S-38.3159 Protocol Design Jörg Ott, Varun Singh

Exam 2 September 2014

Please write readably.

There are three classes of questions: (a) expecting (relatively) short answers, (b) expecting more elaborate answers, and (c) a small design task. The questions are marked accordingly.

Questions:

1. [6p, a] Which basic two options for error repair do you have? What are their tradeoffs?
2. [6p, a] Using multicast makes data authentication more complex than it is with a unicast protocol. Why?
3. [6p, a] Which options for limiting resource consumption in the network are available to a protocol designer? Sketch one approach with and one without network support.
4. [6p, b] "Make or take" decisions are often an important design alternative in protocol design. Discuss two reasons in favor of each choice. Give one example for a protocol aspect where a make decision is a good idea and one example where a take decision is preferred; in each case, briefly explain why.
5. [6p, b] (a) Describe four aspects of scalability.

   (b) For one of them, discuss through which mechanisms scalability can be achieved.

   (c) For (another or the same) one, discuss under which circumstances the scalability requirements may be relaxed for a protocol or system design.

   For (b) and (c), explain why.
6. [6p, a] Which advantage(s) do you get when encoding numbers in plain text (e.g., in the HTTP Content-Length header)? Which implementation issue may arise?
7. [6p, b] Compare topological (e.g., hierarchical) addressing and flat address spaces? What are their respective advantages and disadvantages?
8. [12p, c] The following figure shows the UDP header defined in RFC 768:

```
 0      7 8     15 16    23 24    31
+--------+--------+--------+--------+
|     Source      |   Destination   |
|      Port       |      Port       |
+--------+--------+--------+--------+
|                 |                 |
|     Length      |    Checksum     |
+--------+--------+--------+--------+
|
|          data octets ...
+---------------- ...
```

   Sketch which additional fields and corresponding protocol mechanisms (local algorithms and packet flow rules) would be needed to add a minimum of congestion control to UDP. (Remember that there are many alternative ways to perform congestion control; pick your favorite.)