COMP3331 LAB 3
Abanob Tawfik
Z5075490

Contents

Exercise 3: Digging into DNS (all dig output is also put in dig.txt)

What is the IP address of www.cecs.anu.edu.au? What type of DNS query is sent to get this answer?

The IP address of www.cecs.anu.edu.au is 150.203.161.98. The type of DNS query sent to get this answer is an iterative query as we are first redirected to rproxy.cecs.anu.edu.au then given the IP address from there. The IP address is highlighted below in Figure 1.

```
weber % dig www.cecs.anu.edu.au

; <<>> DiG 9.7.3 <<>> www.cecs.anu.edu.au
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 59160
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 3, ADDITIONAL: 6

;; QUESTION SECTION:
;www.cecs.anu.edu.au.           IN      A

;; ANSWER SECTION:
www.cecs.anu.edu.au.    3596    IN      CNAME   rproxy.cecs.anu.edu.au.
rproxy.cecs.anu.edu.au. 1641    IN      A       150.203.161.98

;; AUTHORITY SECTION:
cecs.anu.edu.au.        355     IN      NS      ns3.cecs.anu.edu.au.
cecs.anu.edu.au.        355     IN      NS      ns4.cecs.anu.edu.au.
cecs.anu.edu.au.        355     IN      NS      ns2.cecs.anu.edu.au.

;; ADDITIONAL SECTION:
ns2.cecs.anu.edu.au.    2154    IN      A       150.203.161.36
ns2.cecs.anu.edu.au.    354     IN      AAAA    2001:388:1034:2905::24
ns3.cecs.anu.edu.au.    2154    IN      A       150.203.161.50
ns3.cecs.anu.edu.au.    2154    IN      AAAA    2001:388:1034:2905::32
ns4.cecs.anu.edu.au.    2154    IN      A       150.203.161.38
ns4.cecs.anu.edu.au.    2154    IN      AAAA    2001:388:1034:2905::26

;; Query time: 0 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 03:31:29 2018
;; MSG SIZE  rcvd: 260
```

Figure 1 IP address of cecs.anu.edu.au

What is the canonical name for the CECS ANU web server? What is its IP address? Suggest a reason for having an alias for this server.

The canonical name for the CECS ANU web server is rproxy.cecs.anu.edu.au highlighted in Figure 2. The IP address associated to that server is 150.203.161.98, similar to the one above. A reason for having an alias for this server is that it is much easier to remember and identify a simpler name. it is much easier to identify www.cecs.anu.edu.au than rproxy.cecs.anu.edu.au.

```
weber % dig www.cecs.anu.edu.au

; <<>> DiG 9.7.3 <<>> www.cecs.anu.edu.au
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43210
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 3, ADDITIONAL: 6

;; QUESTION SECTION:
;www.cecs.anu.edu.au.           IN      A

;; ANSWER SECTION:
www.cecs.anu.edu.au.    3292    IN      CNAME   rproxy.cecs.anu.edu.au.
rproxy.cecs.anu.edu.au. 1337    IN      A       150.203.161.98

;; AUTHORITY SECTION:
cecs.anu.edu.au.        51      IN      NS      ns3.cecs.anu.edu.au.
cecs.anu.edu.au.        51      IN      NS      ns4.cecs.anu.edu.au.
cecs.anu.edu.au.        51      IN      NS      ns2.cecs.anu.edu.au.

;; ADDITIONAL SECTION:
ns2.cecs.anu.edu.au.    1850    IN      A       150.203.161.36
ns2.cecs.anu.edu.au.    50      IN      AAAA    2001:388:1034:2905::24
ns3.cecs.anu.edu.au.    1850    IN      A       150.203.161.50
ns3.cecs.anu.edu.au.    1850    IN      AAAA    2001:388:1034:2905::32
ns4.cecs.anu.edu.au.    1850    IN      A       150.203.161.38
ns4.cecs.anu.edu.au.    1850    IN      AAAA    2001:388:1034:2905::26

;; Query time: 0 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 03:36:33 2018
;; MSG SIZE  rcvd: 260
```

Figure 2 the canonical name of cecs.anu.edu.au

What can you make of the rest of the response (i.e. the details available in the Authority and Additional sections)?

The Authority section contains all the authoritative name servers for the domain name. we have 3 authoritative name servers for www.cecs.anu.edu.au

- ns2.cecs.anu.edu.au
- ns3.cecs.anu.edu.au
- ns4.cecs.anu.edu.au

The additional section contains the IP addresses for all the name servers, it contains both A records and AAAA records, which are the IPv4 address and IPv6 address for those servers respectively.

What is the IP address of the local nameserver for your machine?

The IP address of the local nameserver for my machine was 129.94.242.2. this is highlighted below in Figure 3.

```
weber % dig www.cecs.anu.edu.au

; <<>> DiG 9.7.3 <<>> www.cecs.anu.edu.au
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43210
;; flags: qr rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 3, ADDITIONAL: 6

;; QUESTION SECTION:
;www.cecs.anu.edu.au.           IN      A

;; ANSWER SECTION:
www.cecs.anu.edu.au.    3292    IN      CNAME   rproxy.cecs.anu.edu.au.
rproxy.cecs.anu.edu.au. 1337    IN      A       150.203.161.98

;; AUTHORITY SECTION:
cecs.anu.edu.au.        51      IN      NS      ns3.cecs.anu.edu.au.
cecs.anu.edu.au.        51      IN      NS      ns4.cecs.anu.edu.au.
cecs.anu.edu.au.        51      IN      NS      ns2.cecs.anu.edu.au.

;; ADDITIONAL SECTION:
ns2.cecs.anu.edu.au.    1850    IN      A       150.203.161.36
ns2.cecs.anu.edu.au.    50      IN      AAAA    2001:388:1034:2905::24
ns3.cecs.anu.edu.au.    1850    IN      A       150.203.161.50
ns3.cecs.anu.edu.au.    1850    IN      AAAA    2001:388:1034:2905::32
ns4.cecs.anu.edu.au.    1850    IN      A       150.203.161.38
ns4.cecs.anu.edu.au.    1850    IN      AAAA    2001:388:1034:2905::26

;; Query time: 0 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 03:36:33 2018
;; MSG SIZE  rcvd: 260
```

Figure 3 IP address of my local machine highlighted

What are the DNS nameservers for the “cecs.anu.edu.au” domain? Find out their IP addresses. What type of DNS query is sent to obtain this information?

