ETH zürich

Networked Systems
ETH Zürich — seit 2015

Communication Networks

Prof. Laurent Vanbever

Solution: Exercise 11 – DNS & HTTP

11.1 Local vs. authoritative DNS server

Perform a DNS query for `uzh.ch` using first the authoritative DNS server (`ns1.uzh.ch`) and then your local server.

Note: If using `nslookup` on Windows, specify the `-debug` flag to get the relevant information for this task. E.g.:

```
nslookup -debug <Domain Name> <DNS-Server>
```

- Compare the `ANSWER SECTION` of the responses. Can you see differences between the answers from your local DNS server and the authoritative server? Run the query to your local server multiple times to make the differences more obvious.

  Solution: The answers differ in the time to live (TTL). While the TTL is constant in the replies from the authoritative DNS server, it varies in the replies from the local server.

- What is the reason for this difference?

  Solution: The local DNS server caches replies to requests. To ensure that it does not keep outdated information in its cache, each authoritative name server attaches a TTL to its replies. The TTL tells the local DNS server how long it can store the reply in the cache and use it to reply to requests.

- As you have seen in the lecture, DNS can be used to balance the incoming load. What are the considerations one has to make when using DNS load balancing with respect to the TTL?

  Solution: With low TTLs we can ensure that we can shift the load quickly. However, low TTLs also mean that our authoritative DNS server will get many more requests.

11.2 DNS basics (Exam Question 2019)

In this task, we look at how your web browser figures out how to connect to `http://www.mail.ethz.ch/login`.

Note: You can assume the following:

- Each level of the domain hierarchy uses a separate DNS server.
- All caches are initially empty.

a) List all the DNS servers involved when resolving `http://www.mail.ethz.ch/login`. Sort them according to the order in which they receive queries if you use an *iterative* resolver (start with the one which receives the first query). Write your entries in the form "*DNS server for domain x*". *Hint:* The answer requires 7 lines or less.

Solution:

A	Local DNS resolver installed on the client
B	*DNS server for* `ch.`
C	*DNS server for* `ethz.ch.`
D	*DNS server for* `mail.ethz.ch.`
E	*DNS server for* `www.mail.ethz.ch.`
F	*DNS server for*
G	*DNS server for*

b) List all DNS queries (including their source and destination) for a *recursive* resolver to determine the IPv4 address hosting `http://www.mail.ethz.ch/login`. You can use the letters A–G from the list above to specify the DNS servers and X for the client. *Hint:* The correct answer requires 10 lines or less.

Solution:

1.	source: **X**, destination: **A**, query: `www.mail.ethz.ch.`
2.	source: **A**, destination: **B**, query: **www.mail.ethz.ch**
3.	source: **B**, destination: **C**, query: **www.mail.ethz.ch**
4.	source: **C**, destination: **D**, query: **www.mail.ethz.ch**
5.	source: **D**, destination: **E**, query: **www.mail.ethz.ch**
6.	
7.	
8.	
9.	
10.	

c) Immediately after you determined the IPv4 address for `http://www.mail.ethz.ch/login`, you want to determine the IPv4 address for `http://www.ethz.ch`. List all DNS queries (including their source and destination) for a *recursive* resolver to determine the IPv4 address hosting `http://www.ethz.ch` *assuming that*

all the responses from above are still in the caches. You can use the letters A–G from the list above to specify the DNS servers and X to for the client. *Hint:* The correct answer requires 10 lines or less.

Solution:

1.	source: **X**, destination: **A**, query: **www.ethz.ch**
2.	source: **A**, destination: **B**, query: **www.ethz.ch**
3.	source: **B**, destination: **C**, query: **www.ethz.ch**
4.	
5.	
6.	
7.	
8.	
9.	
10.	

11.3 HTTP host header

Perform a DNS lookup for `google.ch` and open `http://172.217.168.35` in your browser. What do you observe?

Now try to repeat the same process for `nsg.ee.ethz.ch` and `comm-net.ethz.ch`. Open the websites in your browser using the IP(s) from the DNS lookup. Do you see the expected websites?

Normally, one machine can host multiple websites at the same time. To distinguish which website has to be provided by the server, clients can add a so called "host header" in their HTTP request which specifies the website they want to access. You can try that yourself with the two websites from above. For example with the following commands:

```
telnet comm-net.ethz.ch 80

GET / HTTP/1.1
Host: comm-net.ethz.ch
```

Do you see another way how you could host multiple websites on the same machine? Can you see potential problems with this approach compared to the host header?

Solution: You could assign the server multiple IP addresses and link each IP address to a single website. The biggest drawback of this solution is the need for multiple IP addresses. IPv4 addresses are limited and hence expensive.

11.4 Loading a website (Exam Question 2017)

The website `www.your-shop.ch` consists of the following elements:

- HTML `www.your-shop.ch/index.html`
- Stylesheet `www.your-shop.ch/style.css`
- image `www.your-shop.ch/logo.png`
- image `images.your-shop.ch/product.jpg`
- Facebook like button `cdn.facebook.com/like.png`
- Facebook "tracking" code `www.facebook.com/track.js`
- Google "tracking" code `www.google.com/track.js`

a) Assuming that your host is configured to use a local recursive DNS server in your network and all caches are empty. List all the DNS queries that your host sends to this DNS server when you open up `https://www.your-shop.ch/` in your favorite browser.

Solution: The host will send one query per domain:

- `www.your-shop.ch`
- `images.your-shop.ch`
- `cdn.facebook.com`
- `www.facebook.com`
- `www.google.com`

b) After loading the website, you send an email to `contact@your-shop.ch` via a mail server that uses the same DNS server as your host. Does the local recursive DNS server need to run additional queries to other DNS servers if it has all the replies from the queries in the previous task in its cache? Explain why or why not.

Solution: Yes. The queries in the previous task asked for A-entries. The mail server needs an MX-entry for `your-shop.ch`, which was not queried before and is therefore not in the cache.

c) How many TCP connections would an unoptimized browser (also referred to as "naive" in the lecture) open to load `https://www.your-shop.ch/`? Briefly explain your answer.

Solution: It will require 7 TCP connections: one connection per object.

d) During your holidays in Australia, you realize that the Facebook like button loads much faster than the logo of the shop even though both images have the same size. Can you explain the reason for this and why you do not observe this behavior in Switzerland?

Solution: The Facebook button is hosted in a CDN and can be loaded fast from the Australian server of the CDN. The shop logo is not loaded from a CDN but from the Shop's webserver (which could be in Switzerland).

e) One hour later (still in Australia), you open the shop's website again. This time, the logo of the shop and the Facebook button appear at the same time. Explain **two distinct** reasons that would justify this behavior.

Solution: The logo could have been moved to a CDN or it is cached by the browser.