Section 8

No Such A record

Some organizations might get touchy about us browsing their DNS records, so we chose an innocuous target: the National Scrabble Association, whose website is presumably www.nsa.gov.

a) In general, what would we expect if we searched for an A record? An IP address
b) When we searched for the A record, we got:

```
1 dsdn-gh1-uea05.nsa.gov 63.239.67.11 AUTH Recieved 1 Answers , rcode= CNAME:
=wac.87c5.www.nsa.gov.att-acdn.net,
```

What does this mean?

www.nsa.gov is an alias name (those sneaky scrabble players!) for wac.87c5.www.nsa.gov.att-acdn.net

We also note that the NSA is using a Content Distribution Network (CDN).

c) What would we expect if we searched for the NS record for www.nsa.gov?

The name of one (or more!) authoritative name servers for the domain e.g.,

```
NS: dsdn-gh1-uea06.nsa.gov,NS: dsdn-gh1-uea05.nsa.gov
```

d) The A record for wac.87c5.www.nsa.gov.att-acdn.net is:

```
A      wac.87c5.www.nsa.gov.att-acdn.net 12.120.140.185      1 min
```

What does this say about the IP address of www.nsa.gov?

This is **one** possible IP address. The short TTL suggests that they may want to use different ones, such as for load balancing.

e) We'd like to email the NSA a few questions about the finer points of scrabble (is "CDN" a legal acronym? What about "TCP"?). What record type should we look up if we wanted to know what their mail servers were? MX
f) If we looked up the PTR record for 63.239.67.11.in-addr.arpa (this is the IP address mentioned in part b), what would we get?

Nothing! We need reversed IP quads (in which case we would expect 'dsdn-gh1-uea05.nsa.gov').

g) In the introduction, we stated that some organizations might not like us looking up their DNS records. Does the NSA know about us?

~~Depends on whether the record was cached.~~ The NSA knows everything.

HTTP

Trivia[1]: Response Codes

In class we learnt that HTTP response codes fall into broad classes of:

Code	Class	Example
1xx	Informational	100 Continue
2xx	Success	200 OK
3xx	Redirection	304 Not Modified
4xx	Client error	404 Not Found
5xx	Server error	503 Service Unavailable

Categorize the following HTTP response messages into the classes above:

- Unauthorized 403
- Switching Protocols 101
- Partial Content 206
- Not Modified 304
- No Content 204
- Multiple Choices 300
- Moved Permanently 301
- Insufficient Storage 507
- I'm a teapot 418
- Gone 410
- Enhance Your Calm[2] 420

Hint: there is 1 of 1xx, 2 of 2xx, 3 of 3xx, and 4 of 4xx.

HTTP Performance (based on 2011 HW3)

We would like to download three media files, each of size M. However, we can't start downloading the media files until we finish downloading a webpage, of size P. Assume that HTTP request packets, TCP SYNs and ACKs are very small, the connections can each achieve throughput T (but if there are multiple connections they must share it), and we don't need to wait for the HTTP responses to be acknowledged, nor for TCP connections to terminate.

a) For each of the following scenarios, calculate the time to download the page and media files:

	Page	Media 1	Media 2	Media 3	Total

[1] Aka things you don't need to memorize unless you make bets with the professor …

[2] Non-standard (https://dev.twitter.com/docs/error-codes-responses). Can you guess the exact number?

<table>
<tr><td rowspan="2">Sequential requests with non-persistent TCP connections</td><td rowspan="2">1 RTT (TCP)
1 RTT (HTTP)
P/T</td><td rowspan="2">1 RTT (TCP)
1 RTT (HTTP)
M/T</td><td rowspan="2">1 RTT (TCP)
1 RTT (HTTP)
M/T</td><td rowspan="2">1 RTT (TCP)
1 RTT (HTTP)
M/T</td><td rowspan="2">8 RTTs
+ P/T
+ 3M/T</td></tr>
<tr></tr>
<tr><td rowspan="2">Concurrent with non-persistent TCP connections</td><td rowspan="2">1 RTT (TCP)
1 RTT (HTTP)
P/T</td><td rowspan="2">1 RTT (TCP)
1 RTT (HTTP)
M/(T/3)</td><td rowspan="2">1 RTT (TCP)
1 RTT (HTTP)
M/(T/3)</td><td rowspan="2">1 RTT (TCP)
1 RTT (HTTP)
M/(T/3)</td><td rowspan="2">4 RTTs
+ P/T
+ 3M/T</td></tr>
<tr></tr>
<tr><td rowspan="2">Sequential with a single persistent TCP connection</td><td colspan="4">1 RTT (TCP)</td><td rowspan="2">5 RTTs
+ P/T
+ 3M/T</td></tr>
<tr><td>1 RTT (HTTP)
P/T</td><td>1 RTT (HTTP)
M/T</td><td>1 RTT (HTTP)
M/T</td><td>1 RTT (HTTP)
M/T</td></tr>
<tr><td rowspan="4">Pipelined within a single persistent TCP connection</td><td colspan="4">1 RTT (TCP)</td><td rowspan="4">3 RTTs
+ P/T
+ 3M/T</td></tr>
<tr><td>1 RTT (HTTP)
P/T</td><td colspan="3"></td></tr>
<tr><td rowspan="2"></td><td colspan="3">1 RTT (HTTP)</td></tr>
<tr><td>M/T</td><td>M/T</td><td>M/T</td></tr>
</table>

b) If we knew the direct links to the media files, and therefore did not need to download the webpage at all, how would this affect the total time?
For the non-persistent TCP connections: subtract [2 RTTs (1 for TCP, 1 for HTTP) + P/T]
For the persistent TCP connections: subtract [1 RTT (for HTTP) + P/T]

c) If the media files are very small, what dominates the total time for each case?
Propagation delay (the RTTs; different for each case) and transmission delay of the web page (P/T – but we might expect the web page to be small; same for each case)

d) If the media files are very large, what dominates the total time for each case?
Transmission delay of the media files (3M/T; same for each case)