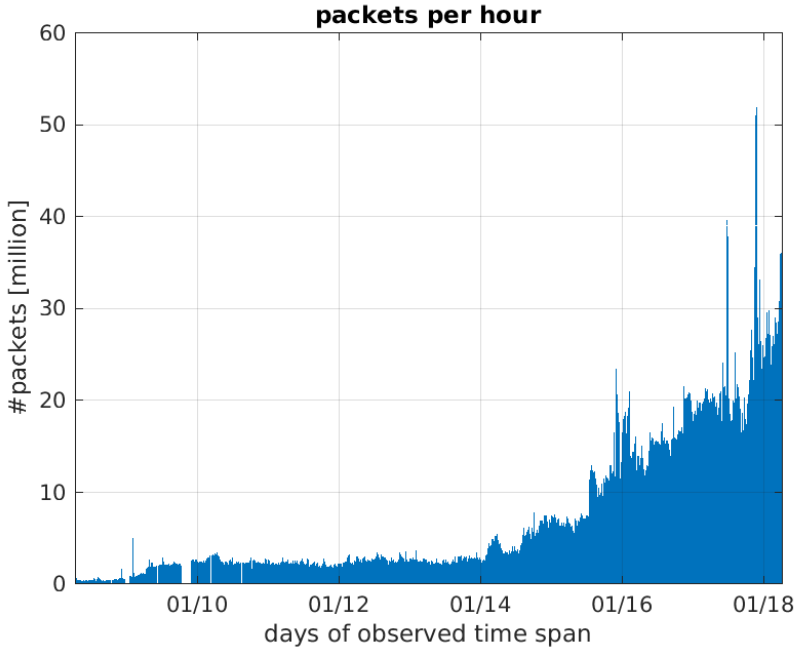
# Network Security
# 389.159 - SS 2018
# Lab Exercise 3 & Lab Exercise 4

TEAM 02
Corentin Bergès (11741629) (066 506)
Christoph Echtinger-Sieghart (00304130) (066 938)

June 4, 2018

## 1 Lab Exercise 3

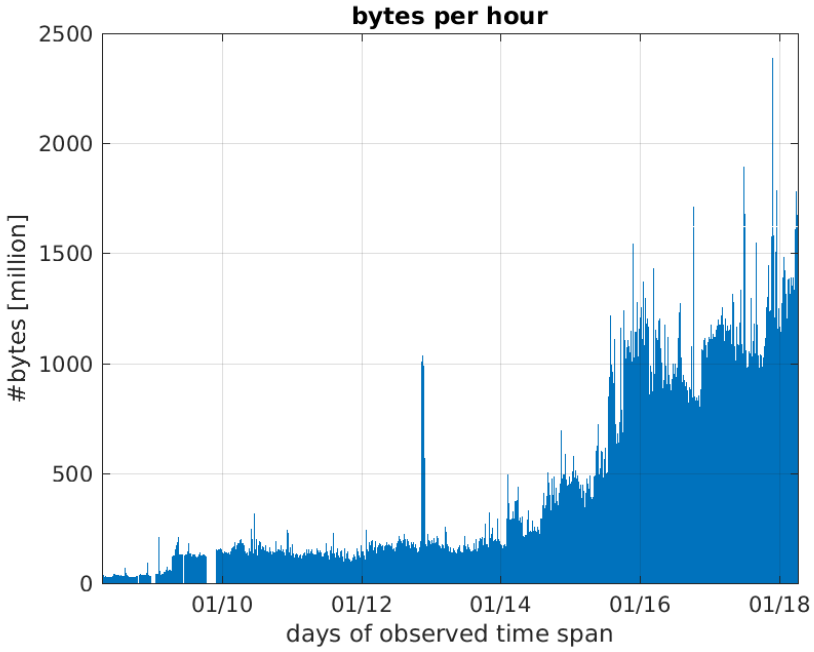
### 1.1 rep-10




### 1.2 rep-11

### 1.3 rep-12

Listing **??** shows the code used to obtain the results.




