**Luke Pepin - CSE3300: Computer Networking**

Homework 3

Due Date: 10/1/2024, Tuesday. Submission through HuskyCT.

Full score: 100 for CSE3300 students; 120 for CSE5299 students (will be normalized to 100 when entering the grade in HuskyCT).

1. **HTTP (15 points).** Suppose within your web browser you click on a link to obtain a web page. The IP address of the associated URL is already cached in your local host, so no DNS look-up is necessary to obtain the IP address. Further suppose that the web page associated with the link contains exactly one object, consisting of a small amount of HTML text (i.e., the URL only has a base HTML file, with no referenced/embedded objects). Suppose TCP is the underlying transport protocol. Let *T* denote the RTT between the local host and the server containing the object. Assuming zero transmission time of the object, how much time elapses from when the you click on the link until the client receives the object?

Total time = TCP establishment + HTML response

TCP establishment = 1T

HTML response = 1T

Total Time = 1T + 1T = 2T

1. **Persistent and non-persistent HTTP (40 points).** In the previous problem, suppose the base HTML references 9 very small objects on the same server. Neglect transmission times. Determine in each case below, (1) how much time elapses from when the client clicks on the link until the client receives the web page (i.e., all the objects, including the base and all referenced objects)? (2) how many TCP connections are opened?
   * 1. non-persistent HTTP with no parallel TCP connections?

Total Time = Total TCP Connections ( TCP establishment + HTML response)T

\*Note Uses TCP & HTML Definitions above (1.5 & 1)

Total TCP Connections = HTML index + HTML refererences = 1+9 = 10

Total Time = 10(1 + 1)T = 20T

* + 1. non-persistent HTTP with three parallel TCP connections?

Total Time = Total TCP Connections ( TCP establishment + HTML response)T

Total TCP Connections = Inital HTML + (references/parallel) = 1 + (9/3) = 4

\*The Total TCP Connections are still 1+9= 10, parallel connections does not change the number of connections only the time it takes with overlapping processing.

Total Time = 4(1 + 1)T = 8T

* + 1. persistent HTTP without pipelining?

Total Time = Total TCP Connections ( TCP establishment + HTML response + HTML reference \* number of HTML references)T

Total TCP Connections = 1

HTML reference = GET HTML reference, HTML response = 2(0.5) = 1

Total Time = 1(1+1+(9\*1))T = 1 + 1 + 9 = 11T

* + 1. persistent HTTP with pipelining?

Total Time = Total TCP Connections ( TCP establishment + HTML response + HTML reference \* number of HTML references)T

\*Pipelining reduces the time for number of HTML references to 1

Total TCP Connections = 1

Total time = 1(1+1+(1\*1))T = 1 + 1 + 1 = 3T

1. **DNS and HTTP (15 points).** Suppose within your web browser you click on a link to obtain a web page. The IP address of the associated URL is not cached in your local host, so a DNS look-up is necessary to obtain the IP address. Suppose that *n* DNS servers are visited before your host eventually receives the IP address from DNS; the successive visits incur an RTT of *T*1*,T*2*,...Tn*. Further suppose that the web page associated with the link contains exactly one object, consisting of a small amount of HTML text (i.e., the URL only has a base HTML file, with no embedded objects). Let *T*0 denote the RTT between the local host and the server containing the object. Assuming zero transmission time of the object, how much time elapses from when the client clicks on the link until the client receives the object?

Total time = DNS lookup + TCP establishment + HTML response

DNS lookup = represent the sum of *T*1*,T*2*,...Tn*  =

TCP establishment = 1.5 *T*0

HTML response = 1 *T*0

Total time = + (1.5 + 1) *T*0 = **+ 2*T*0**

\*Note this is my first time using Word equations so they look a bit strange

1. **Web proxy (30 points).** Fig. 1 shows an institutional network that is connected to the Internet. Specifically, there is a router (*R*1) in the institution that is connected to an access link, which is connected to a router in the Internet (*R*2). Suppose that the average object size is 900*,*000 bits and that the average request rate from the institution’s browsers to the