The DNS name servers for cecs.anu.edu.au are

- ns2.cecs.anu.edu.au
- ns3.cecs.anu.edu.au
- ns4.cecs.anu.edu.au

This is highlighted below in Figure 4.

```
weber % dig cecs.anu.edu.au NS

; <<>> DiG 9.7.3 <<>> cecs.anu.edu.au NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16639
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 6

;; QUESTION SECTION:
;cecs.anu.edu.au.               IN      NS

;; ANSWER SECTION:
cecs.anu.edu.au.        1800    IN      NS      ns4.cecs.anu.edu.au.
cecs.anu.edu.au.        1800    IN      NS      ns2.cecs.anu.edu.au.
cecs.anu.edu.au.        1800    IN      NS      ns3.cecs.anu.edu.au.

;; ADDITIONAL SECTION:
ns2.cecs.anu.edu.au.    115     IN      A       150.203.161.36
ns2.cecs.anu.edu.au.    116     IN      AAAA    2001:388:1034:2905::24
ns3.cecs.anu.edu.au.    115     IN      A       150.203.161.50
ns3.cecs.anu.edu.au.    115     IN      AAAA    2001:388:1034:2905::32
ns4.cecs.anu.edu.au.    115     IN      A       150.203.161.38
ns4.cecs.anu.edu.au.    115     IN      AAAA    2001:388:1034:2905::26

;; Query time: 29 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 04:05:28 2018
;; MSG SIZE  rcvd: 219
```

Figure 4 all the name servers for cecs.anu.edu.au

There are two types of IP addresses for the name servers, IPv4 and IPv6 addresses. IPv4 is a 32-bit IP address and IPv6 is a 128-bit IP address. Below are the IP addresses of the nameservers.

- ns2.cecs.anu.edu.au
  - IPv4 address – 150.203.161.36
  - IPv6 address – 2201:388:1034:2905::24
- ns3.cecs.anu.edu.au
  - IPv4 address – 150.203.101.50
  - IPv6 address – 2001:388:1034:2905::32
- ns4.cecs.anu.edu.au
  - IPv4 address – 150.203.101.38
  - IPv6 address – 2001:388:1034:2905::26

These addresses are highlighted below in Figure 5.

```
weber % dig cecs.anu.edu.au NS

; <<>> DiG 9.7.3 <<>> cecs.anu.edu.au NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16639
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 0, ADDITIONAL: 6

;; QUESTION SECTION:
;cecs.anu.edu.au.               IN      NS

;; ANSWER SECTION:
cecs.anu.edu.au.        1800    IN      NS      ns4.cecs.anu.edu.au.
cecs.anu.edu.au.        1800    IN      NS      ns2.cecs.anu.edu.au.
cecs.anu.edu.au.        1800    IN      NS      ns3.cecs.anu.edu.au.

;; ADDITIONAL SECTION:
ns2.cecs.anu.edu.au.    115     IN      A       150.203.161.36
ns2.cecs.anu.edu.au.    116     IN      AAAA    2001:388:1034:2905::24
ns3.cecs.anu.edu.au.    115     IN      A       150.203.161.50
ns3.cecs.anu.edu.au.    115     IN      AAAA    2001:388:1034:2905::32
ns4.cecs.anu.edu.au.    115     IN      A       150.203.161.38
ns4.cecs.anu.edu.au.    115     IN      AAAA    2001:388:1034:2905::26

;; Query time: 29 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 04:05:28 2018
;; MSG SIZE  rcvd: 219
```

Figure 5 the IP addresses of the name servers

The DNS query sent to obtain this information is a NS query to obtain nameserver.

What is the DNS name associated with the IP address 149.171.158.109? What type of DNS query is sent to obtain this information?

The DNS name associated with the IP address 149.171.158.109 is www.engineering.unsw.edu.au, engplws008.eng.unsw.edu.au and engplws008.ad.unsw.edu.au. The machine with the IP address 149.171.158.109 would be hosting the web server for the UNSW engineering website. There are also other services provided from this machine from the engplws008. The type of DNS query sent to obtain this information was a reverse lookup using the -x flag in the query. This is highlighted below in Figure 6.

```
weber % dig -x 149.171.158.109

; <<>> DiG 9.7.3 <<>> -x 149.171.158.109
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 44131
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 3, ADDITIONAL: 6

;; QUESTION SECTION:
;109.158.171.149.in-addr.arpa.  IN      PTR

;; ANSWER SECTION:
109.158.171.149.in-addr.arpa. 3551 IN   PTR     engplws008.eng.unsw.edu.au.
109.158.171.149.in-addr.arpa. 3551 IN   PTR     www.engineering.unsw.edu.au.
109.158.171.149.in-addr.arpa. 3551 IN   PTR     engplws008.ad.unsw.edu.au.

;; AUTHORITY SECTION:
158.171.149.in-addr.arpa. 2061  IN      NS      ns2.unsw.edu.au.
158.171.149.in-addr.arpa. 2061  IN      NS      ns1.unsw.edu.au.
158.171.149.in-addr.arpa. 2061  IN      NS      ns3.unsw.edu.au.

;; ADDITIONAL SECTION:
ns1.unsw.edu.au.        8344    IN      A       129.94.0.192
ns1.unsw.edu.au.        1791    IN      AAAA    2001:388:c:35::1
ns2.unsw.edu.au.        8344    IN      A       129.94.0.193
ns2.unsw.edu.au.        1791    IN      AAAA    2001:388:c:35::2
ns3.unsw.edu.au.        8344    IN      A       192.155.82.178
ns3.unsw.edu.au.        1791    IN      AAAA    2600:3c01::f03c:91ff:fe73:5f10

;; Query time: 0 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 04:23:18 2018
;; MSG SIZE  rcvd: 330
```

Figure 6 reverse lookup on ip address

Run dig and query the CSE nameserver (129.94.242.33) for the mail servers for Yahoo! Did you get an authoritative answer? Why?

