Assignment: From Traffic Measurements to Conclusions

S-38.3184 Network Traffic Measurements and Analysis

April 1, 2015

Abstract

In this assignment you will perform the whole work flow of measurements and analysis starting from actual measurements and ending with analysis results and conclusions.

1 General guidelines

The assignment consists of two data sets both of which should be analysed. Please note that the assignment **may require quite a lot of work**, especially if you are not familiar with computer oriented analysis methods and tools. Take this into account when planning your schedule.

1.1 Report: IMPORTANT!

You should report your analysis by detailing your results in a *written report*. Submission of the report consists of two phases:

- You should return the report **personally** to the course assistant for discussion, evaluation and feedback. Tentatively the discussions take place 15.4.-30.4. , but more information will be provided via noppa. *At least 50% of the assignment **must** be completed before the meeting* (otherwise, the assignment will be considered as rejected), but it is highly recommended that you have completed the whole assignment at this stage.
- **After the feedback** the report should be revised according to the feedback and returned to the course mailbox by **noon (12:00) 15.5.2015 (Friday)**. Late submissions **are considered as rejected**.

Include your name and student number in the title page of the report.

The report should contain detailed explanations on what is done (supplemented with actual commands used, e.g., to draw a figure, if appropriate). Uncommented commands or code alone are not sufficient. When you are asked to plot or visualise something *make sure that your figure really visualises something*, i.e., the task is *not* just plot the numbers, but to select the appropriate plot, units and scales (e.g. logarithmic) so that the resulting figure is as informative as possible.

Minimum requirements for each data set are defined in the following categories: Preprocessing, analysis, conclusions. **Address all the numbered questions appropriately and in that order**. Organise your report clearly, using sections for data sets and subsections for preprocessing, analysis, and conclusions for each data set. It is recommended that each plot is supplemented with a short description. Pay attention to the conclusions as they are emphasized in grading.

1.2 Assessment

Please pay attention to the deadline, late submissions will be automatically considered as rejected. Note also that the assignment must be shown to the course assistant before the submission and at this state it must be at least 50% complete. The assignment will be graded with a (continuous) scale 1-5 (grade < 1

is rejected) which has a 30% weight in the final grading of the course. Both the assignment and the exam *must* be completed successfully (i.e. you should get a grade ≥ 1 from both) in order to pass the course.

The assignment is individual work. You may co-operate with others by discussing the tasks, but all output should be produced by yourself and you have to present your report to the assistant by yourself. The assignment grade consists of

- Correct and insightful answers (weight 80%)
- Readability, clarity and style of the report (weight 20%)

1.3 Useful information

Some of the tools (e.g. Coral Reef, tstat) that are needed (or otherwise useful) for analysis are provided in Aalto Linux workstations.[1] To have those available you should run one of the following commands (depending on your shell, check with `echo $SHELL` command)

```
tcsh-prompt% source /work/courses/T/S/38/3184/general/use.csh
bash-prompt$ source /work/courses/T/S/38/3184/general/use.sh
```

Documentation for Coral Reef tools:
`http://www.caida.org/tools/measurement/coralreef/doc/doc/`

From that site you can also find Coral Reef software package in case you want to install it to your computer. Some tools available in Coral Reef are useful in analysing traces, check documentation. Those include, but are not limited to:

crl_to_pcap convert and anonymise packet traces

crl_flow summarises packet data to flows

t2_top sorts flow data by criteria

crl_bycountry lists traffic by country

Also tool `tstat` can be useful. Its documentation can be found from
`http://tstat.tlc.polito.it/docs.php`

1.4 Support

The assignment is meant to be individual work, but there are two kinds of support available. First, an interactive exercise class will be arranged on **the 31st of March at 10-12 in L310** where the assignment is introduced with some hints on how to approach the problems.

Second, we will open an IRC channel (#s-38.3184) where you can post questions on the assignment. There will be a log (link at noppa) so that all questions and answers will remain visible to everybody. The questions should address general issues only, i.e., clarifications on tasks, hints on tools, etc. No code debugging, please. Channel will be monitored every now and then during office hours, so the answer may not be immediate.

network IRCnet

[1] If you login remotely using SSH, e.g., to brute.aalto.fi, then in order to access the `/work` directories you need to run first `kinit` (Kerberos ticket).

channel #s-38.3184

If you are unfamiliar with IRC, following are simple instructions that work on Aalto Linux environment.

1. launch command `irssi` from command shell
2. give commands `/server irc.cs.hut.fi` and `/join #s-38.3184`
3. you can end session with `/quit`

Note that you can use the same forum to give peer support, you are allowed to answer to the questions too.

If you have a larger set of information to accompany your question, do not send it to IRC channel but ask for alternative means on the channel.

2 Data set I: Flow data

Data set I consists of anonymised flow measurements from an access network. A sample of users have been selected for the data collection. The time stamps on the flows are given in terms of Unix time[2].

2.1 Acquiring flow data

Flow data is available at directory `/work/courses/T/S/38/3184/general/trace` (please note the file sizes!). After sourcing use script, directory is in variable `$TRACE`.

2.2 Data preprocessing

If necessary, preprocess the data so that you can use it in your analysis software.
In your report, include

1. Commands or code that is used in preprocessing.
2. Short samples (10 lines or so) of the distilled data.

2.3 Analysis

Analyse the data set carefully. The minimum requirements are detailed below, but additional plots and insights are welcome. Each plot should be accompanied with a short description.