### 1.4 rep-13

### 1.5 rep-14

### 1.6 rep-15

### 1.7 rep-16

### 1.8 rep-17

### 1.9 rep-18

### 1.10 rep-19

### 1.11 rep-20

### 1.12 rep-21

### 1.13 rep-22

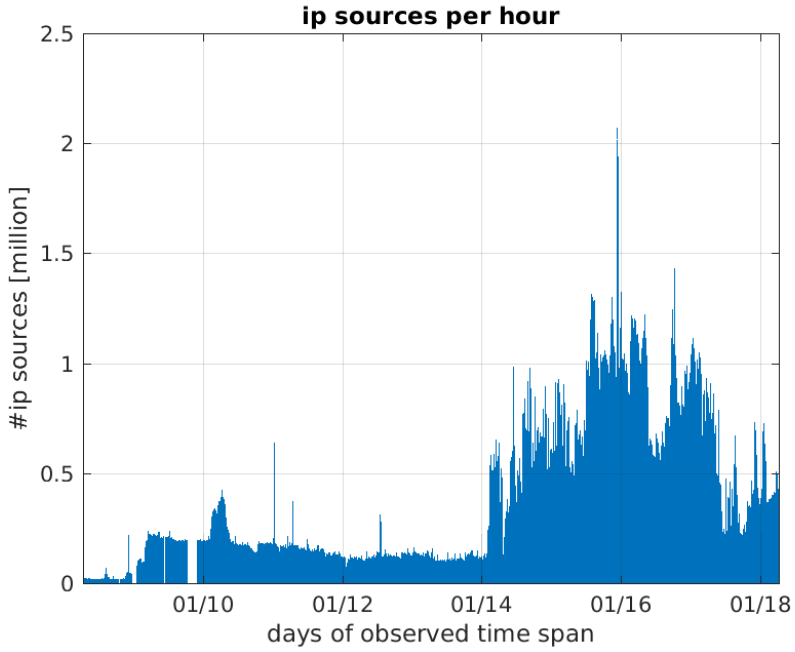
### 1.14 rep-23

Listing 1: Command used to obtain IP address

```
team02@pc01:~$ ip address show dev em1
```

**Port 113**

```
IP 192.168.83.20.1073 > 192.168.83.33.113: Flags [S], seq 0, win 8192, length 0
IP 192.168.83.33.113 > 192.168.83.20.1073: Flags [R.], seq 0, ack 1, win 0, length 0
```




