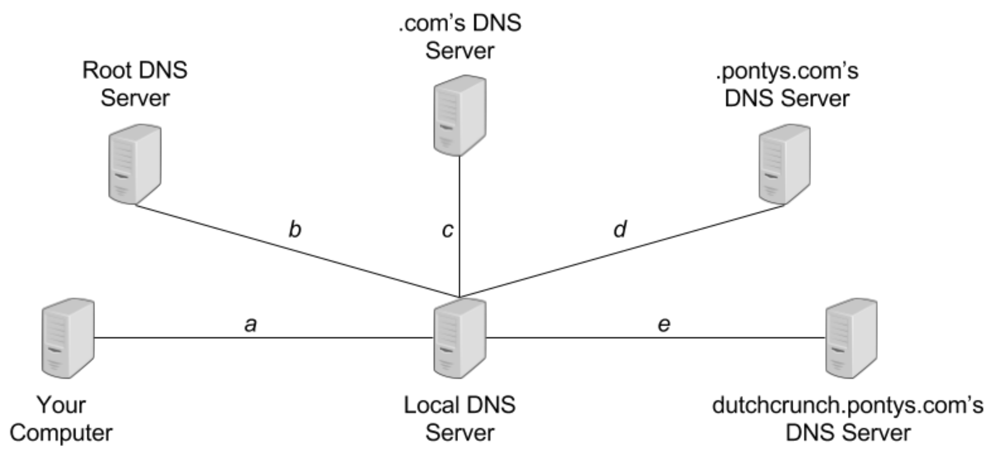
Discussion 10: DNS and Web

1 DNS

Pontague's Sandwiches, a deeply cherished sandwich shop in the Bay Area, is about to go out of business. However, being an adept Internet user with a particular eye for detail, you have realized that their website, www.dutchcrunch.pontys.com, is still operational and accepting online orders for the next T minutes. Consider the following setup of DNS servers, with annotated latencies between servers:



Assume that the latency between your computer and the Pontague's server is t, that once you send an order for a sandwich you must wait for a confirmation response from Pontague's before issuing another, and that your computer never stores any state about Pontague's IP address. In each of the following three scenarios below, determine how many sandwiches you can order before Pontague's closes for good:

(1) Your local DNS server doesn't cache any information.

(2) Your local DNS server caches responses, with a time-to-live $L \geq T$.

(3) Let $T = 600$ seconds and $a = b = c = d = e = t = 1$ second. Your local DNS server caches responses with a finite time-to-live of 30 seconds.

2 Performance

We want to download two files, both of size M. However, to download two files, we need to first download a webpage of size P. Assume the following:

- SYN, ACK, SYNACK, and HTTP request packets are small and take time z to reach their destination.
- Each of our HTTP connections can achieve throughput T for sending files and web pages across the network unless there are concurrent connections, in which case each connection's throughput is divided by the number of concurrent connections.
- You never need to wait for TCP connections to terminate.

For each of the following scenarios, compute the total time to download the web page and both media files.

(1) Sequential requests with non-persistent TCP connections.

(2) Concurrent requests with non-persistent TCP connections.

(3) Sequential requests with a single persistent TCP connection.

(4) Pipelined requests within a single persistent TCP connection.

(5) We have been assuming that the throughput for sending media files is T for a single connection, and $\frac{T}{n}$ for n concurrent connections. However, depending on the size of the media files, we can make more inferences about how fast we can send the media files. If the media files are very small, what kind of delay would dominate the time it would take to send them? What if the files are very large?