Networked Systems (H)

Laboratory exercise 3: Report

Gabriela Georgieva, matr #: 2130120g

Section I: IP addresses

When applying a DNS lookup on different websites, one can discover that some have more than one IP address. This simply means that the website is hosted on multiple servers in order to achieve better load distribution.

The most common technique that is used is Round-Robin DNS (implemented by most of the websites I have chosen), which responds to DNS requests with one IP address chosen from a list of potential addresses corresponding to several servers hosting identical services. The IP addresses are distributed to the different users in order by permuting the list of addresses – on the principle of round-robin, and the geographic location of the visitor is not taken into account. This ensures that the traffic is distributed evenly among the multiple servers, which leads to good load balancing. This can be proved by the fact that different IP addresses are output for the same website when executing the DNS lookup multiple times.

Some websites also use fault-proof techniques, ensuring that if one of the servers is down, the visitor will be redirected to another. This can be seen when performing a DNS lookup as some websites return multiple IP addresses at the same execution of the program. Again, on the principle of round-robin, visitors might be redirected through the addresses in turn, until a functional one is reached (in the case that one or more servers are down).

Another thing that could be discovered is that some websites also have an IPv6 address. For instance, 3 out of 10 of the websites chosen for the purpose of this exercise, already provide a server located at an IPv6 address.

Section II: Router-level Topology Maps

The router-level topology maps presented, create a visual representation of the routes taken by packets across the IP networks, showing sequences of addresses used for each hop when tracerouting the IP addresses of the servers of websites.

The longest path on the map represents the route with most hops, i.e. the server on the network whose location is the farthest from the origin. On the IPv4 route-level topology map the longest path is a partial path (traceroute blocked by a firewall) to a website called blogilates.com while on the IPv6 map the longest path is the path to the IP address of a server of Facebook.

In both maps there exists a single unique route/path from the root to each destination (leaf of the tree), because all websites used have only been tracerouted once. However, as this is a map and not a tree, the root (the IP address of the machine the map was created from) could also be a possible destination which could be reached from all leaves, because packets can travel in both directions. This means that multiple routes exist to the same destination, but they are not disjoint as each IP address is only represented once in the map.

Looking at the prefixes of the IP addresses (or even at the digits before the first dot in the prefix for IPv4 addresses) in the maps, one can infer where geographic and organisational boundaries lie. Starting from the IP address at the top, which represents the location where traceroute was executed from, routers lead to IP addresses with different prefixes. UK websites are used which means that it must be the case that even before the first branching occurs, the routers are located in the same country, inferring that the difference between their prefixes comes from the fact that they are served by different ISPs. Looking at the branches, some more noticeable differences can be discovered which implies that not only are the routers served by different ISPs but their geographic locations are very far from each other (most probably in different countries).

Section III: IPv4 and IPv6

The IPv4 and IPv6 router-level topology maps have very similar structures but do not match perfectly as only 3 out of the 10 websites used for their generations have IPv6 addresses. Their structures should be very similar because they represent the same paths between the locations. However, if only websites that have both IPv4 and IPv6 addresses were used to generate the maps, very similar structures will be encountered, but it is not absolutely necessary that the routers found in each map will be the same ones, as traceroute might show different paths with each execution.

Section IV: The traceroute Tool

Traceroute is a computer network diagnostic tool used to find the entire path that a packet travels through from one location to another, to name and identify the routers and devices in the path and to find the time taken to send and receive data from one device to another (a.k.a. network latency).

Traceroute achieves all this by sending a sequence of packets addressed to the destination host. In Linux, the default type is UDP packets, but TCP- and ICMP-based implementations also exist. Each of the packets sent has a TTL field, standing for “Time To Live”, which indicates the maximum number of hops that the packet can travel through across the Internet, before it is discarded. If the destination is not found before the TTL limit is reached, the last router that received the packet will drop it and will inform the original sender. All routers the packet travels through ensure that the TTL is consistent by decreasing it with 1 before passing it for the next hop. The router that changes the TTL to 0 is the one that discards it (if it is not the destination) and informs the original sender that the TTL value has exceeded by sending an “ICMP TTL exceeded” message. Traceroute makes use of this functionality by assigning sequential values to the packets it sends, starting from 1. This forces all routers to decrement it to 0 one at a time and send “ICMP TTL exceeded” messages, where their own IP address as a sender is stored. This is how traceroute finds the IP addresses of the routers on the path to a specific destination.

This is exactly what happens during the traceroute:

1: The original sender passes a UDP packet with a TTL value set to 1 which ensures that the first router on the path to the destination will set it back to 0, discard it and send a “ICMP TTL exceeded” message;

2: The original sender passes another UDP packet with a TTL value set to 2, which ensures that it will pass through the first router and will be discarded by the second one on the path – it will be the one to set the TTL to 0 and send the message;

…

n: The original sender will pass as many UDP packets as necessary with TTL values increased by 1 sequentially until one packet reaches its real destination before its TTL value has been set to 0. The destination router will send a different message: a “ICMP Destination/PORT Unreachable” message, which will indicate that the packet has been received and the original sender can cease sending further packets. (Note: All packets are being sent to an invalid port address at the remote host, as traceroute does not aim to send any real data to the destination.)

All this would not be possible without the Internet Control Message Protocol (ICMP), which represents an error-reporting protocol that can be used by network devices to generate error message and send them to the source IP address when network problems prevent the delivery of IP packets. Traceroute depends on ICMP for all returning messages that routers send back to the original sender. Depending on the type of the message, traceroute determines whether the sender is a router on the path to the destination or the actual destination itself and acquires data about the IP address of the routers as real messages are being sent to report the errors.