Mobile Communication
Summer term 2016

Practical Assignment Sheet # 2

Matthias Frank, Fabian Rump

- Release date: **May 24th, 2016**
- The practical assignment sheet must be solved in groups of two students, e.g. the groups that have already been registered for PA1.
- Submission of solutions:
  - **Not later than June 28th, 2016 23:59 CEST.**
  - Via e-mail to assignment4@cs.uni-bonn.de
  - The e-mail subject must be `MoCo-PA Solution` followed by your matriculation numbers.
  - The attachment of the e-mail must be a single ZIP file containing all your solutions. The file must be called PA2.zip.
  - After unpacking there have to be three directories *scripts*, *plots* and *doc*. Place all your scripts in the *scripts* directory, PDF files of your plots in the *plots* directory and **one** PDF file with your documentation written out in full (again including all plots you are interpreting!) in the *doc* directory.
- General information about practical exercises:
  - Successful participation in **both** practical exercises is mandatory for the admission to the exam.
  - The practical exercises are to be **solved in groups of two**. In order to discourage abusive copying, all solutions participating in a copy (in part or in whole) will result in a failed exercise.
  - If anything is still unclear, post on the mailing list or ask us.

PA2: Performance Measurements in Ad-Hoc Networks

The essential concepts of wireless ad-hoc networks have been presented in the MoCo lecture. Now, we want to study the performance of such networks in a real-world scenario. To do so, we have set up a test network within the LBH building. As can be seen in Figure 1, the network consists of 6 wireless routers placed in different rooms across the 2nd floor of the building.

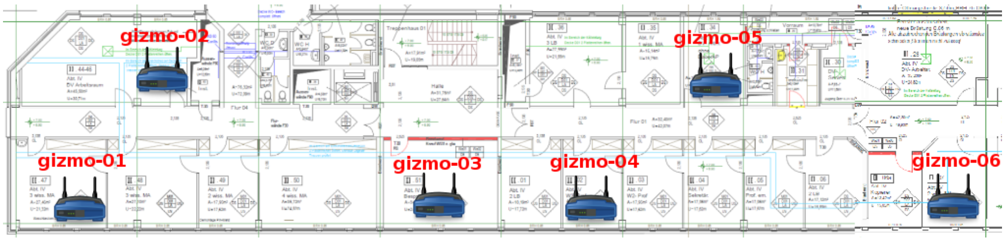

Figure 1: Topology of the OLSR test network

Inside the wireless network, each router is assigned an IP address from the `10.0.0.0/24` range: Router *gizmo-01* has address `10.0.0.1`, *gizmo-02* has address `10.0.0.2`, . . .

All routers use the OLSR protocol to discover their neighbors and to distribute topology information throughout the network. To measure the strength of the individual wireless links, the ETX metric is used. Information about the OLSR protocol can be found in the MoCo lecture slides. Information about ETX is available online, e.g. here: `http://www.olsr.org/docs/README-Link-Quality.html`

Network topology information

The current status data of the OLSR daemon on each of the routers can be requested in a JSON file format **(inside the computer science network or via VPN!)**:

- *gizmo-01* – `http://gizmo-01.informatik.uni-bonn.de/status.json`
- *gizmo-02* – `http://gizmo-02.informatik.uni-bonn.de/status.json`
- *gizmo-03* – `http://gizmo-03.informatik.uni-bonn.de/status.json`
- *gizmo-04* – `http://gizmo-04.informatik.uni-bonn.de/status.json`
- *gizmo-05* – `http://gizmo-05.informatik.uni-bonn.de/status.json`
- *gizmo-06* – `http://gizmo-06.informatik.uni-bonn.de/status.json`

The JSON file contains the following entries:

- `systemTime` – Unix timestamp of the last status update
- `neighbors` – Information about the neighbors of each router
- `links` – Information about the links to the neighbors
- `topology` – Topology information received from other routers
- `routes` – Generated routing table for all reachable nodes

