Peter-Jan Derks 11065540

1.1
1 Almere, The Netherlands
2 Paris, France
3 Dallas, Texas

1.2
An IP address is a way of uniquely identifying a device or server on internet. (Uniquely is not exactly true because of Private IP addresses.)

The difference between IPv4 and IPv6 is that in IPv4 an IP address is a 32-bit number while in IPv6 an IP address is 128-bit number.

IPv6 is necessary because IPv4 there weren't enough available IP addresses left. IPv6 has more addresses available because the number consist of more bits, so more there are more possible combinations.

1.3
The reason for this is that electronic devices connected to the internet are not all directly connected to each other. They are connected to each other via networks. The smallest networks are LANs, a local area network. If "LANs" are connected then a WAN is formed, a wide area network. All WANs together form the inter network, called the internet.

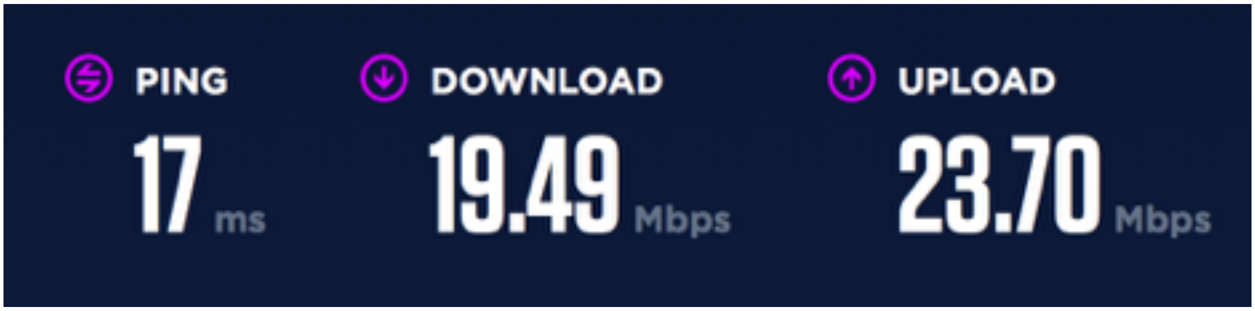
This means that not every router knows how to get to every IP address on the internet. Each router has a routing table, that tells them approximately which direction to go.

2.1
A public IP address is unique over the entire internet, while a private IP address is just unique over a small network. A computer on another network cannot simply use my private IP address to communicate with me Common prefixes are:
10.*.*.* , 172.16.*.* , 192.168.*.*

2.2
My computer then does not have a public IP address but a private IP address. It goes back to the routers public IP address with the source port of the computer as an additional identifier. The source port was

added by the router when the request was made.

2.3
When a computer connects to a network as a client it broadcasts to everyone: “I would like to join”. The DHCP server then responds only to the computer with: “I can help”. (A DHCP server is just a machine that gives IP addresses.) The client then requests an IP address. The DHCP server then sets aside an IP address and tells the computer what it is. Then the client confirms and uses the IP address.



PING 17 ms

DOWNLOAD 19.49 Mbps

UPLOAD 23.70 Mbps

ISP reports 25 Mbps

4.1
A hierarchical process takes place when we map a domain to a IP address. The root DNS servers are the “authoritative source of information” and is were all the information is originally stored. These servers don’t contain a list of domains but contain a list of servers that lead to a list of domains.
This means the that the root DNS servers are at the top of the hierarchical process. So if the root DNS server is compromised we could not go through the first step of the process of mapping a domain to an IP, so we wouldn’t know which domain corresponds to which IP address if a DNS server was ever compromised. This is why it is important they are kept secure.

4.2
A record maps a domain to an IP address. A CNAME record maps a domain to another domain. You might want to use an CNAME record to make it easier for someone to remember the name of a domain and faster to type in. For example another hostname for ghs.google.com is “mail”.

4.4

No .PJD is not a recognized TLD.

5.1
My opinion is that an ISP should not be able to decide which parts of the internet you can and can't access. If ISPs do this they suddenly become more than just a provider, the ISPs choices could make a big influence to modern politics, economics and how we see the world. However I do think ISPs should be aloud to block access to sites involved in illegal practices

5.2
API limits outside program access to specific features of the code. This has two advantages for Google or Facebook. The first one being that if another app or program would like to make use of one of the functions from Facebook or Google it doesn't have to comb through the whole code. The second advantage is that Facebook and Google can still share a bit of there code without having to share things they don't want to share.

6. Putting it All Together

1. You just typed http://youtu.be/C_S5cXbXe-4 into your web browser's address bar and hit Enter, at which point a YouTube video appears on a page. In a technically detailed paragraph, what happened? You can assume that you are indeed connected to the Internet and that URL points to a valid page on YouTube (which it does, so you should click it). Your response should reference terms like routers and DNS!