# 2 Lab Exercise 4

## 2.1 rep-24

## 2.2 rep-25

## 2.3 rep-26

## 2.4 rep-27

## 2.5 rep-28

## 2.6 rep-29

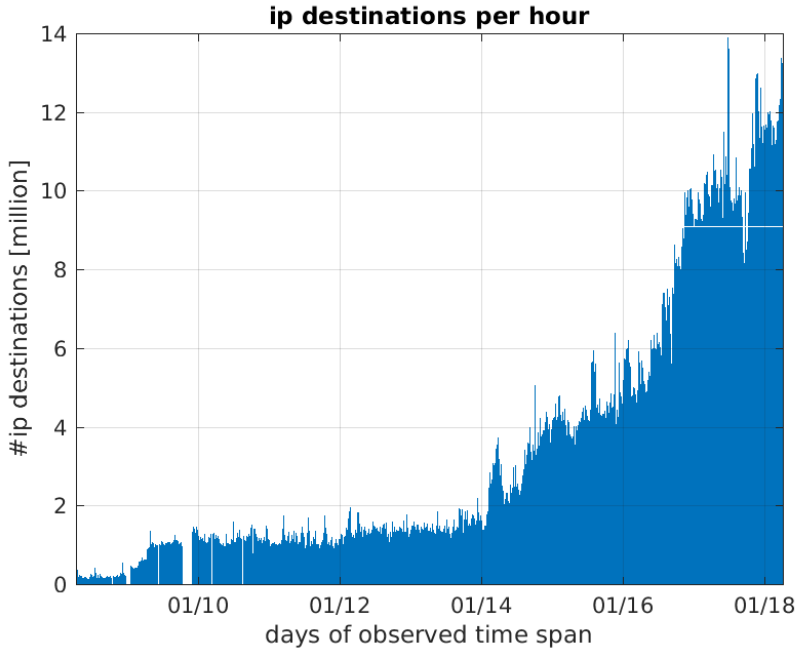
## 2.7 rep-30

# A Matlab code

```
function team02_rep10
% rep-10
    [timestamps, bytes, packets, ip_s, ip_d] = read_custom_csv('global_last10years.csv');

    function save_stem_plot(data, my_title, y_label, filename)
    % Do a stem plot of data in millions and write it to filename.png
        set (gca, 'fontname', 'Helvetica', 'fontsize', 20)
        figure
        stem(timestamps, data/10^6, 'marker', 'none')
        datetick('x', 'mm/yy');
        xlabel('days of observed time span');
        ylabel(y_label);
        title(my_title);
        grid on
        set(gca, 'layer', 'top');
        xlim([min(timestamps) max(timestamps)]);
        saveas(gcf, filename, 'png')
    end

    save_stem_plot(bytes, 'bytes per hour', '#bytes [million]', 'plots/rep_10_2');
    save_stem_plot(packets, 'packets per hour', '#packets [million]', 'plots/rep_10_1');
    save_stem_plot(ip_s, 'ip sources per hour', '#ip sources [million]', 'plots/rep_10_3');
    save_stem_plot(ip_d, 'ip destinations per hour', '#ip destinations [million]', 'plots/rep_10_4')
```

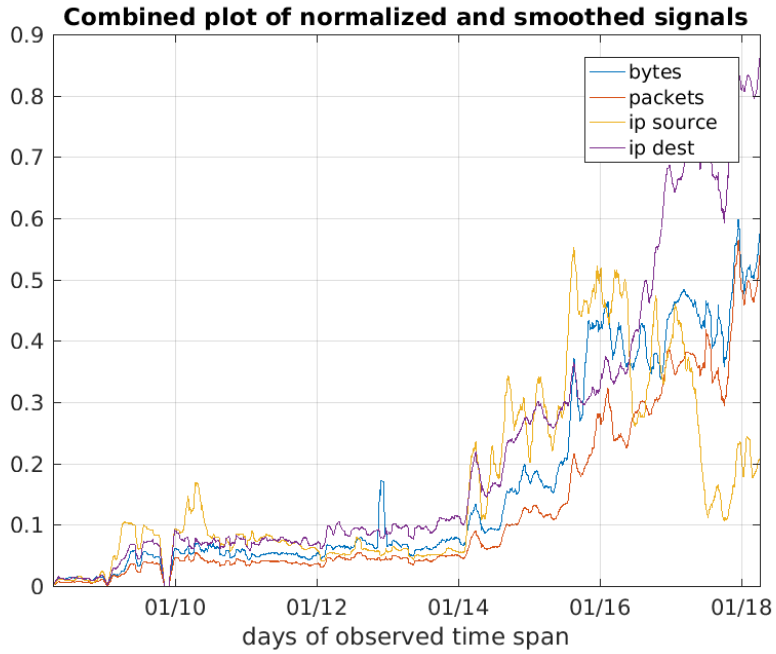



```
    % optional part

    function result = smooth_filter(data)
    % Moving averages filter for data
        window_size = 30;
        b = (1 / window_size) * ones(1, window_size);
        a = 1;
        % 1-D digital filter
        result = filter(b, a, data);
    end

    smooth_bytes = smooth_filter(bytes / unique(max(bytes)));
    smooth_packets = smooth_filter(packets / unique(max(packets)));
    smooth_ip_s = smooth_filter(ip_s / unique(max(ip_s)));
    smooth_ip_d = smooth_filter(ip_d / unique(max(ip_d)));

    figure
    plot(...
        timestamps, smooth_bytes, '-', ...
        timestamps, smooth_packets, '-', ...
        timestamps, smooth_ip_s, '-', ...
        timestamps, smooth_ip_d, '-' ...
    );
    legend('bytes', 'packets', 'ip source', 'ip dest');
    datetick('x', 'mm/yy');
    xlabel('days of observed time span');
    title('Combined plot of normalized and smoothed signals');
    grid on
    set(gca, 'layer', 'top');
    xlim([min(timestamps) max(timestamps)]);
    saveas(gcf, 'plots/rep_10_optional', 'png')
end
```


Combined plot of normalized and smoothed signals
bytes
packets
ip source
ip dest
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0
01/10
01/12
01/14
01/16
01/18
days of observed time span