When running dig and performing the query, we receive a non-authoritative answer. This can be seen from the flags in the header, highlighted in Figure 7. There is no aa flag, which is the "authoritative answer" flag. This is because the CSE nameserver has no authority over the Yahoo domain, it only has authority over the CSE domain.

```
weber % dig @129.94.242.33 yahoo.com MX

; <<>> DiG 9.7.3 <<>> @129.94.242.33 yahoo.com MX
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 23915
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 5, ADDITIONAL: 8

;; QUESTION SECTION:
;yahoo.com.                     IN      MX

;; ANSWER SECTION:
yahoo.com.              1345    IN      MX      1 mta5.am0.yahoodns.net.
yahoo.com.              1345    IN      MX      1 mta6.am0.yahoodns.net.
yahoo.com.              1345    IN      MX      1 mta7.am0.yahoodns.net.

;; AUTHORITY SECTION:
yahoo.com.              123528  IN      NS      ns3.yahoo.com.
yahoo.com.              123528  IN      NS      ns1.yahoo.com.
yahoo.com.              123528  IN      NS      ns2.yahoo.com.
yahoo.com.              123528  IN      NS      ns4.yahoo.com.
yahoo.com.              123528  IN      NS      ns5.yahoo.com.

;; ADDITIONAL SECTION:
ns1.yahoo.com.          172415  IN      A       68.180.131.16
ns1.yahoo.com.          36836   IN      AAAA    2001:4998:130::1001
ns2.yahoo.com.          454051  IN      A       68.142.255.16
ns2.yahoo.com.          32811   IN      AAAA    2001:4998:140::1002
ns3.yahoo.com.          216138  IN      A       203.84.221.53
ns3.yahoo.com.          72892   IN      AAAA    2406:8600:b8:fe03::1003
ns4.yahoo.com.          460855  IN      A       98.138.11.157
ns5.yahoo.com.          446926  IN      A       119.160.253.83

;; Query time: 0 msec
;; SERVER: 129.94.242.33#53(129.94.242.33)
;; WHEN: Sun Aug 12 04:54:00 2018
;; MSG SIZE  rcvd: 360
```

Figure 7 the flags for the dig query

Repeat the above but use one of the nameservers obtained in Question 5. What is the result?

When we repeat the above using ns2.cecs.anu.edu.au we obtain no response. This is seen from the status: REFUSED in the header, shown in Figure 8. This could possibly be due to the cecs.anu network prohibiting users who are not part of the network from performing DNS queries outside their network. This result also happens when trying to do any look up for any domain name. however not when it is done on the cecs website itself. As shown in Figure 9.

```
weber % dig @150.203.101.50 yahoo.com MX

; <<>> DiG 9.7.3 <<>> @150.203.101.50 yahoo.com MX
; (1 server found)
;; global options: +cmd
;; connection timed out; no servers could be reached
weber % dig @ns2.cecs.anu.edu.au yahoo.com MX

; <<>> DiG 9.7.3 <<>> @ns2.cecs.anu.edu.au yahoo.com MX
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: REFUSED, id: 9498
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;yahoo.com.                     IN      MX

;; Query time: 7 msec
;; SERVER: 150.203.161.36#53(150.203.161.36)
;; WHEN: Sun Aug 12 05:07:21 2018
;; MSG SIZE  rcvd: 27
```

Figure 8 refusal of DNS query

```
weber % dig @ns2.cecs.anu.edu.au cecs.anu.edu.au

; <<>> DiG 9.7.3 <<>> @ns2.cecs.anu.edu.au cecs.anu.edu.au
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5315
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 6
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;cecs.anu.edu.au.               IN      A

;; ANSWER SECTION:
cecs.anu.edu.au.        3600    IN      A       150.203.161.98

;; AUTHORITY SECTION:
cecs.anu.edu.au.        3600    IN      NS      ns3.cecs.anu.edu.au.
cecs.anu.edu.au.        3600    IN      NS      ns2.cecs.anu.edu.au.
cecs.anu.edu.au.        3600    IN      NS      ns4.cecs.anu.edu.au.

;; ADDITIONAL SECTION:
ns2.cecs.anu.edu.au.    3600    IN      A       150.203.161.36
ns2.cecs.anu.edu.au.    3600    IN      AAAA    2001:388:1034:2905::24
ns3.cecs.anu.edu.au.    3600    IN      A       150.203.161.50
ns3.cecs.anu.edu.au.    3600    IN      AAAA    2001:388:1034:2905::32
ns4.cecs.anu.edu.au.    3600    IN      A       150.203.161.38
ns4.cecs.anu.edu.au.    3600    IN      AAAA    2001:388:1034:2905::26

;; Query time: 8 msec
;; SERVER: 150.203.161.36#53(150.203.161.36)
;; WHEN: Sun Aug 12 05:17:12 2018
;; MSG SIZE  rcvd: 235
```

Figure 9 successful DNS query to cecs.anu.edu.au website using name server as authority.