Measurement data

In order to study the performance of the wireless ad-hoc network, two experiments are executed:

- **HTTP throughput measurement**
  Available at `http://gizmo-01.informatik.uni-bonn.de/http.txt` **(inside the computer science network or via VPN!)**

  Router *gizmo-01* (`10.0.0.1`) repeatedly downloads a 4 megabyte file from a webserver running at *gizmo-06* (`10.0.0.6`). For each download attempt, a log entry of the following form is created:

```
1463735587
Bytes: 4194304
Time: 30.987
```

  The first line of the log entry is a unix timestamp and marks the time where the download has been started. The second line is the number of bytes downloaded from the server and the third line is the total time taken for downloading the file. Note that the download will be automatically aborted after about 60 seconds, i.e. the number of downloaded bytes will not always be equivalent to 4 megabytes.

- **End-to-end delay and loss measurement**
  Available at `http://gizmo-01.informatik.uni-bonn.de/ping.txt` **(inside the computer science network or via VPN!)**

  Router *gizmo-01* (`10.0.0.1`) repeatedly measures the end-to-end packet loss and latency to *gizmo-06* (`10.0.0.6`) using the `ping` utility. The log entries of the ping measurements look like this:

```
1463736248
PING 10.0.0.6 (10.0.0.6): 56 data bytes

--- 10.0.0.6 ping statistics ---
50 packets transmitted, 44 packets received, 12% packet loss
round-trip min/avg/max = 6.236/10.683/90.880 ms
```

  Again, the first line is a unix timestamp and marks the time where the ping measurement has been started. It is followed by the statistics output of the ping utility, which contains information about packet loss and round-trip-time.

Exercise 1: Collection of toplogy information and measurement data [30 Points]

Using a programming language of your choosing, implement a program that fetches topology information and measurement data from the URLs given above, processes the contained data and stores all information in a suitable file format for further analysis and processing.

Specifically, your program should …

1. **access the URLs in a suitable interval** (e.g. repeat each 30 seconds) and download the contained data for further processing.
2. **parse and extract the contained information** according to the file formats described above.
3. **transform the extracted information to a data format of your choosing**. The data format must contain all relevant parts of the topology information and measurement data along with the temporal information, i.e. the timestamps of the topology status data and the single measurements.
4. **store the transformed data on the computer's hard disk** so that it can easily be used for further analysis.

Explain the structure of your data format and give a brief description of how to compile and run your program.

Notes:

- The measurement data files will grow over time since new measurements are appended to the end of the file. Hence for subsequent requests to the file, be sure to filter out duplicate measurement results in your program.
- Additionally, the measurement results are not persisted on the server. Thus, old measurement values may be dropped from the data files at any time.

Exercise 2: Performance analysis [70 Points]

Using the program implemented in Exercise 1, collect the topology information and measurement data of **at least a 2 hour timeframe**. Then, using the collected data, perform an in-depth analysis of the network performance. This analysis should cover the following aspects:

1. Find a way to detect and visualize changes to the network topology over time. This should not only include changes to the used routes, but also cover all significant changes to the quality of the individual links. Give reasons for your solution and describe your findings.
2. Evaluate the HTTP measurement data. Provide a plot of the data throughput and the total number of transferred bytes over time. Can you observe any patterns in this plots? Describe your findings and try to correlate them to the results of the topology analysis.
3. Analyze the ping measurement data and provide plots of the end-to-end packet loss and RTT over time. Additionally, do a statistical evaluation of the individual measurement values. Based on your findings, state whether or not such a network is suitable for the following application areas: Multiplayer computer games, Voice-over-IP telephony, E-Mail

Notes:

- Useful tools to analyze the data and create the plots are R, gnuplot, SPSS, Matlab, Octave.
- Try to reuse the scripts and tools that you have developed for PA1 where possible.

Total ___/100

You need 50 points or more to pass this practical assignment sheet!

Good luck and have some fun!