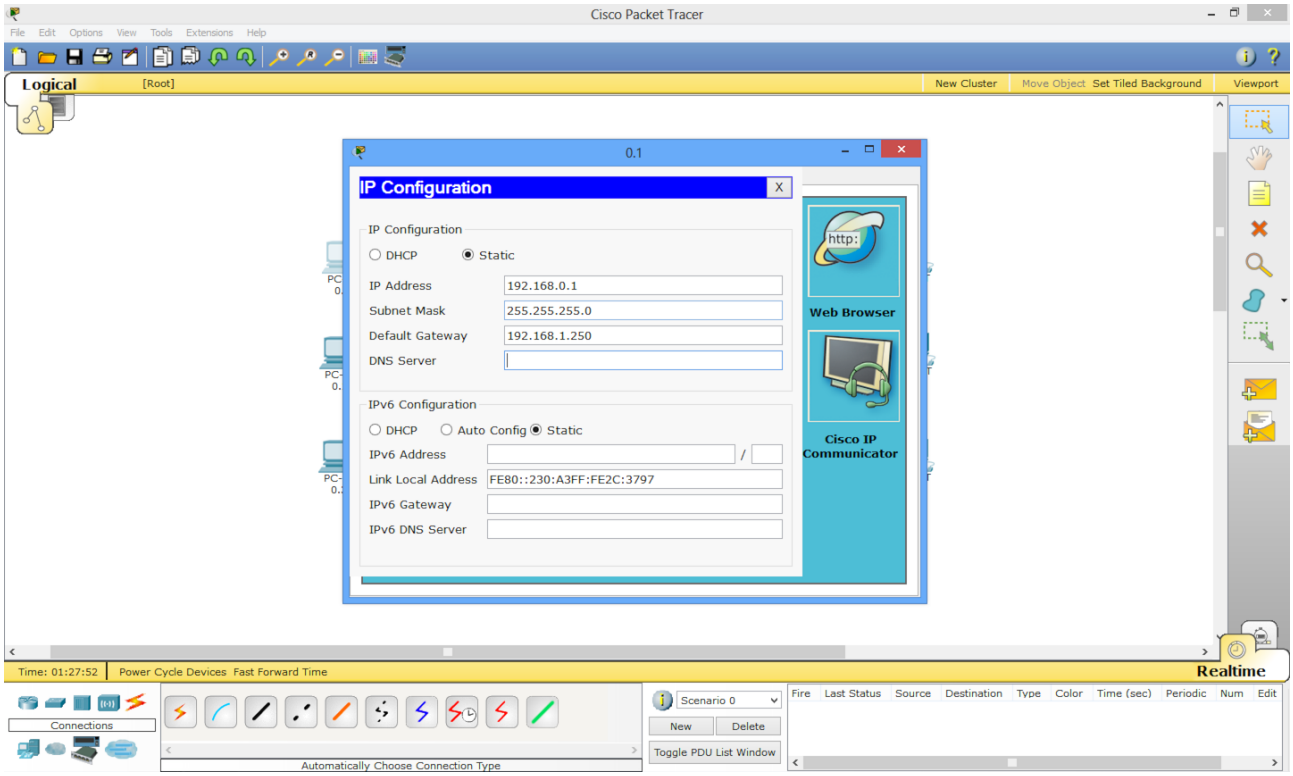
22. Now follow those same steps to configure FastEthernet 1/0 on your router. Type 'exit' to return to normal configuration mode and then start from step 17 to configure it with the same command but using 1/0. Use the following IP address and subnet mask for it:

    192.168.1.250 255.255.255.0

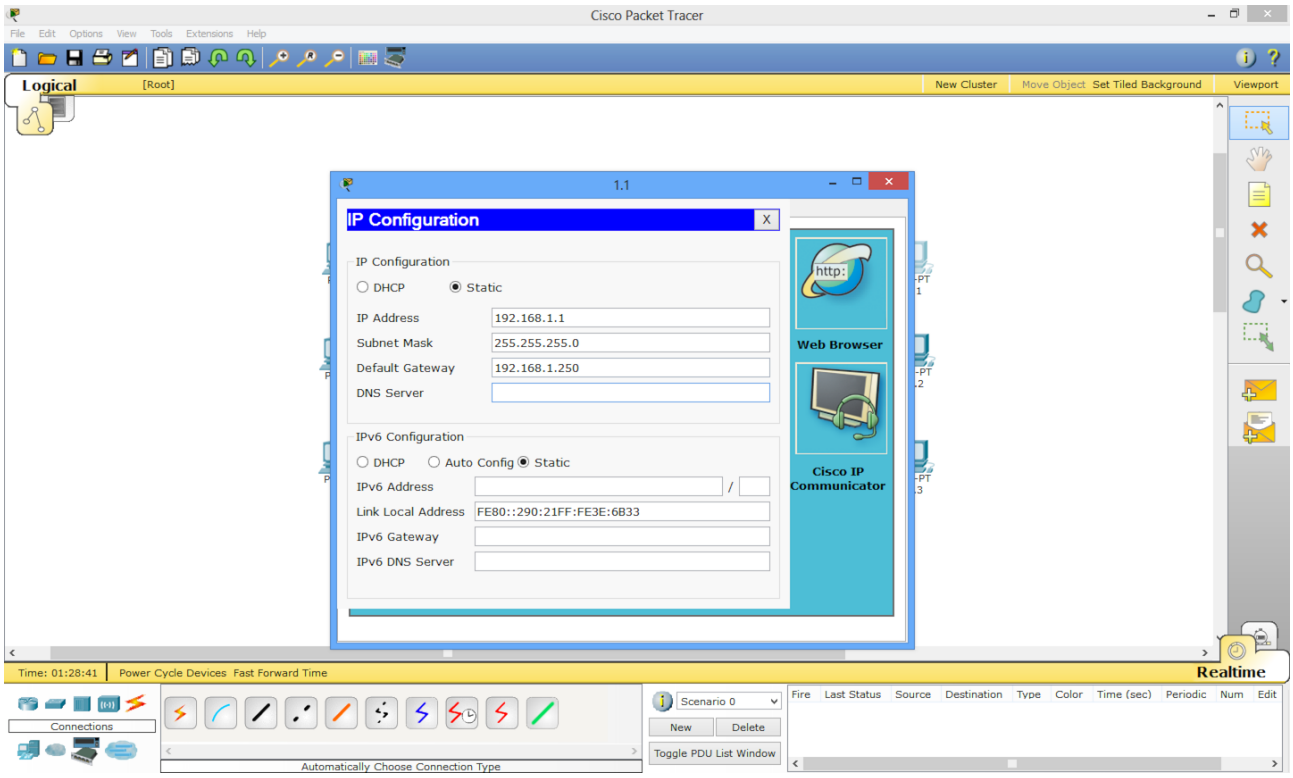
23. Congratulations! Both your ports should now be active and showing green status lights! Don't forget to exit to configuration mode and **save the configuration like in step 5 command 7**.