Obtain the authoritative answer for the mail servers for Yahoo! mail. What type of DNS query is sent to obtain this information?

To obtain an authoritative answer we can use the yahoo name server obtained from the authority section in question 7 when we did our query using the CSE nameserver. Using ns1.yahoo.com to perform our query we obtain our answer with aa (authoritative answer) in the flag shown in figure 10.

The mail servers for yahoo.com are the following

- Mta5.am0.yahoodns.net
- Mta6.am0.yahoodns.net
- Mta7.am0.yahoodns.net

```
weber % dig @ns1.yahoo.com yahoo.com MX

; <<>> DiG 9.7.3 <<>> @ns1.yahoo.com yahoo.com MX
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 59762
;; flags: qr aa rd; QUERY: 1, ANSWER: 3, AUTHORITY: 5, ADDITIONAL: 8
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;yahoo.com.                     IN      MX

;; ANSWER SECTION:
yahoo.com.              1800    IN      MX      1 mta7.am0.yahoodns.net.
yahoo.com.              1800    IN      MX      1 mta5.am0.yahoodns.net.
yahoo.com.              1800    IN      MX      1 mta6.am0.yahoodns.net.

;; AUTHORITY SECTION:
yahoo.com.              172800  IN      NS      ns5.yahoo.com.
yahoo.com.              172800  IN      NS      ns3.yahoo.com.
yahoo.com.              172800  IN      NS      ns4.yahoo.com.
yahoo.com.              172800  IN      NS      ns2.yahoo.com.
yahoo.com.              172800  IN      NS      ns1.yahoo.com.

;; ADDITIONAL SECTION:
ns1.yahoo.com.          1209600 IN      A       68.180.131.16
ns2.yahoo.com.          1209600 IN      A       68.142.255.16
ns3.yahoo.com.          1209600 IN      A       203.84.221.53
ns4.yahoo.com.          1209600 IN      A       98.138.11.157
ns5.yahoo.com.          1209600 IN      A       119.160.253.83
ns1.yahoo.com.          86400   IN      AAAA    2001:4998:130::1001
ns2.yahoo.com.          86400   IN      AAAA    2001:4998:140::1002
ns3.yahoo.com.          86400   IN      AAAA    2406:8600:b8:fe03::1003

;; Query time: 145 msec
;; SERVER: 68.180.131.16#53(68.180.131.16)
;; WHEN: Sun Aug 12 05:24:13 2018
;; MSG SIZE  rcvd: 360
```

Figure 10 authoritative answer for mail servers for yahoo

In this exercise you simulate the iterative DNS query process to find the IP address of your machine. How many DNS servers do you have to query to get the authoritative answer?

Performing a NS DNS query on root we achieve the result shown in Figure 11.

```
; <<>> DiG 9.7.3 <<>> . NS
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 65411
;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 13

;; QUESTION SECTION:
;.                              IN      NS

;; ANSWER SECTION:
.                       309590  IN      NS      g.root-servers.net.
.                       309590  IN      NS      k.root-servers.net.
.                       309590  IN      NS      l.root-servers.net.
.                       309590  IN      NS      h.root-servers.net.
.                       309590  IN      NS      b.root-servers.net.
.                       309590  IN      NS      d.root-servers.net.
.                       309590  IN      NS      j.root-servers.net.
.                       309590  IN      NS      c.root-servers.net.
.                       309590  IN      NS      i.root-servers.net.
.                       309590  IN      NS      e.root-servers.net.
.                       309590  IN      NS      m.root-servers.net.
.                       309590  IN      NS      a.root-servers.net.
.                       309590  IN      NS      f.root-servers.net.

;; ADDITIONAL SECTION:
a.root-servers.net.     398716  IN      A       198.41.0.4
a.root-servers.net.     376947  IN      AAAA    2001:503:ba3e::2:30
b.root-servers.net.     297094  IN      A       199.9.14.201
c.root-servers.net.     43526   IN      A       192.33.4.12
e.root-servers.net.     31828   IN      A       192.203.230.10
e.root-servers.net.     28577   IN      AAAA    2001:500:a8::e
f.root-servers.net.     342555  IN      A       192.5.5.241
g.root-servers.net.     28577   IN      AAAA    2001:500:12::d0d
i.root-servers.net.     31828   IN      A       192.36.148.17
j.root-servers.net.     377637  IN      A       192.58.128.30
k.root-servers.net.     299464  IN      A       193.0.14.129
l.root-servers.net.     118074  IN      A       199.7.83.42
m.root-servers.net.     196650  IN      A       202.12.27.33

;; Query time: 0 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 05:30:17 2018
;; MSG SIZE  rcvd: 472
```

Figure 11 DNS query on . to find authoritative name servers

Now we perform a query to find .au using a root server. The result is shown below in figure 12.

```
weber % dig @a.root-servers.net lyre00.cse.unsw.edu.au

; <<>> DiG 9.7.3 <<>> @a.root-servers.net lyre00.cse.unsw.edu.au
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37350
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 10, ADDITIONAL: 15
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;lyre00.cse.unsw.edu.au.                IN      A

;; AUTHORITY SECTION:
au.                     172800  IN      NS      a.au.
au.                     172800  IN      NS      b.au.
au.                     172800  IN      NS      c.au.
au.                     172800  IN      NS      d.au.
au.                     172800  IN      NS      q.au.
au.                     172800  IN      NS      r.au.
au.                     172800  IN      NS      s.au.
au.                     172800  IN      NS      t.au.
au.                     172800  IN      NS      u.au.
au.                     172800  IN      NS      v.au.

;; ADDITIONAL SECTION:
a.au.                   172800  IN      A       58.65.254.73
b.au.                   172800  IN      A       58.65.253.73
c.au.                   172800  IN      A       162.159.24.179
d.au.                   172800  IN      A       162.159.25.38
q.au.                   172800  IN      A       65.22.196.1
r.au.                   172800  IN      A       65.22.197.1
s.au.                   172800  IN      A       65.22.198.1
t.au.                   172800  IN      A       65.22.199.1
u.au.                   172800  IN      A       211.29.133.32
v.au.                   172800  IN      A       202.12.31.53
a.au.                   172800  IN      AAAA    2407:6e00:254:306::73
b.au.                   172800  IN      AAAA    2407:6e00:253:306::73
c.au.                   172800  IN      AAAA    2400:cb00:2049:1::a29f:18b3
d.au.                   172800  IN      AAAA    2400:cb00:2049:1::a29f:1926
q.au.                   172800  IN      AAAA    2a01:8840:be::1

;; Query time: 213 msec
;; SERVER: 198.41.0.4#53(198.41.0.4)
;; WHEN: Sun Aug 12 05:38:42 2018
;; MSG SIZE  rcvd: 500
```