1. Plot traffic volume as a function of time at least on two (sufficiently different) time scales
2. Visualise flow distribution by port numbers
3. Plot origin-destination pairs by both by data volume and by # of flows (Zipf-type plot)
4. Compute the aggregate data volume for each user and draw a histogram to visualise distribution of user-aggregated data
5. Plot flow length distribution, its empirical cumulative distribution function and key summary statistics
6. Fit a distribution for the flow lengths and validate the model.

2.4 Conclusions

Draw your conclusions on the data. In particular,

1. Discuss the traffic volume at different time scales. Are there any recognisable patterns?
2. What are the 5 most common applications (study the port numbers)?
3. What kind of users there are in the network? Speculate on what kind of network this network could be based on traffic volumes and user profiles.

You may use additional visualisation to support you claims if necessary.

[2] `http://en.wikipedia.org/wiki/Unix_time`

3 Data set II: Packet capture

Data set II is obtained by packet capture.

3.1 Acquiring packet capture data

The recommended way get the packet trace is to carry out your own measurements. You will need to use your own computer (or a network you have an access to, and the right to measure) to get the data.

For about all systems, you can use Wireshark program[3]. Step-by-step instructions can be found from

`http://wiki.wireshark.org/CaptureSetup`

Note that you will need administrative privileges.

In case of Linux (or *BSD) system the command below (needs root priviledges) is the alternative way to capture traffic destined and sent by local computer. The packet header data is stored into file `myown.pcap`.

```
sudo tcpdump -p -s 0 -w myown.pcap -p -i eth0
```

The measurement period should be at least two hours long, while a day-long trace is better. Depending on your traffic volumes, of course. Use your computer (normally) while measuring, empty data is not interesting.

If you do not have a personal computer that you could use for measurements, you can ask Markus Peuhkuri to provide you an access to computer that you could be used to record a session. As a last resort, you can use DEC traces from Internet Traffic Archive:

`http://ita.ee.lbl.gov/html/contrib/DEC-PKT.html`

In your report

1. Write clearly what trace file you are using (date, time, measurement setting or file name if using the DEC-traces).
2. Provide a short sample (10 lines or so) of the data you are using.

3.2 Data preprocessing

After you have the raw packet data, you need to convert it to a suitable format. The data will be analysed both at packet level and at flow level.

At first phase, you can anonymise your traces using `crl_to_pcap` command. If you choose to anonymise the trace, use the anonymised trace then consistently in all analyses to avoid confusion.

Three (3) data sets will be distilled from the raw data. We refer to these as ds2.1, ds2.2, and ds2.3, respectively. In your report, include

1. Commands or code that is used in preprocessing for each case.
2. Short samples (10 lines or so) of the distilled data in each case (for ds2.3, one connection summary is enough).

The data sets are described next in more detail.

[3] `http://www.wireshark.org`

3.2.1 Cleaning packet data: ds2.1

Have a look at the analysis section 3.3, to get an idea which information on individual packets are needed and clean the data to contain only the relevant columns.

3.2.2 Converting packet trace to flow data: ds2.2

To produce flow data, use `crl_flow` from CoralReef package. Use time-out of 60 seconds. Flow output includes comments and statistics in lines starting with hash mark (#).

Alternatively, you can code your own script to extract the flow data (optional!)

3.2.3 TCP connection statistics: ds2.3

You can use `tcptrace` command to produce statistics from TCP connections.

```
tcptrace -l -r -n --csv myown.pcap > myown-tcp.csv
```

Command produces statistics about every TCP connection seen. You get more verbose output if you omit `--csv` option (try it, csv may just be easier to parse). You can find more details from `man tcptrace`.

3.3 Analysis

Analyse the data sets carefully. In each case the minimum requirements are detailed in below, but additional plots and insights are welcome. Each plot should be accompanied with a short description.

3.3.1 Packet data: ds2.1

1. Visualise packet distribution by port
2. Plot traffic volume as a function of time at least on two different time scales
3. Plot packet length distribution (use bins of width 1 byte), its empirical cumulative distribution function and key summary statistics

3.3.2 Flow data: ds2.2

1. Visualise flow distribution by port
2. Visualise flow distribution by country
3. Plot origin-destination pairs by both by data volume and by flows (Zipf-type plot)
4. Plot flow length distribution, its empirical cumulative distribution function and key summary statistics
5. Fit a distribution for the flow lengths and validate the model
6. Compare the number of flows with 1, 10, 60, 120 and 1800 second timeouts.

3.3.3 TCP connection data: ds2.3

For the TCP connection statistics we are this time interested in retransmissions. Study the association of retransmissions to

1. Round-trip times and their variance
2. Total traffic volume during the connection (you get the volume from 3.3.1)

3.4 Conclusions

Draw your conclusions on the data. In particular,

1. Discuss the traffic volume at different time scales. Are there any recognisable patterns?
2. What are the 5 most common applications (study the port numbers)?
3. Compare the above results to those of data set I
4. Discuss the differences of flow and packet measurements in the example case
5. Discuss your findings on retransmissions

You may use additional visualisation to support you claims if necessary.