24. Now we need to tell our computers where their exit to the rest of the network is. Go back and configure your computers to have their default gateways as specified below:

192.168.0.0 Subnet - Default Gateway: 192.168.0.250



192.168.1.0 Subnet - Default Gateway: 192.168.1.250

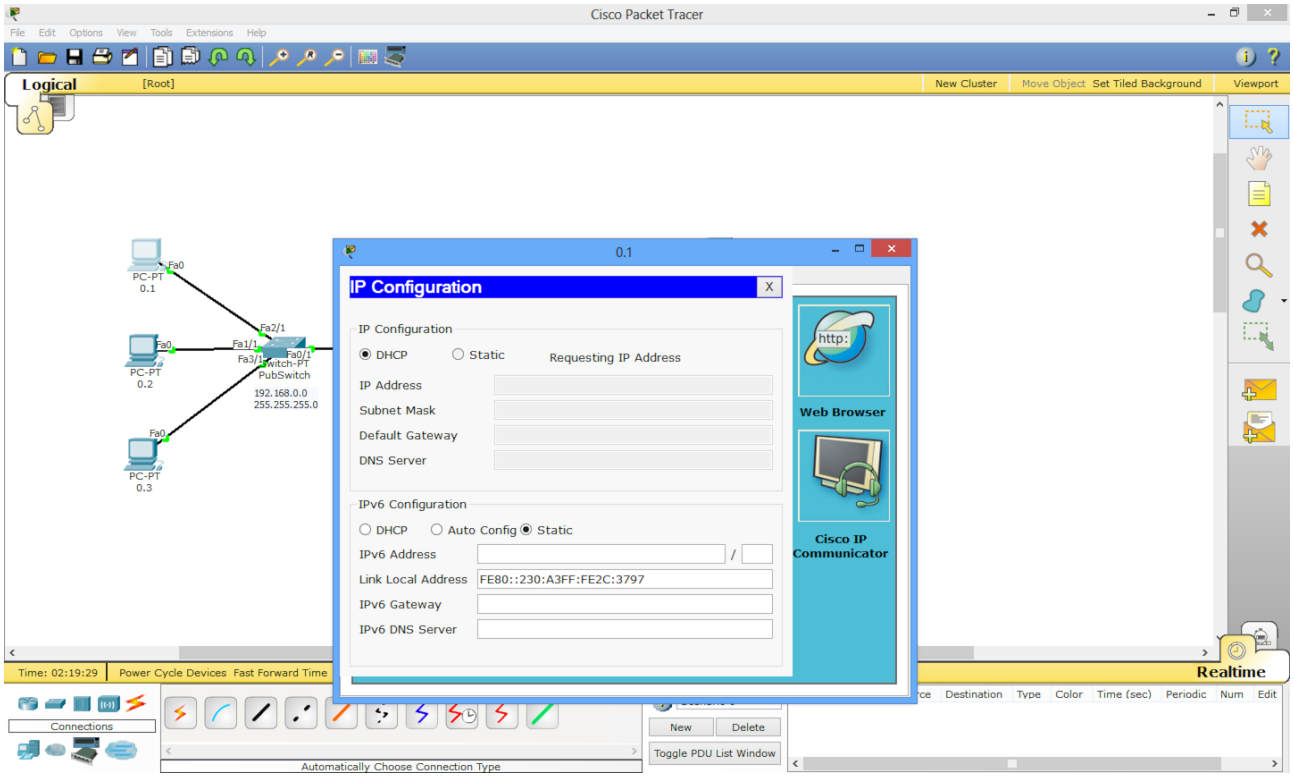


25. Try running the single packet simulation found in the Packet Tracer Simulation Guide as per the required downloads then consider the questions in the guide. Points to consider on next page.

The traffic when using the ping command on the router failed at first as the router had not yet built up it's own ARP table. Meanwhile the router knows where the network is, it still has not related a physical address to a logical address. Sometime between the first ICMP ping and the second ICMP, it has now learned the location (by sending an ARP request), and can now route the ICMP ping from our source to the destination.

