Introduction to Networks & their Applications - HW2

Samuel Barth
3/6/2017

1. homepage.divms.uiowa.edu/~sbrth/

2. (I) Tyically, Web caches are installed by ISPs or by network mannagers such that there exists a high-speed connection between clients on the network and the web cache. Thus, the bottleneck bandwidth between the client and the origin server is often much greater than the bottleneck bandwidth between the client and the web cache, decreasing the response time of HTTP requests. (Kurose, Ross 111)

Web caches also indirectly improve the performance of core networks and links by decreasing overall traffic on the internet. (Kurose, Ross 112)

(II) When a client requests a web document at some URL, the application creates a DNS query message with the host-name in question, and the transport layer creates a <u>UPD</u> segment with a destination port 53. The DNS reply contains the IP address associated with the hostname in question. Now, the application can initiate a TCP connection to the HTTP server process. (Kurose, Ross 127, 503). In this scenario, DNS protocol (application layer) and UDP (transport layer) are needed in addition to HTTP protocol.

(III.) (a) In the non-persistent regime of no parallel connections, where the initiation of a TCP connection requires 1 RTT, we see that a web page with 8 objects can be loaded in the following time:

$$\underbrace{\left[8 \times (1\ \text{RTT})\right]}_{\text{establish TCP connection}} + \underbrace{\left[8 \times (1\ \text{RTT} + \text{time to transmit at server})\right]}_{\text{complete 3-way handshake and request single object}}$$

(b.) In the non-persistent regime that is configured for 5 parallel connections:

$$\underbrace{\Big[(\underset{\substack{\downarrow \\ \text{establish connection}}}{2 \times 1\ \text{RTT}}) + (\underset{\substack{\downarrow \\ \text{complete 3-way handshake and} \\ \text{send first object}}}{\text{time to transmit at server}})\Big]}_{\substack{\text{can't utilize parallel connections yet, because} \\ \text{browser doesn't know how many objects to request yet.}}} + \underbrace{\Big[(2 \times 1\ \text{RTT}) + (5 \times \text{time to transmit at server})\Big]}_{\text{can now request 5 objects simultaneous}}$$

$$+ \underbrace{\Big[(2 \times 1\ \text{RTT}) + (2 \times \text{time to transmit @ server})\Big]}_{\text{the last 2 objects can be requested in parallel}}$$

(c.) In the persistent HTTP regime, we can use pipelining to our advantage, requesting objects from the server in a back-to-back manner on a single TCP connection (i.e. initializing the connection only once). So the total time to load the 8 objects is given by:

(1 RTT) + (8 × time to transmit file request on client side) + (1 RTT) + (8 × time to transmit object on server side)

(Kurose, Ross 98-103)

(II) (a.) day time:

$$\frac{500\text{ Kb}}{1\text{ request}} \cdot \frac{0.5 \times \cancel{1000}\text{ requests}}{1\text{ second}} \cdot \frac{1\text{ sec}}{700\text{ Mb}} \cdot \frac{1\text{ Mb}}{\cancel{1000}\text{ Kb}} \cong 0.357$$

$\cong 35.7\%$

night time:

$$\frac{500\text{ Kb}}{1\text{ request}} \cdot \frac{0.5 \times 600\text{ requests}}{1\text{ second}} \cdot \frac{1\text{ sec}}{700\text{ Mb}} \cdot \frac{1\text{ Mb}}{1000\text{ Kb}} \simeq 0.214$$

$\cong 21.4\%$

(b.) probability of cache <u>miss</u> is $(1-h)$, so requests per second is effectively only $(1-h)\cdot(0.5 \times 1000 \text{ requests/second})$. So, the consumption of the link's capacity is $\cong \underline{(1-h)\cdot 0.357}$ during the day.