Figure 12 dns query using the .a root to find the authoratative ip for .au on lyre

Now we can find the authoritative nameserver for .edu.au using the a.au to perform our DNS query shown in Figure 13.

```
weber % dig @a.au lyre00.cse.unsw.edu.au

; <<>> DiG 9.7.3 <<>> @a.au lyre00.cse.unsw.edu.au
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18219
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 4, ADDITIONAL:
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;lyre00.cse.unsw.edu.au.                IN      A

;; AUTHORITY SECTION:
edu.au.                 86400   IN      NS      t.au.
edu.au.                 86400   IN      NS      r.au.
edu.au.                 86400   IN      NS      s.au.
edu.au.                 86400   IN      NS      q.au.

;; ADDITIONAL SECTION:
q.au.                   86400   IN      A       65.22.196.1
r.au.                   86400   IN      A       65.22.197.1
s.au.                   86400   IN      A       65.22.198.1
t.au.                   86400   IN      A       65.22.199.1
q.au.                   86400   IN      AAAA    2a01:8840:be::1
r.au.                   86400   IN      AAAA    2a01:8840:bf::1
s.au.                   86400   IN      AAAA    2a01:8840:c0::1
t.au.                   86400   IN      AAAA    2a01:8840:c1::1

;; Query time: 15 msec
;; SERVER: 58.65.254.73#53(58.65.254.73)
;; WHEN: Sun Aug 12 05:41:41 2018
;; MSG SIZE  rcvd: 280
```

Figure 13 authoritative name server for .edu.au

Now we can use q.au. to find the authoritative name server for unsw that contain lyre00.cse.unsw.edu.au shown in Figure 14.

```
weber % dig @q.au lyre00.cse.unsw.edu.au

; <<>> DiG 9.7.3 <<>> @q.au lyre00.cse.unsw.edu.au
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4220
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 3, ADDITIONAL: 5
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;lyre00.cse.unsw.edu.au.                IN      A

;; AUTHORITY SECTION:
unsw.edu.au.            900     IN      NS      ns3.unsw.edu.au.
unsw.edu.au.            900     IN      NS      ns2.unsw.edu.au.
unsw.edu.au.            900     IN      NS      ns1.unsw.edu.au.

;; ADDITIONAL SECTION:
ns1.unsw.edu.au.        900     IN      A       129.94.0.192
ns2.unsw.edu.au.        900     IN      A       129.94.0.193
ns3.unsw.edu.au.        900     IN      A       192.155.82.178
ns1.unsw.edu.au.        900     IN      AAAA    2001:388:c:35::1
ns2.unsw.edu.au.        900     IN      AAAA    2001:388:c:35::2

;; Query time: 13 msec
;; SERVER: 65.22.196.1#53(65.22.196.1)
;; WHEN: Sun Aug 12 05:45:57 2018
;; MSG SIZE  rcvd: 198
```

Figure 14 authoritative name server for unsw.edu.au domains

Now we can use the authoritative unsw name server to find the authoritative name server for CSE domains. This is shown in Figure 15.

```
weber % dig @ns1.unsw.edu.au lyre00.cse.unsw.edu.au

; <<>> DiG 9.7.3 <<>> @ns1.unsw.edu.au lyre00.cse.unsw.edu.au
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 36089
;; flags: qr rd; QUERY: 1, ANSWER: 0, AUTHORITY: 2, ADDITIONAL: 4
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;lyre00.cse.unsw.edu.au.                IN      A

;; AUTHORITY SECTION:
cse.unsw.edu.au.        10800   IN      NS      maestro.orchestra.cse.unsw.edu.au.
cse.unsw.edu.au.        10800   IN      NS      beethoven.orchestra.cse.unsw.edu.au

;; ADDITIONAL SECTION:
beethoven.orchestra.cse.unsw.edu.au. 10800 IN A 129.94.242.2
beethoven.orchestra.cse.unsw.edu.au. 10800 IN A 129.94.172.11
beethoven.orchestra.cse.unsw.edu.au. 10800 IN A 129.94.208.3
maestro.orchestra.cse.unsw.edu.au. 10800 IN A   129.94.242.33

;; Query time: 3 msec
;; SERVER: 129.94.0.192#53(129.94.0.192)
;; WHEN: Sun Aug 12 05:48:56 2018
;; MSG SIZE  rcvd: 160
```