Task 2 - DHCP vs Static IP & DHCP Traffic

1. Before we look into DHCP and it's benefits, let's remove the static IP addresses we have assigned our workstations. Go into one of the workstations 'Desktop' tab and click on the 'IP Configuration' icon. Then select the DHCP option, as below:



2. You'll notice after a short while it will say that the 'DHCP request failed'. That's because we haven't configured DHCP yet. What is DHCP you ask?

   DHCP means Dynamic Host Configuration Protocol which is a protocol that automatically leases out IP addresses to network enabled devices which send out a request for one. It has the ability to reserve IP addresses, or ranges of IP addresses for multiple purposes such as a statically set IP address (for a server as an example), or to limit the pool available. It enables a network to be dynamic and removes the need for the network administrator to manually assign every IP address in use.

3. Let's learn how to turn our router into a DHCP server so that our workstations can have their IP addresses dynamically set.
4. Open your router to it's CLI tab again and enter configuration mode, like below:

```
BorderPatrol(config)#
```

5. Our first step is to declare our valid IP ranges, or pools, that can be used and what the default router (gateway) is for each of these pools, start by entering pool configuration. This is done with the following command:

```
BorderPatrol(config)#ip dhcp pool NAME
```

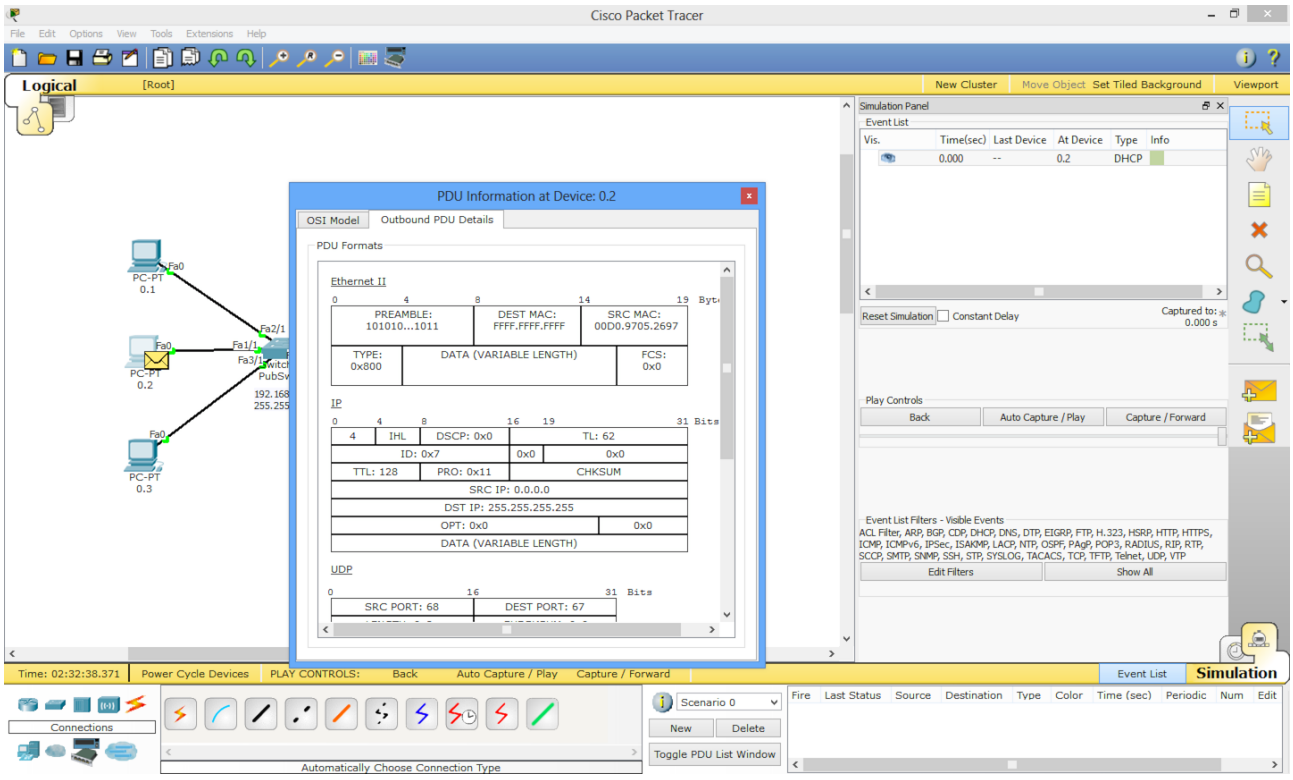
   You can name your dhcp pool anything, I will name mine '`public`'.
6. You will once again notice that our prompt has changed to indicate we are configuring the dhcp pool. Now, let's assign the network (subnet) to be used in this pool, we'll start with our public switches network of 192.168.0.0 with 255.255.255.0. This is done similarly to how we set the interfaces IP address:

```
BorderPatrol(dhcp-config)#network 192.168.0.0 255.255.255.0
```

7. Now that we have our allocated network for public declared, let's tell that pool what it's default router (gateway) will be:

```
BorderPatrol(dhcp-config)#default-router 192.168.0.250
```

8. Now exit the dhcp pool configuration and re-enter to create our other dhcp pool, mine will be called private. This time use the same commands, but use the 192.168.1.0 network instead.
9. Exit to privileged mode and save the configuration to NVRAM again like in step 5 part 7.
10. We will now witness a DHCP request in simulation mode. Set your Packet Tracer into simulation mode and follow the steps below.
11. Open a different workstation and go to the 'Desktop' tab and open 'IP Configuration', change this to DHCP and then step through (using 'Capture/Forward' not auto capture) and take note on the process of obtaining an IP address.
12. First, a DHCP Discover packet is generated at the source (workstation) and sent out via broadcast. If you click on the packet you can analyse the information in the DHCP request. You'll notice the destination address is a physical (MAC) address of `ff:ff:ff:ff:ff:ff` - the broadcast physical address:

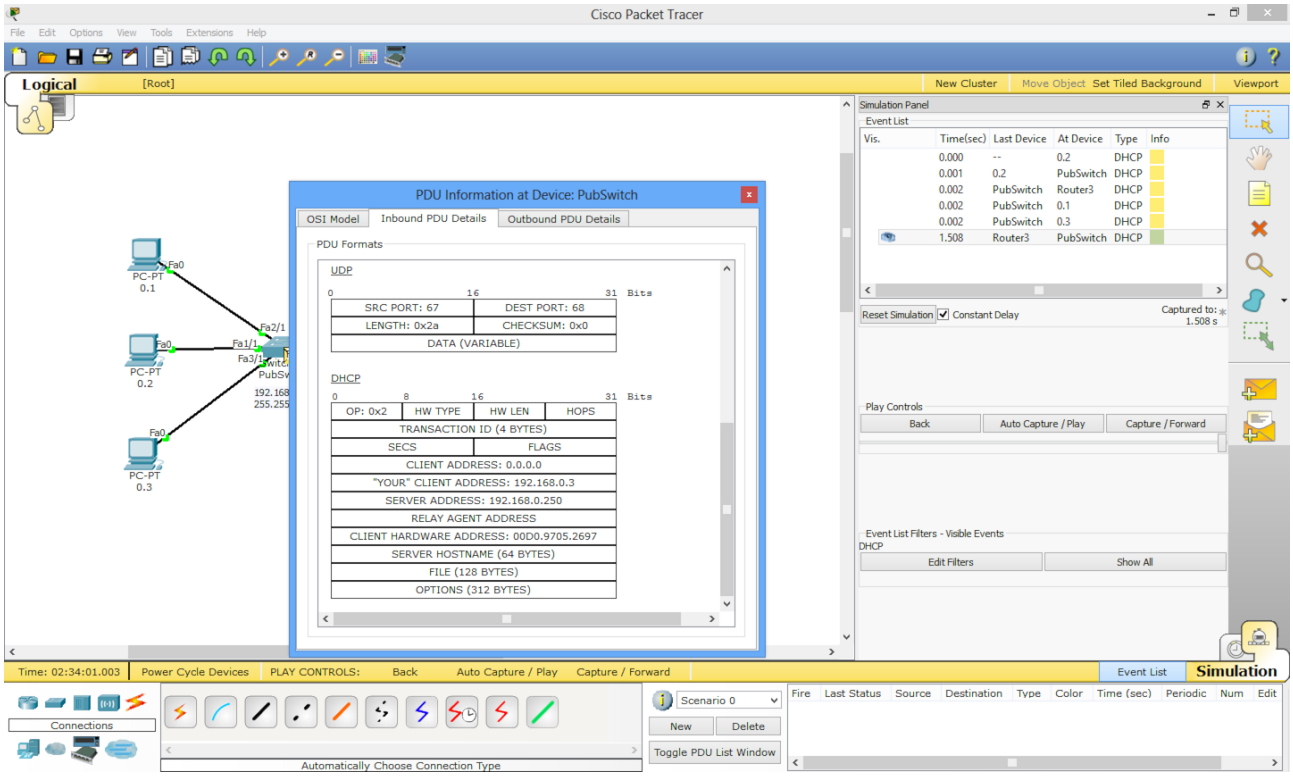


13. Stepping forward we'll then see the switch broadcasts this packet out all of it's ports as the packet is addressed.

14. The router then accepts the DHCP Discover request and responds by broadcasting the DHCP Offer request to the source workstation. Why does it broadcast this response?

    It broadcasts the response because the workstation does not have an IP address as yet.