When you hit enter your browser translates your request using a certain protocol to communicate with your router. The domain you typed in has to be translated to an IP address. This happens thanks to a protocol called DNS. Once it's translated thanks to your ISP it connects to the world wide web and you will reach the server. The server then responds and sends the YouTube page back.

1.1
All these technologies are protocols which is a formal set of rules for communicating. Thanks to protocols different pieces of hardware can communicate, for example a client can make a request and a server can give a response.

1.2
A web browser is a program that kind handles the process of sending a request to a servers and handling the response that the server sends back. The information about which web browser you're using might be helpful because there is not a bug on the website itself but on how the browser handles this website.

2.1

```
GET: /home.php HTTP/1.1
Host: www.catgifpage.com
```

GET: Verb that describes what we want from server, in this situation we're "getting" some information from a server.

/home.php: Path commands to get some information from /home.php but doesn't say were it's located

HTTP/1.1: specifies the version of http that's being used.

Host: www.catgifpage.com: Header, host is the key and on the right of the double colon is what we associate with this key, the value.

2.2

```
POST / HTTP/1.1
Host: raspberrycats.wordpress.com
Accept: */*
Email: unicodelovehotel@gmail.com
Content-Length: 0
Content-Type: application/x-www-form-urlencoded
```

POST / HTTP/1.1

Post is the verb/method, no path is given and I'm using version 1.1

Host: raspberrycats.wordpress.com

Tells which website I'm sending my request to.

Email: unicodelovehotel@gmail.com

Email is the header, and the email address is the value I want associated to it.

2.3

500 Internal Server Error is the HTTP status code. This code tells us if our request was successful or not and why not. In this case the server has somehow messed up the request and there isn't a mistake in the request.

2.4

301 Moved Permanently is the HTTP status code. This code tells us if our request was successful or not and why not. In this case it means that what page were looking for is not found on that domain, but the server knows on which domain it is found. The domain on which it is found is given in the last row: "Location: http://cse1.net"

3.1

When you send an email it first goes to a gmail SMTP (simple mail transfer protocol) server. This server is going to follow the SMTP protocol to communicate with you and the server of the email address you are sending to, in this case an example SMTP server. The example SMTP server will communicate with the someone@example.com's inbox

3.2
The difference between a queue and a stack is that in a queue the first thing that joins the waiting list is also the first thing to be sent, while on a stack the last thing to join a waiting list is the first thing to be sent. A queue is much more appropriate than a stack for emails that are being sent. If they would be sent as a stack than the first email in the stack might not be sent for a long time.

3.3
An email is sent in multiple parts and each part needs to contain the Received header.

3.4
POP is one directional, IMAP is two directional. This means that for example IMAP will remember if you have read an email. The commands are also different because they're different protocols.

3.5
A phishing attack is when you pretend that you're sending someone an email from an email address you can't access. Hackers do this to find out someones personal information. It is actually possible to send an email with a header that is not your email address. This means you've sent it from your own address but the email says it's been sent from another address.

4. Mission Control

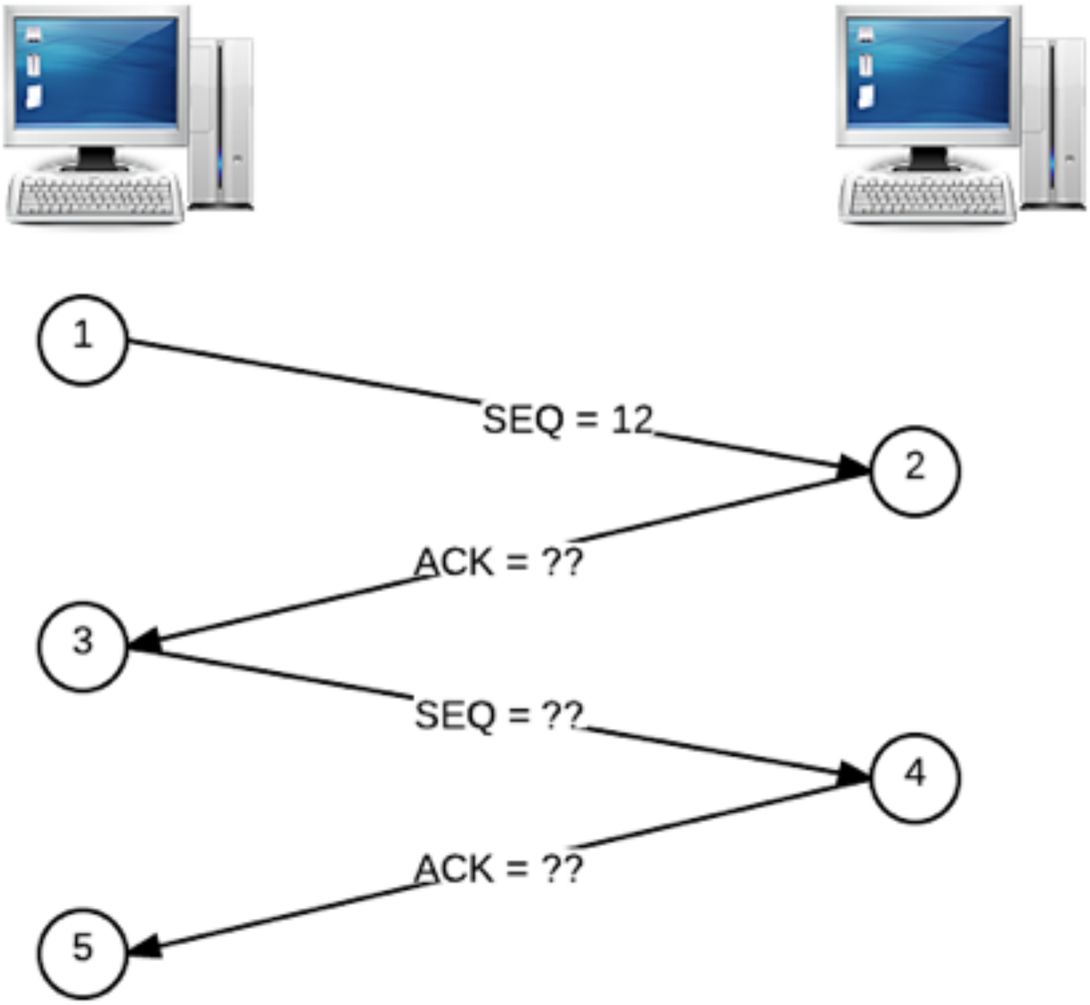
1. When we say that TCP ensures reliable data transfer, what two guarantees are we making about the delivery of segments? What's an ACK, and how does it relate to these guarantees? Similarly, what is a sequence number, and how does it relate to these guarantees?

