Computer Networks
Jacobs University Bremen
Dr. Jürgen Schönwälder

Course: CO20-320301
Date: 2018-05-08
Due: 2018-05-15

CN 2018 Problem Sheet #6

Problem 6.1: *http/2* (2+1+2 = 5 points)

The specification of HTTP/2 was published as RFC 7540 in May 2015. Many big web sites seem to support HTTP/2 today and hence it is time to look a bit into the details of HTTP/2. Please consult RFC 7540 to find the answers to the following questions and point to the appropriate sections of the RFC when you justify your answers. (You can of course search on the web but be warned that not everything you find on the web is correct — and you often do not find the pointer to the correct section in the specification.)

a) How do client and server determine whether they can make use of HTTP/2? Explain.

b) HTTP/2 supports multiple streams that are multiplexed over a single (secure) transport connection. Is flow control done at the stream level or at the connection level? Explain. What is the initial value of the flow-control window?

c) HTTP/2 supports server push. Which problem does server push solve and how does server push work? Describe an example exchange. Are there any special rules for stream identifiers?

Problem 6.2: *HTTP/1.1 caching* (1+1 = 2 points)

HTTP introduced the `Cache-Control` header field in RFC 7234 to let servers control the caching of responses and to let clients request a certain behaviour of caches. Answer the following two questions.

a) What is a cache validator and what is the difference between a strong and a weak validator? Why are only pages cacheable that have a cache validator?

b) What is the consequence of the response header:

```
Cache-Control: max-age=3600, must-revalidate
```

What is the consequence of the request header:

```
Cache-Control: max-stale=300
```

Problem 6.3: *representational state transferl (rest)* (1+1+1 = 3 points)

Representational State Transfer (REST) is an architectural style that builds on top of HTTP. The concepts were defined by Roy Fielding in his PhD thesis. Read about REST and answer the following questions:

a) What are the six architectural constraints?

b) REST APIs are often able to return representations of a resource in different formats. How does a client request a certain representation format?

c) What is the meaning of “Hypermedia as the Engine of Application State (HATEOAS)”?