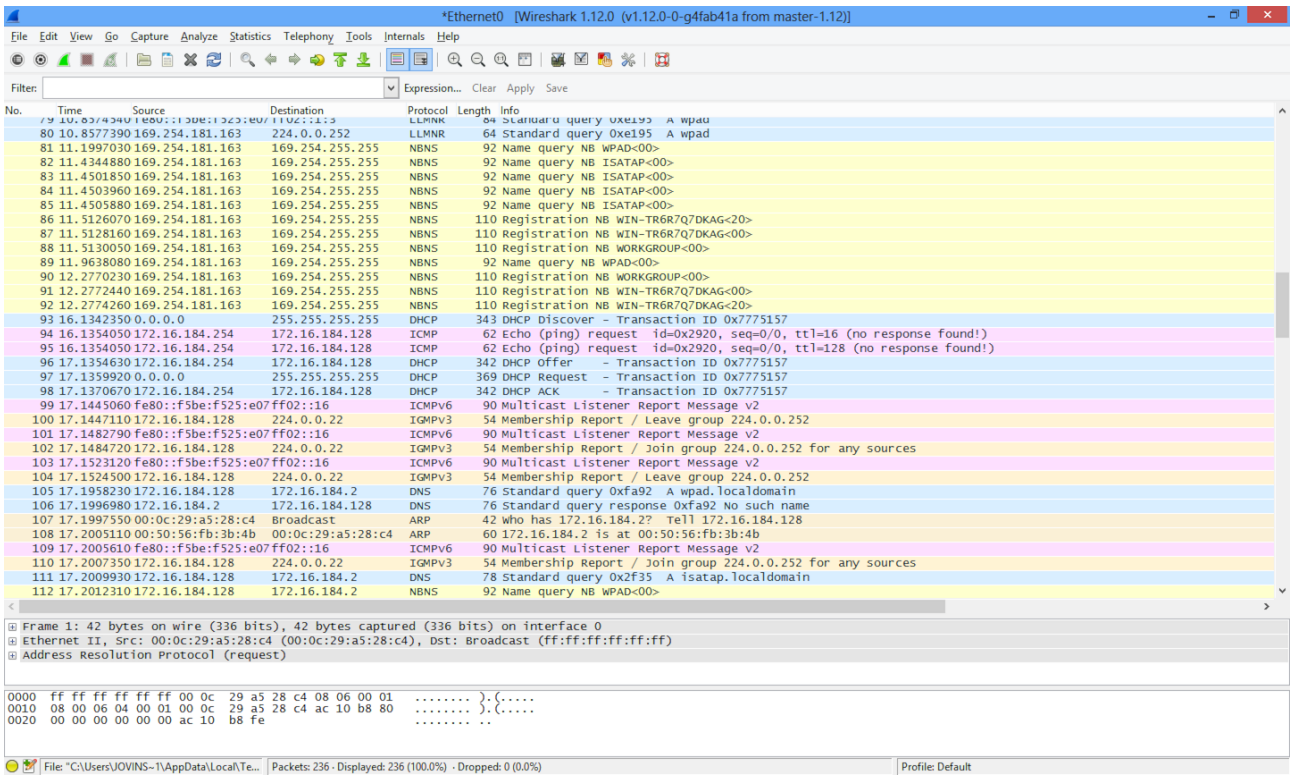
15. If we look at the data in the DHCP Offer request we can see the new logical address to be used by our workstation:



16. We can see the new IP address to be used by the workstation and it's new default gateway address ('Server Address'). The Workstation then responds to this DHCP Offer with a DHCP Request - officially requesting this IP address to be related to it's physical address.
17. Once the server receives the DHCP Request packet, it then responds with the final piece of communication - the DHCP Acknowledgement packet, and the workstation then applies it's new logical address to all outbound traffic.
18. In the next task we'll look at this on a real network using Wireshark.

Task 3 - Wireshark and DHCP, DNS and HTTP

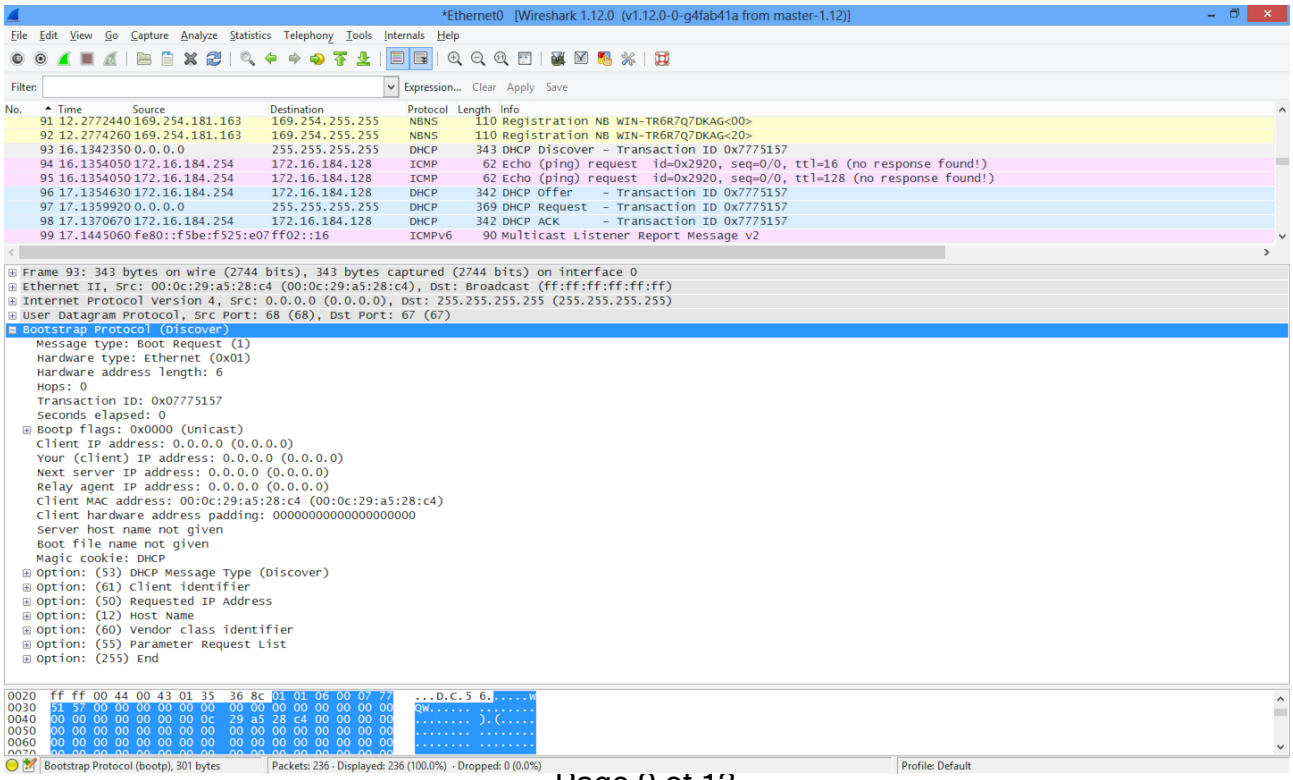
1. Last week we looked at ARP requests and TELNET in Wireshark. This week we will look over DHCP, DNS and HTTP, in that order.
2. First, open Wireshark and your Network Configuration panel in Windows or Mac.
3. Disable your network connection.
4. Start capturing packets with Wireshark and then go back to your Network Configuration and re-enable your network connection. Wait for network connectivity to return and then stop capturing in Wireshark. You should have caught DHCP traffic and you're list may look something like:



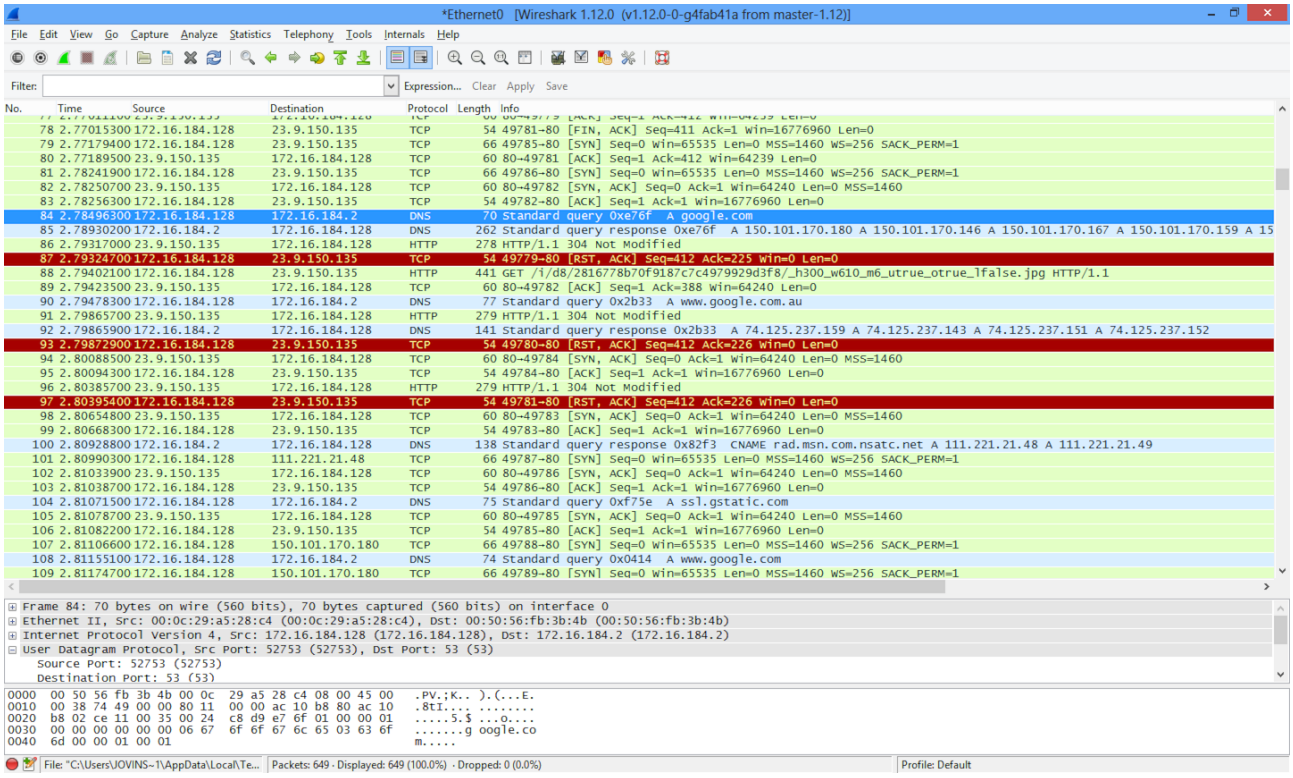
5. You may have seen a lot of packets captured here. Let's drill down and find our DHCP trace and follow the process of DHCP Discover, Offer, Request and Acknowledge. The key thing to look for is in the 'Info' column and should look like:

   `DHCP Discover - Transaction ID …`

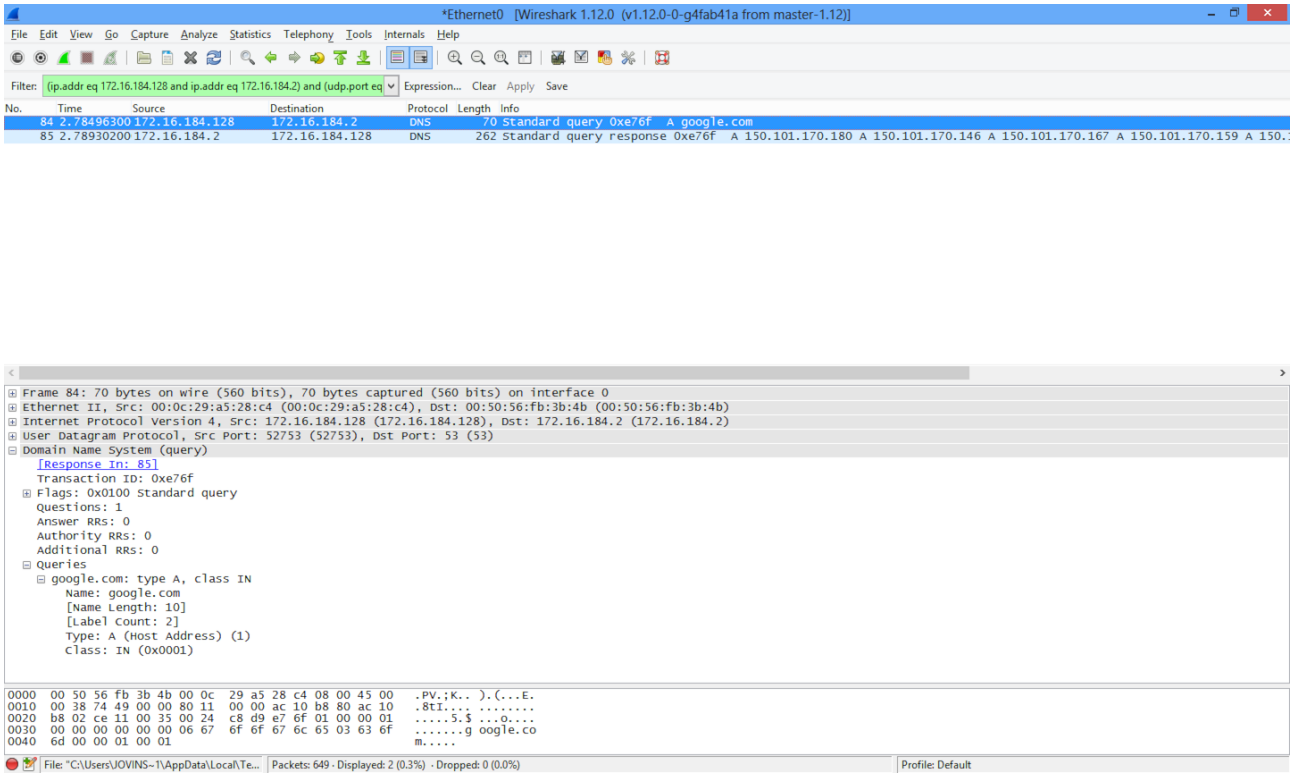
6. We should then see the other packets shortly after this one as DHCP Offer, DHCP Request and DHCP ACK. Let's have a look at each packet's structure. We can see all the same information as last week, such as what layer of information we're looking at, the source and destination MAC addresses and also the protocols in use. Let's look at the new part, we'll expand the layer `Bootstrap protocol (Discover)`.



7. Here we can see the DHCP communication slowly fill itself out as we progress through the four stages. At first we only have the following information, amongst other information:
   - `Client Identifier` - the MAC Address
   - `Requested IP Address` - if it has had an IP before, it may request the same one
   - `Host Name` - the name of the computer
8. Moving onto the DHCP Offer packet, we can see some new information appears, focussing only on the client relevant information:
   - `DHCP Server Identifier` - the IP address of the DHCP server
   - `IP Address Lease Time` - how long the IP address will be assigned before another DHCP communication will take place
   - `Subnet Mask` - the subnet mask of the network
   - `Domain Name` - the domain, also called 'workgroup'
   - `Router` - the default gateway of the network
   - `Domain Name Server` - the address of where website addresses can be translated into IP addresses
9. Now in the DHCP Request we can see that the client (workstation) confirms some of the information and relays back it's information such as it's identifier and it's own domain name.
10. Finally, in the DHCP ACK packet we can see that the server has made no modifications and the workstation understands that it now has it's assigned IP address for all communications.
11. Now, let's generate some DNS and HTTP traffic to capture. Start capturing in Wireshark and open your web browser and browse to www.google.com. Once it has fully loaded, stop capturing in Wireshark. You should have captured quite a bit of traffic:

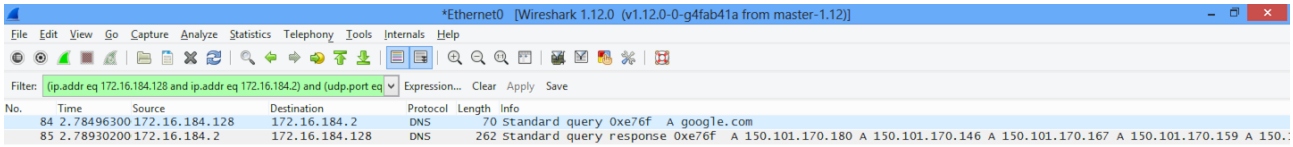


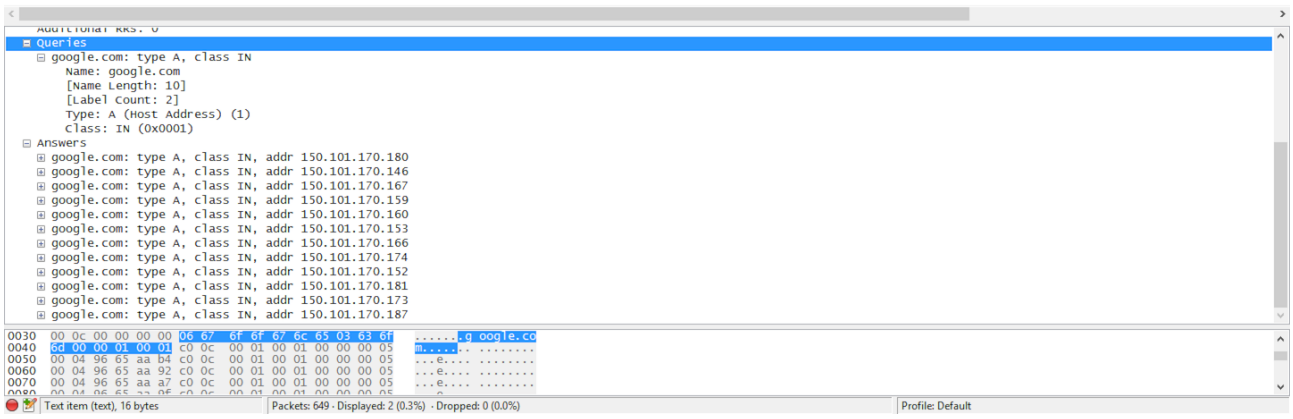
12. Scroll through your list and keep an eye out in the description for the website address, it may have been restructured slightly, as mine has to simply google.com (removed the www.)
13. We can narrow our list down to only the relevant DNS query by right clicking the captured packet and then selecting 'Follow UDP Stream', similar to last week when we analysed the TELNET communication. Now we should only see two packets, the query and the response, such as on the next page:



14. If we expand the final layer being '`Domain Name System (query)`' we can see the data involved for the translation request. Looking over this information we can see the following:
    - `Transaction ID` - this is a unique identifier which lets the DNS respond in a manner for the workstation to understand which query it relates to
    - `Queries` - the domains requesting to be translated
    - `type A` - we are requesting an IPv4 address for this domain
    - `class IN` - this is an internet request
15. Using this information we can find out what kind of DNS query it is we are making, whether it's a redirect, a mail server query or simply a TCP/IP query to enable website browsing, and allows the server to identify it as an internet request as opposed to a query about itself.

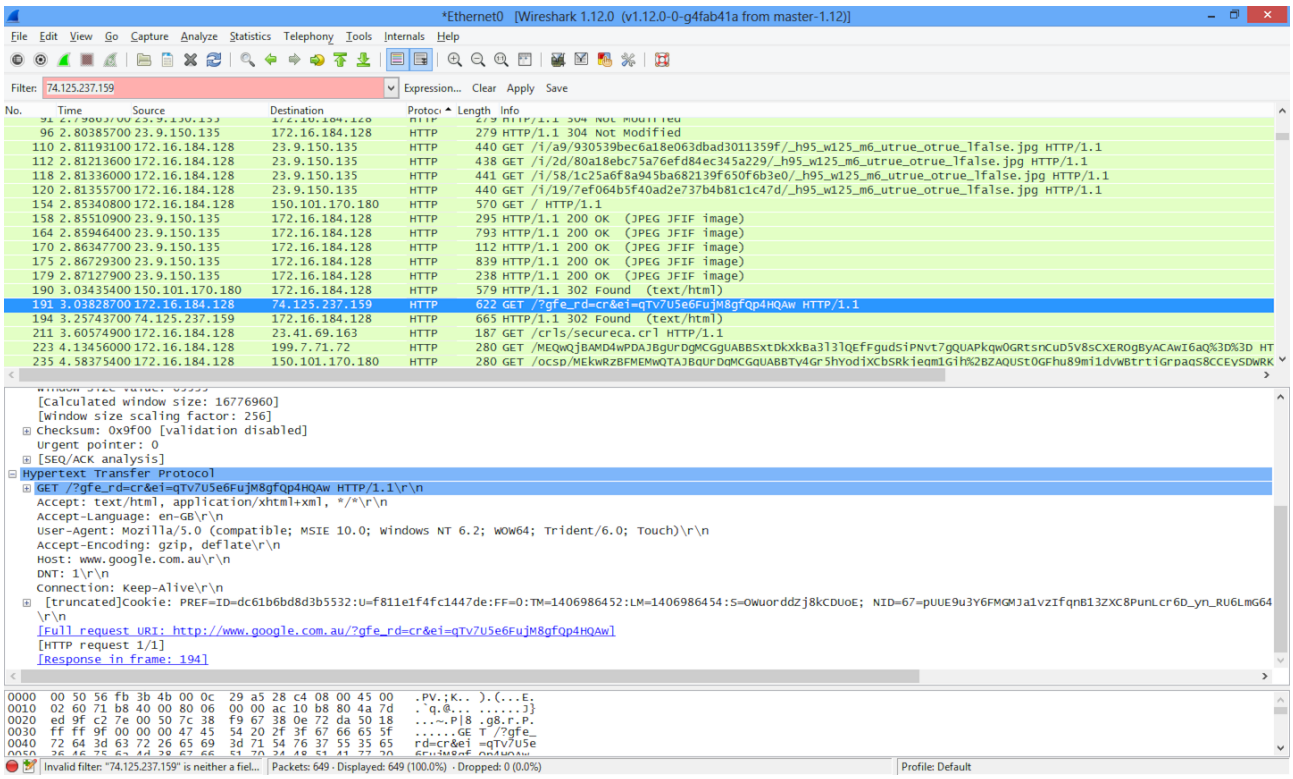
16. Now, let's look at the response:





17. The response contains all the same information as the query, including the unique Transaction ID along with it's set of answers for the question. We requested the IP address of google.com. It has provided us with all of Google's publicly accessible addresses so that should one of the links be down, we can still access the website.
18. Now we have the hard part of locating the HTTP information associated with this website request. The easiest way to achieve this is to organise the list by protocol type and look for HTTP, then look for the first IP address listed in the Answers section of our DNS query.

    Note: Some websites which have location based versions such as google.com will redirect you through to google.com.au. Make sure you follow all the DNS queries to ensure you get the final IP address - for Google this is 74.125.237.159.
19. Once we have found that we can begin to look at the data in the HTTP requests:



20. Here we can see the encoding, the identities of the browser and operating systems, whether it is a touch enabled display, what kind of connection we have and full addresses, including any cookies required. This is one way cookie sniffing can be done to steal someones session on a website and effectively steal their account - you won't have access to their username and password but you'll have their current session.
21. You can do this for any website and anything else - the key is to play around and be curious!