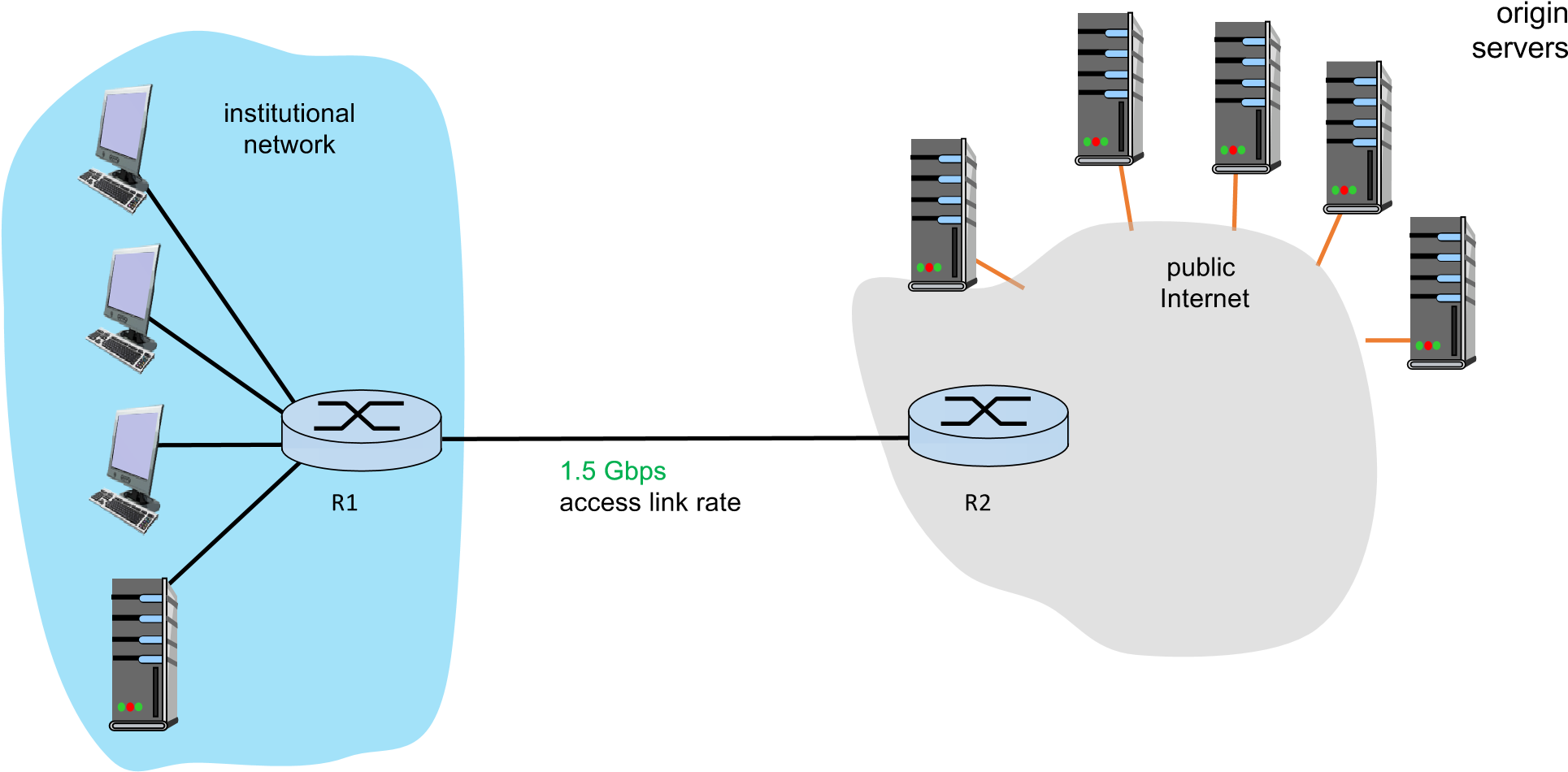


Figure 1: Illustration of an institutional network that is connected to the external Internet through an access link.

original servers is 1*.*66 × 103 requests per second. The access link bandwidth is 1*.*5 Gbps. Also suppose that the amount of time it takes from when *R*2 forwards an HTTP request until it receives the response is 100 ms on average (i.e., the average Internet delay, denoted as *d*, is 100 ms). We model the total average response time as the sum of the average access delay (that is, the delay from the Internet router *R*2 to the institution router *R*1) and the average Internet delay (i.e., *d* = 100 ms); all the other types of delays (e.g., the delays in the LAN) are ignored in this problem. For the average access delay, we model it as ∆*/*(1 − ∆*β*), where ∆ is the average time required to send an object over the access link and *β* is the arrival rate of objects to the access link.

* + 1. (10 points) Calculate the total average response time.

Total average response time = Average access delay + Average internet delay

Average internet delay (given) = 100ms

Average access delay = ∆*/*(1 − ∆*β*) = Average time to send an object over link/(1- change in arrival rate of objects to the access link)

Average time to send an object over link = ∆ =Object size/ access link bandwidth =

900,000 bits / 1.5 Gbps = 900,000 /(1.5 \* 10^9) = 0.0006 = 0.6 ms

Arrival rate of objects to the access link = *β =* Object size \* average request rate =

900,000 \* 1.66 \* 10^3 = 1.494 \* 10^9

Average access delay = ∆*/*(1 − ∆*β*) = (0.0006/ (1 – ( (1.494 \* 10^9 )/(1.5\*10^9))) = 0.15 = 150ms

Total average response time = 150 ms + 100ms = 250ms

* + 1. (20 points) Now suppose a cache is installed in the institutional LAN. Suppose the hitrate is 0.4. Calculate the total response time.

Total response time = hit rate \* cache time + miss rate \* access delay

Cache time = 0ms (is installed)

Missrate = 1 – 0.4 = 0.6

Total response time = 0.4 \* 0ms + 0.6 \* 250ms = 150ms