Eidgenössische Technische Hochschule Zürich
Swiss Federal Institute of Technology Zurich

Distributed Computing

FS 2017 Prof. R. Wattenhofer

Computer Engineering II

Exercise Sheet Chapter 4

We categorize questions into four different categories:

Quiz Short questions which we will solve rather interactively at the start of the exercise sessions.

Basic Improve the basic understanding of the lecture material.

Advanced Test your ability to work with the lecture content. This is the typical style of questions which appear in the exam.

Mastery Beyond the essentials, more interesting, but also more challenging. These questions are **optional**, and we do not expect you to solve such exercises during the exam.

Quiz

1 Quiz Questions

a) How are users authenticated in modern websites?

b) Why are two separate ports used for encrypted and unencrypted versions of a protocol?

c) Does the sender of a mail always contact the destination mailserver directly?

d) Does using an ISP provided DNS server reduce resolution time?

e) When opening `http://google.ch` in a modern browser, multiple connections to host 195.176.255.251 will be opened. Why?

Basic

2 Send me a comment

We have created a website on which users can leave comments on the lecture. It is currently hosted on the server `virt13.ethz.ch` and can be accessed using HTTP on the default port.

In this exercise we will make extensive use of the telnet client. The telnet protocol was a simple protocol that allowed the remote access to a Unix terminal with minimal overhead. While the telnet protocol is no longer in use, being unencrypted and insecure, the telnet client remains a popular tool to experiment with human readable protocols since it simply opens a TCP connection and accepts input which is then transferred over the raw TCP connection and also accepts incoming data and prints it on screen. We suggest to either use the `telnet` command or the `netcat` client on Linux, or PuTTY on windows. Due to a difference in what constitutes a newline in various operating systems, please make sure that the client is configured to send both a carriage return (CR) and a line feed (LF). This should already be the case with `telnet`, `netcat` might require the `-C` option, and PuTTY requires the options *Implicit CR in every LF* and *Implicit LF in every CR* to be enabled. Also make sure to use the `Raw` connection type in PuTTY.

a) Visit the website with your browser and try to explain what is happening in the background in order to display the page. Most modern browsers have developer tools to inspect the currently loaded page and trace the HTTP requests that the browser issued. In Firefox and Google Chrome the developer tools can be opened pressing `Ctrl+Shift+I`. Make sure to have the developer tools open while loading the page, otherwise the request trace may be incomplete.

b) Now retrieve the HTML page with telnet, by connecting to port 80 on the server and issuing an HTTP request manually.

c) The website is not yet fully functional, e.g., it does not have a form to enter new comments, however the server already accepts new comments if they are sent to the `/` path using the `PUT` method. Use telnet to add a new comment to the page.

3 Send me a mail

A website may have many interfaces and interact with any number of services. To give our users more flexibility we also accept mail messages. The mail server is using SMTP on port 25 on the same server as the website.

a) Using a mail client to send mails to `somebody@virt13.ethz.ch` will not work, can you guess why?

b) Use telnet and connect to the SMTP port to send us a comment.

Advanced

4 DNS

In this exercise, we want to get more familiar with the Domain Name System (DNS). We will use the popular `dig` command-line tool. `dig` is available on Linux, MacOS and Windows. You can use your own PC, or go to one of the computer rooms to solve this exercise.

4.1 Getting Started

a) Use the `dig` tool to find the IP address of `disco.ethz.ch`.

b) In the `dig` answer, you can see an entry with record type `CNAME`. What does the `CNAME` record mean?

c) There is also a record of type `A` in the `dig` answer that seems to refer to another server, namely `disco01-srv.ethz.ch`. What is the meaning of this record? What is the difference between a `A` and `AAAA` record?

d) Find the mail servers that are responsible for the `ethz.ch` domain. What does the number in front of the mail servers' DNS names mean?

e) Run the following commands:

```
dig +domain=ethz.ch disco
dig +domain=ethz.ch disco.
```

What IP addresses are returned? Why do the results differ?

4.2 DNS Queries

A name resolution consists of recursive and iterative queries. First, a client (e.g. a browser) asks its local stub resolver if it knows the IP address that belongs to a given domain name. If the stub resolver does not find the answer in its cache, it will issue a recursive query to a local DNS server. If the local DNS server does not have the answer in its cache, the stub resolver will then issue an iterative query to the lowest known DNS server in the hierarchy. If the local DNS server does not know any other DNS servers in the hierarchy of the URL, it will ask a DNS root server. The DNS root server usually does not have an authoritative answer, so it returns a list of DNS servers that might know more. The local DNS resolver then asks one of the DNS servers on the list, and so on, until it receives an authoritative answer, i.e., the IP address we were looking for.

Usually, this is done automatically, and the client just receives the final result of the recursive query. In order to better understand DNS, you will now manually perform such a recursive query and trace every step.

a) Use the `dig` tool to get a list of all root servers. What command do you use?

b) Use `dig` to issue a non-recursive query to one of the root servers to ask for the IP address of `disco.ethz.ch`. Explain the result. Make sure to use the right command, and not to accidentally use a recursive query.

c) Now, use `dig` to issue further non-recursive queries until you find the IP address of `disco.ethz.ch`.

d) Now, issue a standard recursive query to your standard DNS server and verify that the results are the same.

4.3 DNS Caching

In order not to flood the root servers with DNS queries and to improve response times, DNS servers can cache answers.

a) Find your default nameserver's IP address.

b) Issue a non-recursive query for a website of your choice to your default nameserver. Was the website cached? How can you know?

- If yes: Find a website that is not in your default nameserver's cache and repeat the non-recursive query.
- If no: Perform a recursive query for the same website. Now, perform the non-recursive query again. Is the answer in the cache now?

c) How do the query times differ?