Figure 15 the authoritative name server for cse domains

Finally, we can use the CSE name server to find the IP address for the local host, Shown in Figure 16.

```
weber % dig @beethoven.orchestra.cse.unsw.edu.au lyre00.cse.unsw.edu.au

; <<>> DiG 9.7.3 <<>> @beethoven.orchestra.cse.unsw.edu.au lyre00.cse.unsw.edu.au
; (3 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 41969
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

;; QUESTION SECTION:
;lyre00.cse.unsw.edu.au.                IN      A

;; ANSWER SECTION:
lyre00.cse.unsw.edu.au. 3600    IN      A       129.94.210.20

;; AUTHORITY SECTION:
cse.unsw.edu.au.        3600    IN      NS      maestro.orchestra.cse.unsw.edu.au.
cse.unsw.edu.au.        3600    IN      NS      beethoven.orchestra.cse.unsw.edu.au.

;; ADDITIONAL SECTION:
maestro.orchestra.cse.unsw.edu.au. 3600 IN A    129.94.242.33
beethoven.orchestra.cse.unsw.edu.au. 3600 IN A  129.94.242.2

;; Query time: 0 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 05:52:24 2018
;; MSG SIZE  rcvd: 144
```

Figure 16 The ip address of our machine

The IP address of our machine is 129.94.210.20 simulating an iterative DNS request. First, we asked root for all the. name servers so we can begin our requests, then we went down the tree searching for lyre00.cse.unsw.edu.au.

In total we needed

- 1 request to find the root name servers
- 1 request to find the .au name servers
- 1 request to find the edu.au name servers
- 1 request to find the unsw.edu.au name servers
- 1 request to find the cse.unsw.edu.au name servers
- Finally using that name server to give us an authoritative answer for the IP address.

In total there were 6 requests in order to retrieve the IP address of the machine giving an authority answer.

Can one physical machine have several names and/or IP addresses associated with it?

One physical machine can have several names and/or IP addressed associated with it. A single IP address can have multiple names associated with it as shown in Figure 17. The extra names are the aliases all referring to 1 canonical name.

```
weber % dig -x 149.171.158.109

; <<>> DiG 9.7.3 <<>> -x 149.171.158.109
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38450
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 3, ADDITIONAL: 6

;; QUESTION SECTION:
;109.158.171.149.in-addr.arpa.  IN      PTR

;; ANSWER SECTION:
109.158.171.149.in-addr.arpa. 3600 IN   PTR     engplws008.eng.unsw.edu.au.
109.158.171.149.in-addr.arpa. 3600 IN   PTR     www.engineering.unsw.edu.au.
109.158.171.149.in-addr.arpa. 3600 IN   PTR     engplws008.ad.unsw.edu.au.

;; AUTHORITY SECTION:
158.171.149.in-addr.arpa. 10800 IN      NS      ns2.unsw.edu.au.
158.171.149.in-addr.arpa. 10800 IN      NS      ns3.unsw.edu.au.
158.171.149.in-addr.arpa. 10800 IN      NS      ns1.unsw.edu.au.

;; ADDITIONAL SECTION:
ns1.unsw.edu.au.        2228    IN      A       129.94.0.192
ns1.unsw.edu.au.        9822    IN      AAAA    2001:388:c:35::1
ns2.unsw.edu.au.        2228    IN      A       129.94.0.193
ns2.unsw.edu.au.        6486    IN      AAAA    2001:388:c:35::2
ns3.unsw.edu.au.        2228    IN      A       192.155.82.178
ns3.unsw.edu.au.        6486    IN      AAAA    2600:3c01::f03c:91ff:fe73:5f10

;; Query time: 6 msec
;; SERVER: 129.94.242.2#53(129.94.242.2)
;; WHEN: Sun Aug 12 06:05:14 2018
;; MSG SIZE  rcvd: 330
```

Figure 17 one ip address having multiple names

Exercise 4: A Simple Web Server

The code for the webserver is provided in the file WebServer.java in the src folder as well as further down this report. To run the demonstration first do the following
Javac WebServer.java
Java WebServer 8888 ← port number
Now open a browser and type the following to request html files
127.0.0.1:8888/(FILE)
The browser should now display the file contents and the server will send confirmation (backend) shown below in Figure 18 and Figure 19