Transmission control protocol, gonna establish a connection between to devices. Only once it knows it has a connection it will start sending data.
reliable data transfer

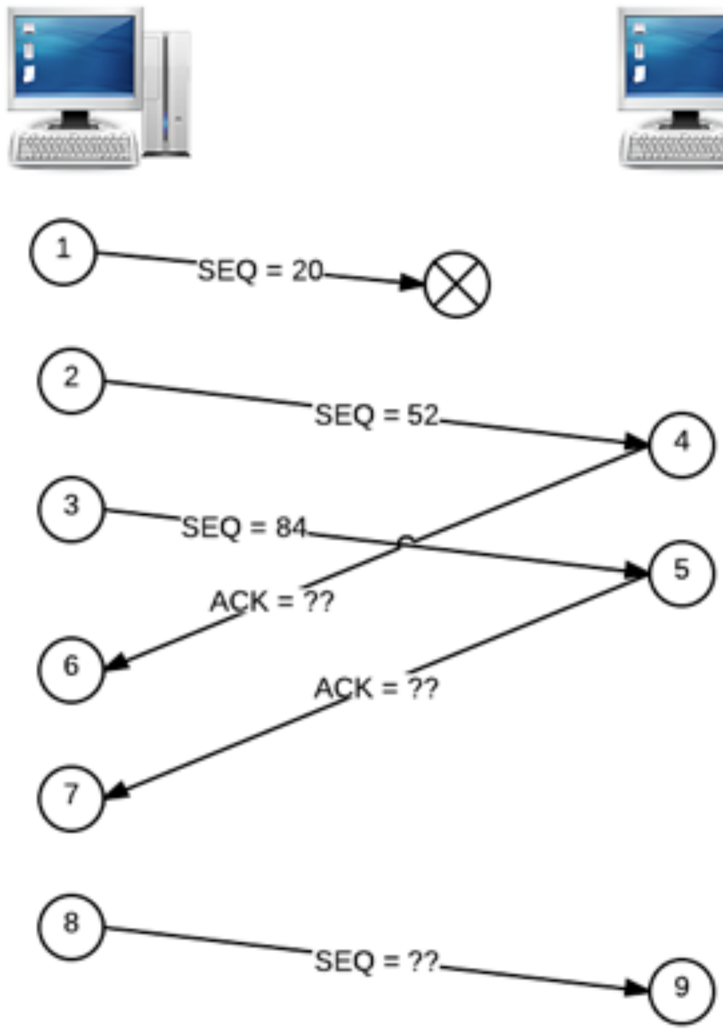
I'm going to try my best

The first guarantee we make is that the server knows when it's message has been read. The server knows this thanks to ACK.

Every time a segment is sent from point A to point B, point B is actually going to respond with a message called an ACK that it got it.



ACK 2 to 3 = 44 (32+12)
SEQ 3 to 4 = 44 (equal to ACK 2 to 3)
ACK 4 to 5 = 76 (44+32)



ACK 4 to 6 = 84 (32 +52)

ACK 5 to 7 = 116 (32 + 84)

SEQ 8 to 9 = 20 (the segment that is lost in SEQ 1)