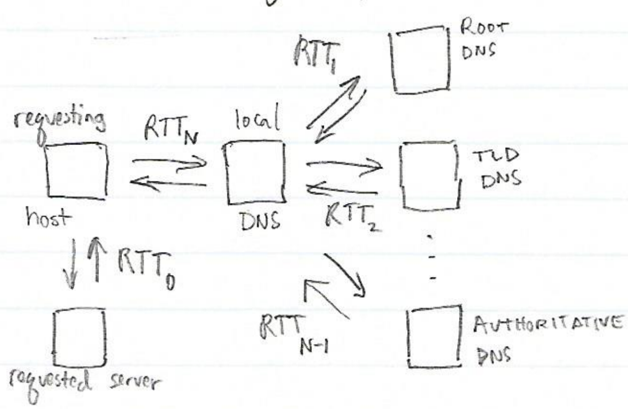
(3.) (I) The DNS is required to reconcile the preferences of humans (who prefer hostnames) and routers (who prefer IP addresses) (Kurose, Ross 127). That is to say, many requests for web objects are initiated by humans at the application layer in the Internet Protocol Stack. Because it is often quite inconvenient to remember / write out IP addresses for requested web pages, these requests often utilize hostnames that the DNS translates to IP addresses so that routers (who do not have an application layer) can find web servers. IP addresses are more convenient — it turns out — when your goal is to locate a host on the internet (Kurose, Ross 127)

(3)(II.) The DNS is operated in a distributed, hierarchical manner "in order to deal with the issue of scale" (Kurose, Ross 130). The distributed nature of DNS spreads out traffic, improving the performance of the internet. Additionally, as long as the DNS service is distributed, a single DNS crash won't bring down the internet. Distributed DNS serves also ensure that users are close to at least one DNS server (Kurose, Ross 130).

(III.) I have included a copy of the result of the host-v uiowa.edu below.
Analysis: It looks as though the response is divided into three areas — or three Question-Answer sections. I hypothesize that these three sections correspond to the responses from the DNS servers in each of the levels of the DNS server hierarchy — Root DNS server, top-level domain server (.edu), and authoritative server. It looks like the response from the Root DNS server is on the bottom, the response from the top-level domain server is in the middle, and the authoritative server is on the top (we finally get an IP: 54.163.225.50.

IV. DNS queries can be directed at specific DNS servers using the nslookup program (Kurose, Ross 138). To ensure that all queries are made to a specific DNS server, however, we'll need to modify our IPv4 and IPv6 settings on our computer, providing the desired DNS server's IP address (source: https://developers.google.com/speed/public-dns/docs/using)

V We'll assume the following diagram represents the problem as it was described



In this case, the total time is then $(RTT_1 + \dots + RTT_N) + \underbrace{2(RTT_0)}_{\text{3-way handshake + delivery of the zero-transmission-time object.}}$

(Kurose, Ross 133)

```
 -v uiowa.edu
du"
 opcode: QUERY, status: NOERROR, id: 5492
 ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

TION:
                    IN      A

ON:
          25665     IN      A       54.163.225.50

es from 127.0.1.1#53 in 121 ms
du"
 opcode: QUERY, status: NOERROR, id: 49868
 ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 0

TION:
                    IN      AAAA

CTION:
          740       IN      SOA     dns0.uiowa.edu. hostmaster.uiowa.edu. 201610304 7200 3600 3600000 21600

es from 127.0.1.1#53 in 19 ms
du"
 opcode: QUERY, status: NOERROR, id: 1032
 ra; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 0

TION:
                    IN      MX

ON:
          52770     IN      MX      10 smtp2.its.uiowa.edu.
          52770     IN      MX      10 smtp1.its.uiowa.edu.
          52770     IN      MX      10 smtp3.its.uiowa.edu.
          52770     IN      MX      10 smtp4.its.uiowa.edu.

tes from 127.0.1.1#53 in 19 ms
a
ta' found, did you mean:
' from package 'pinta' (universe)
not found
a
nta' is currently not installed. You can install it by typing:
```