15-440 HW 4: Distributed Systems Examples

Due: In recitation on Wednesday, 3/29/09

April 22, 2009

1 DNS

In this question, you will explore part of the DNS tree using `dig` and `whois`. For each of the hostnames below, give a list of the DNS servers that would be queried to eventually find out the IP address for the host. Annotate each entry with the entity (individual or company) that controls that DNS server.

To achieve this, you will first want to use `dig`. `dig` allows you to query a DNS server for information about a particular domain. For instance,

```
dig @a.root-servers.net www.fmylife.com
```

will query the `a.root-servers.net` root DNS server for information on `www.fmylife.com`. You may start from any root server you wish. You need only provide one chain of DNS servers that lead you to the IP address for the given hostname.

For each DNS server you use, you will need to use `whois` to determine who controls the server. For instance,

```
whois betacie.com
```

will return information on `betacie.com`, leading us to Maxime Valette.

a) www.cs.cmu.edu

b) www.google.com

c) www.xkcd.org

d) www.whitehouse.gov

2 Routing

When we discussed the distance vector routing (i.e. RIP) in class, we said it suffers from the count to infinity problem.

a) Give an example showing how the count to infinity problem arises, despite attempts to address it with split horizon with poison reverse.

b) What "solution" does RIP use to avoid counting forever?

c) How does OSPF avoid suffering from the counting to infinity problem?

3 Shared Memory Revisited

As we discussed in class, shared memory systems are typically NUMA (Non-Uniform Memory Access) systems, meaning that for each processor some memory is faster than other memory. This difference is hidden from the programmer as all of the memory collectively fills one address space. Thus, the programmer has no idea which memory is local and fast and which memory is non-local and therefore slow.

a) A typical access to physical memory in the same machine takes on the order of 200 cycles. Assuming a 2.0Ghz processor, how long will this take in seconds?

b) `pople` uses a technology called NUMAlink to access memory on other nodes. NUMAlink has a worst-case latency of 800ns. How many orders of magnitude slower is non-local memory on `pople` than local memory?

c) Now lets consider what would happen if we used shared memory for a system going over the internet. We'll assume that we have good, fast connections to the internet at all machines. What is the roundtrip time from CMU to Google? (Hint: use `ping`)

d) How many orders of magnitude slower would non-local memory across the internet be than local memory?

e) With shared memory, we trade away knowledge of and control over where in memory data resides for a convenient abstraction. Why are we willing to make this trade on a system like `pople`, but not on a system spanning the internet?

4 Distributed File Systems

Consider the original NFS design as compared to the familiar AFS design. NFS's servers were "stateless". AFS's servers maintain certain state – what do they "remember" and why? What is the cost-benefit trade-off?