

Lab 2- c00225954 Packets

Question 2 – Capture a packet.

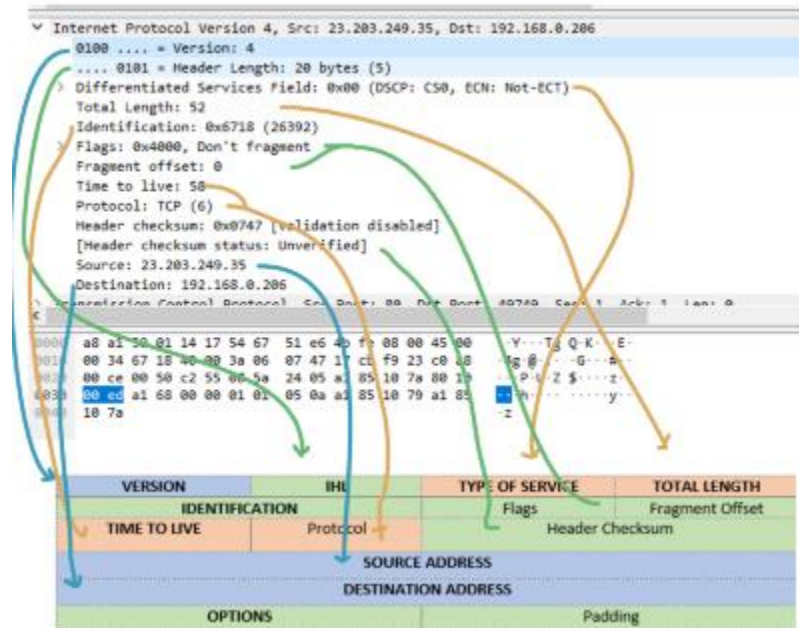
```
> Frame 1: 66 bytes on wire (528 bits), 66 bytes captured (528 bits) on interface \Device\NPF_{6D319FDF-...}
> Ethernet II, Src: CompalBr_e6:4b:fe (54:67:51:e6:4b:fe), Dst: ASRockIn_01:14:17 (a8:a1:59:01:14:17)
> Internet Protocol Version 4, Src: 23.203.249.35, Dst: 192.168.0.206
> Transmission Control Protocol, Src Port: 80, Dst Port: 49749, Seq: 1, Ack: 1, Len: 0
```

<																
0000	a8	a1	59	01	14	17	54	67	51	e6	4b	fe	08	00	45 00	..Y...Tg Q·K...E·
0010	00	34	67	18	40	00	3a	06	07	47	17	cb	f9	23	c0 a8	·4g·@:· ·G...#·
0020	00	ce	00	50	c2	55	08	5a	24	05	a1	85	10	7a	80 10	...P·U·Z \$...z·
0030	00	ed	a1	68	00	00	01	01	05	0a	a1	85	10	79	a1 85	...h... ..y·
0040	10	7a														·z

Question 3 – Draw an IP Header

VERSION	IHL	TYPE OF SERVICE	TOTAL LENGTH
IDENTIFICATION		Flags	Fragment Offset
TIME TO LIVE	Protocol	Header Checksum	
SOURCE ADDRESS			
DESTINATION ADDRESS			
OPTIONS		Padding	

Question 4 - Explain the fields for a particular IP packet captured. Try to explain the purpose of each.



Version – Version of the current IP

IHL- header Length : The length of the header in 32-bit words. The one captured is the minimum length possible at 20 bytes (maximum is 60 bytes)

Type Of Service: Specifies how to handle the datagram.

Total Length. Header length + data. Total length of the packet being sent/received.

Identification: Used for differentiating fragmented packets between datagrams.

Flags: control/identify fragments.

Fragment Offset: If the packet is too large it will need to be fragmented to go through, (Max size for a packet is 65,535 bytes). The offset is used to fragment large packets and to reassemble them.

Time To Live: if a packet doesn't make it to the destination before time to live runs out it will be discarded.

Protocol: defines protocol in data portion of IP datagram.

Header Checksum: If the router calculates a different checksum than the one specified in this field the packet will be discarded. This is used for error checking.

Source Address: The host IP (system that sent the packet)

Destination Address: The receiver's IP of above packet.

Options – For testing, debugging security etc.