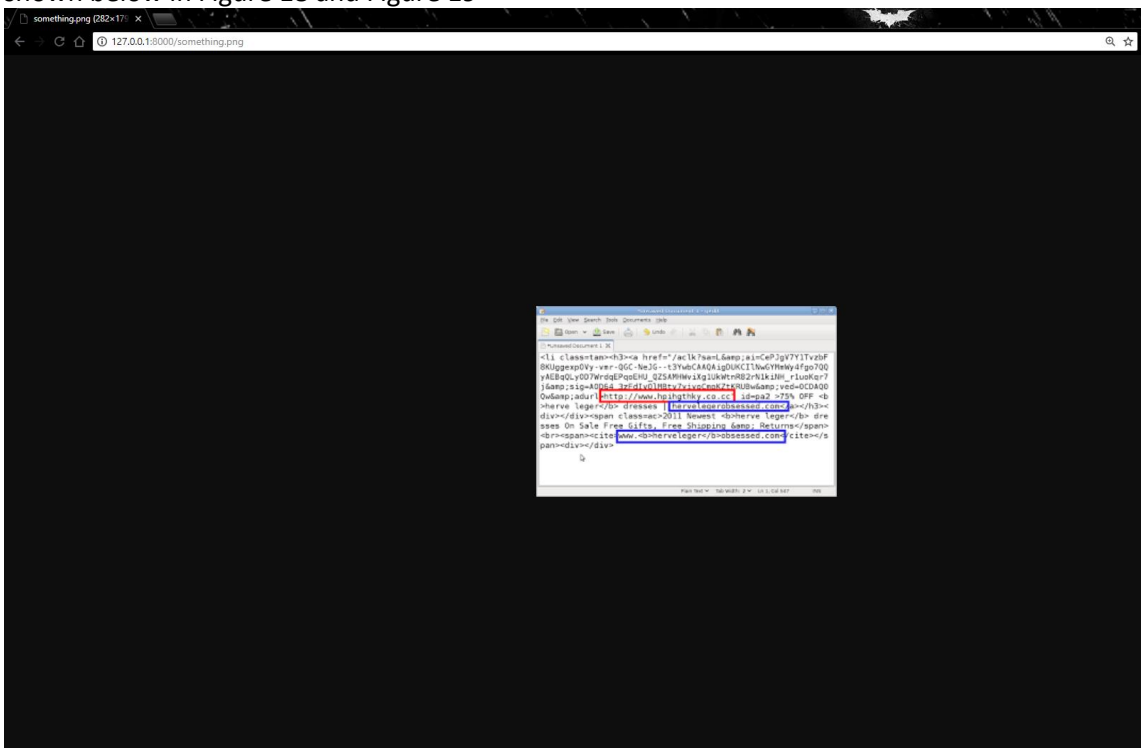

Figure 18 file something.jpg being displayed in browser

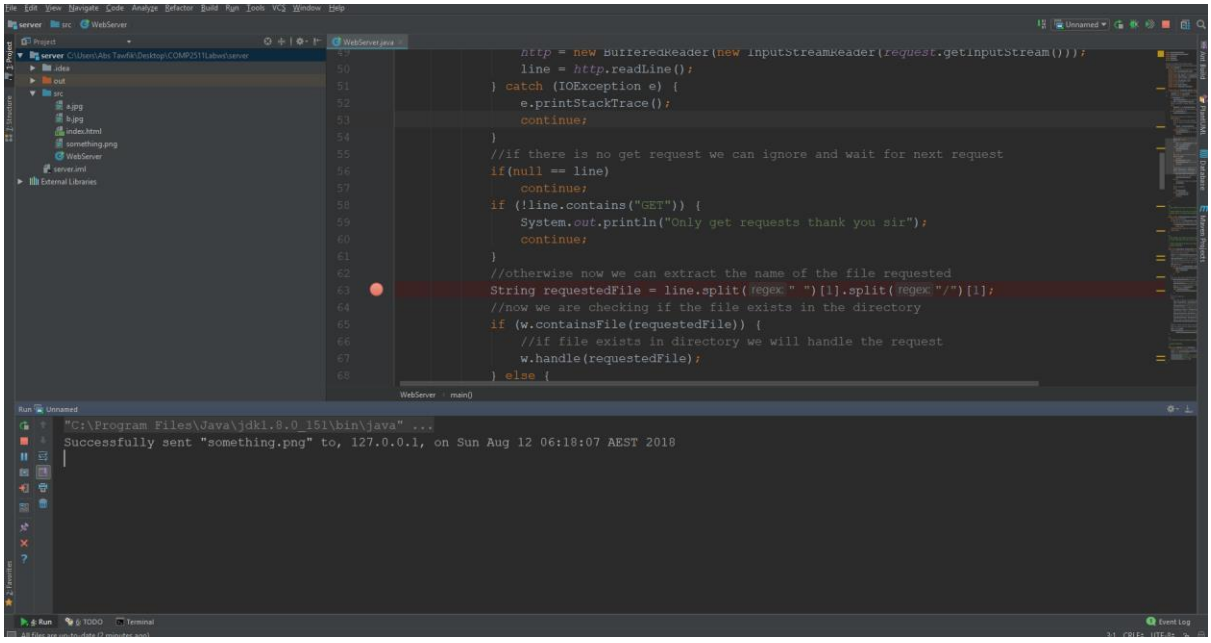

Figure 19 server message showing it sent the file over successfully

In the case the file doesn’t exist error 404 is sent shown below in Figure 20.