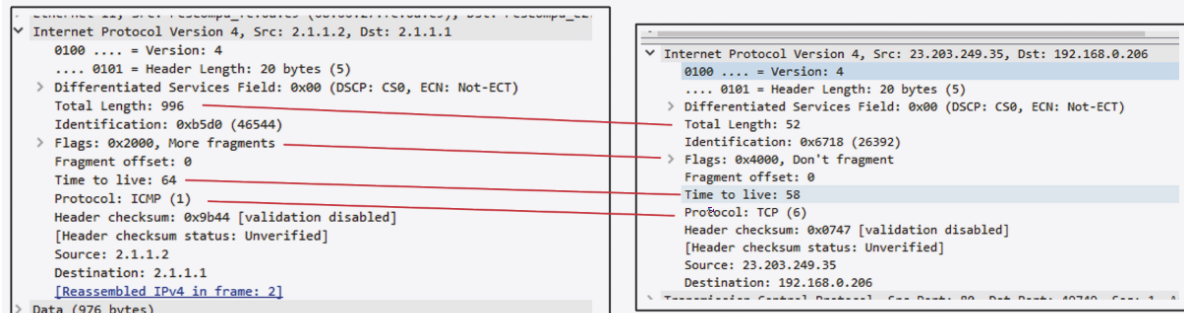
References: <https://erg.abdn.ac.uk/users/gorry/eg3567/inet-pages/ip-packet.html>

<https://blog.apnic.net/2018/06/18/a-closer-look-at-ip-headers/>

<https://study-ccna.com/ip-header/>

and more from google

Question 4



Example

My cap

Apart from different sizes and fragmentation occurring in the packet I captured versus the example the most significant difference is the protocol that is used.

ICMP is used in the example packet you provided. It is used as a control protocol and is not meant to carry application data. It carries information about the network and other protocols used, it pings remote hosts for responsiveness and times the probe messages.

TCP- Is a transportation protocol used for passing actual data from applications. TCP is a connection-oriented protocol and guarantees sent packets will reach the destination in the correct order.

Question 6 List three games you like and list their technical/design highlights.

Monster Hunter World : Recently the monster hunter series released it's first console based game in many years and a PC port, most games in the past have been on handheld systems and with the new release the development team made a new engine to make use of the stronger hardware. Sadly there isn't much details on the actual technical side of things however one change that occurred was more dynamic environments to fight monsters in, including the dreaded slopes. They had to tackle the change of inclination in the ground the rather large monsters appear in, make sure their parts don't cut into the ground and can be placed on a surface that isn't flat. They did this by using what I'll call an "anchor point" in several places around the monster's models that collide with the ground and act as the foundation for where each limb/tail tip rests. This is the main reason snake like monsters aren't in the game because of the amount of points that would be needed where as for the monsters in the game now it is usually 2/4 points (+ a few for when the monster falls over on it's side). I never thought much about how to make sure a model doesn't clip into non flat ground until I saw this pop up.

Another thing with monster hunter is each monster has it's own AI, and with the use of modding you can actually see which part of their behaviour tree they hop too when using different attacks/ are just wandering the environments.

Minecraft: Procedural Generation

Minecraft has an infinitely generated world and it's done so by using perlin noise to generation height maps, it generates one using the seed for a world and based on that seed everything about a world is generated, placement of trees, lakes, biomes etc. But what is just as interesting is what is the same for every Minecraft world, the very bottom layer of the world known as bedrock, which can't be broken isn't randomly generated for every world it is the same, some parts have 3 "blocks of thickness" and some parts are only 1. Grass and flower placement is also the same for every world, and by that I mean the offset from the center of the block they are placed on. People have managed to reverse engineer seeds by just matching the offsets with certain x/y/z co ordinates and finding the "chunk seeds". (<https://www.youtube.com/watch?v=ea6py9q46QU>) if you are interested.

The binding of Isaac: rebirth (and the flash version)

The binding of Isaac is a rougelike and while it also has procedural generation I've already touched on that, it does do it in a different way, it generates a path and selects fitting premade rooms (from thousands) and places them. Im more interested in the "synergies" between the power ups in the game.

The game came out as a flash game and some players noticed some power ups had cool interactions with one another. You might get a power up that lets you shoot straight lasers and then later pick up a power up that changes your shots into homing shots, with the two combined you get a homing laser, this is what I think pulled the game into the mainstream at the time, along with the insane themes and replayability that was made as a result of hundreds of power ups and synergies. But the flash version had a lot of performance issues and a lot of the original ideas had to be scrapped, the game got remade in a new engine and the developer managed to hardcode hundreds of new synergies, new items and new mechanics. And the fact that a change in engine made it possible is cool to me.