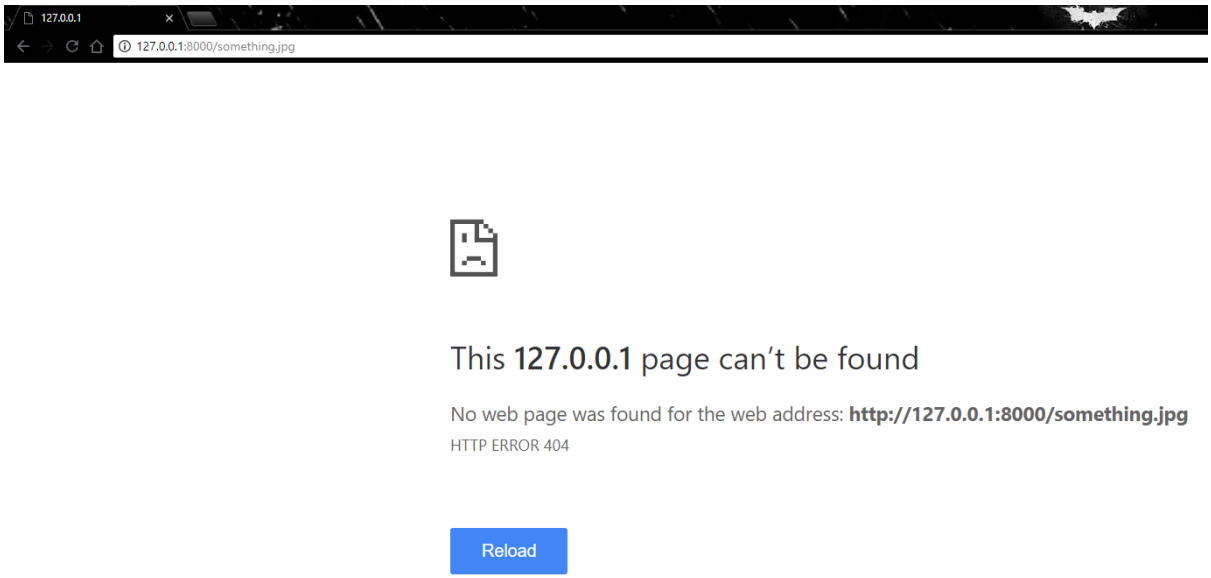

Figure 20 error 404 message if requested file doesn't exist

WebServer.java code below

```java
import javax.imageio.ImageIO;
import java.awt.image.BufferedImage;
import java.net.*;
import java.io.*;
import java.util.*;

//source for sending image through sockets
//https://stackoverflow.com/questions/25086868/how-to-send-images-through-sockets-in-java
public class WebServer {
    //the outputstream for the client who is connecting to website
    private static DataOutputStream output;
    //reading all files and stored them in a list for requests
    private static File folder = new File(System.getProperty("user.dir"));
    private static File[] allfiles = folder.listFiles();
    //allows us to read our http request
    private static BufferedReader http;
    //client + server sockets
    private static Socket request;
    private static ServerSocket webServer;

    public static void main(String args[]) throws IOException {
//      bootstrapping so we can access from a non static context
        WebServer w = new WebServer();
//      if the number of arguements is not 1 print error message
        if (args.length != 1)
            System.out.println("Try java WebServer Port");
        int port = Integer.parseInt(args[0]);
        //try to establish a server connected to a port
        try {
            webServer = new ServerSocket(port);
            //if io exception i.e port number doesnt work catch the exception
        } catch (IOException e) {
            System.out.println("Could not listen on that port number try a different one");
        }
        //now that our server is connected to a port we want it continually processing HTTP
requests
        while (true) {
            //now we want to check for requests
            //try to accept any incoming requests
            try {
                request = webServer.accept();
                //if request could not be established return error message
            } catch (IOException e) {
                System.out.println("could not establish connection");
                continue;
            }
            //now we want to read the first line to proccess request and check if it was a get
request
            String line = null;
            try {
                http = new BufferedReader(new InputStreamReader(request.getInputStream()));
                line = http.readLine();
            } catch (IOException e) {
                e.printStackTrace();
                continue;
            }
            //if there is no get request we can ignore and wait for next request
            if(null == line)
                continue;
            if (!line.contains("GET")) {
                System.out.println("Only get requests thank you sir");
                continue;
            }
            //otherwise now we can extract the name of the file requested
            String requestedFile = line.split(" ")[1].split("/")[1];
            //now we are checking if the file exists in the directory
            if (w.containsFile(requestedFile)) {
                //if file exists in directory we will handle the request
                w.handle(requestedFile);
            } else {
                //otherwise we will send error404
                w.error404();
            }
```

```
            //close connection
            try {
                request.close();
            } catch (IOException e) {
                e.printStackTrace();
                continue;
            }
        }
    }

    /**
     * This function will check if a file exists in the currently directory
     *
     * @param fileName the file name we are checking for in the directory
     * @return true if it exists, false otherwise
     */
    private boolean containsFile(String fileName) {
        //scan through directory
        for (int i = 0; i < allfiles.length; i++)
            //if the file name matches one in the directory we return true
            if (allfiles[i].getName().equals(fileName))
                return true;
        //otherwise return false if no files match
        return false;
    }

    /**
     * This function will handle the incoming HTTP request
     * it will first check the type of file it is, and then proccess it accordingly
     *
     * @param requestedFile the name of the file being requested by client
     * @throws IOException
     */
    public void handle(String requestedFile) throws IOException {
        //now we want to set our output stream to the client socket's output stream
        //so all data goes to client
        output = new DataOutputStream(request.getOutputStream());
        //now we want to send status code 200 + ok to signify them valid request
        output.writeBytes("HTTP/1.1 200 OK \r\n");
        //we want to attatch the date to the header incase someone wants to wireshark us xD
        //now we want to find the file which we are returning to user
        File file = null;
        for (int i = 0; i < allfiles.length; i++) {
            if (allfiles[i].getName().equals(requestedFile)) {
                file = allfiles[i];
                break;
            }
        }
        //if its a html file we write code outright in text
        if (file.getName().contains(".html")) {
            //set output mode for the clients content, text + html
            output.writeBytes("Content-type: text/html\r\n\r\n");
            //now we read through the file and write it to the client socket's output stream
            Scanner sc = new Scanner(file);
            while (sc.hasNextLine()) {
                output.writeBytes(sc.nextLine());
            }
        }
        //if its a image file
        else {
            //we want to retrieve the specific image extension, we have .jpg/.png or others
            String fileExtension = file.getName().split("/")[0].split("\\.")[1];
            //now we want to set output content type to an image of extension from above
            output.writeBytes("Content-type: image/" + fileExtension + "\r\n\r\n");
            //we want to create a buffered image of our file
            BufferedImage picture = ImageIO.read(file);
            //now we want to write our image to a byte array
            ByteArrayOutputStream pictureOutput = new ByteArrayOutputStream();
            //now our image is written to the byte array we can write directly to the client
socket's output stream
            ImageIO.write(picture, fileExtension, pictureOutput);
            //now we want to write our image to client
            output.write(pictureOutput.toByteArray());
        }
        //print a success message client side user received their file
        //get the current date for header information
```

```
        Date d = new Date();
        System.out.println("Successfully sent \"" + requestedFile + "\" to, " +
                request.getInetAddress().getCanonicalHostName() + ", on " + d.toString());
    }

    /**
     * In the case of file not existing in directory we are sending an error 404 message
     *
     * @throws IOException
     */
    public void error404() throws IOException {
        //output will now set stream to client's socket output stream
        output = new DataOutputStream(request.getOutputStream());
        //write the error 404 message
        output.writeBytes("HTTP/1.1 404 Not Found Error\r\n");
        output.writeBytes("Content-type: text/html\r\n\r\n");
        System.out.println("ERROR 404");
    }
}
```

References

Loveall, E., 2015, What are all the flags in a dig response?, [ONLINE] Available from: https://serverfault.com/questions/729025/what-are-all-the-flags-in-a-dig-response [Accessed 12 August 2018].