Speed

March 31, 2024

[1]:
```
import os

#       "./files"      files
folder_path = './glog_files'

#  os.listdir()
file_names = os.listdir(folder_path)

#
file_paths = ["glog_files/"+  file for file in file_names if os.path.isfile(os.
 ↪path.join(folder_path, file))]

file_paths
```

[1]:
```
['glog_files/send_pcap_log_100000_20240328_150947.txt',
 'glog_files/send_pcap_log_100000_20240328_163451.txt',
 'glog_files/send_pcap_log_10000_20240328_123904.txt',
 'glog_files/send_pcap_log_10000_20240328_145136.txt',
 'glog_files/send_pcap_log_1000_20240328_092832.txt',
 'glog_files/send_pcap_log_1000_20240328_152443.txt',
 'glog_files/send_pcap_log_1000_20240329_104307.txt',
 'glog_files/send_pcap_log_1000_20240329_105515.txt',
 'glog_files/send_pcap_log_1000_20240331_103106.txt',
 'glog_files/send_pcap_log_1000_20240331_172059.txt',
 'glog_files/send_pcap_log_1000_20240331_185113.txt',
 'glog_files/send_pcap_log_100_20240328_154020.txt',
 'glog_files/send_pcap_log_5000_20240328_101717.txt',
 'glog_files/send_pcap_log_5000_20240328_113135.txt',
 'glog_files/send_pcap_log_5000_20240331_082427.txt',
 'glog_files/send_pcap_log_5000_20240331_083441.txt',
 'glog_files/send_pcap_log_5000_20240331_084454.txt',
 'glog_files/send_pcap_log_5000_20240331_085506.txt',
 'glog_files/send_pcap_log_5000_20240331_090521.txt',
 'glog_files/send_pcap_log_5000_20240331_100049.txt',
 'glog_files/send_pcap_log_5000_20240331_101601.txt',
 'glog_files/send_pcap_log_5000_20240331_171118.txt',
 'glog_files/send_pcap_log_tk_20240330_211305.txt',
 'glog_files/send_pcap_log__1000_20240329_102807.txt']
```

[2]:

```python
#
total_packets = 0
total_bytes = 0
rates_mbps = []

#
for file_path in file_paths:
    with open(file_path, 'r') as file:
        for line in file:
            if "Actual:" in line:
                parts = line.split()
                packets = int(parts[1])
                #
                bytes_sent = int(parts[3].replace("(", "").replace(")", ""))
                total_packets += packets
                total_bytes += bytes_sent
            elif "Rated:" in line:
                parts = line.split(",")
                #    Mbps
                rate_mbps = float(parts[1].strip().split()[0])
                rates_mbps.append(rate_mbps)

#
average_rate_mbps = sum(rates_mbps) / len(rates_mbps)
max_rate_mbps = max(rates_mbps)

(total_packets, total_bytes, average_rate_mbps, max_rate_mbps)
```

[2]:

```
(2060915167, 890044759668, 376.1612823504462, 1184.23)
```

[3]:

```python
#
results = []

#
for file_path in file_paths:
    with open(file_path, 'r') as file:
        local_total_packets = 0
        local_total_bytes = 0
        local_rates_mbps = []
        for line in file:
            if "Actual:" in line:
                parts = line.split()
                packets = int(parts[1])
                bytes_sent = int(parts[3].replace("(", "").replace(")", ""))
                local_total_packets += packets
                local_total_bytes += bytes_sent
            elif "Rated:" in line:
                parts = line.split(",")
                rate_mbps = float(parts[1].strip().split()[0])
                local_rates_mbps.append(rate_mbps)
        #
        average_rate = sum(local_rates_mbps) / len(local_rates_mbps) if␣
↪local_rates_mbps else 0
        max_rate = max(local_rates_mbps) if local_rates_mbps else 0
        results.append((file_path, local_total_packets, local_total_bytes,␣
↪average_rate, max_rate))

results
```

[3]:

```
[('glog_files/send_pcap_log_100000_20240328_150947.txt',
  87548742,
  37677643189,
  471.6277728285087,
  1169.37),
 ('glog_files/send_pcap_log_100000_20240328_163451.txt',
  87548742,
  37677643189,
  446.544521158129,
  978.14),
 ('glog_files/send_pcap_log_10000_20240328_123904.txt',
  82800573,
  36918819854,
  128.8326008968609,
  315.99),
 ('glog_files/send_pcap_log_10000_20240328_145136.txt',
  87548742,
  37677643189,
  454.25692650334105,
  1116.78),
 ('glog_files/send_pcap_log_1000_20240328_092832.txt',
  87548742,
  37677643189,
  126.00082405345209,
  472.07),
 ('glog_files/send_pcap_log_1000_20240328_152443.txt',
  87548742,
  37677643189,
  460.04443207126957,
  993.13),
 ('glog_files/send_pcap_log_1000_20240329_104307.txt',
  87548742,
  37677643189,
  409.7998440979957,
  954.93),
 ('glog_files/send_pcap_log_1000_20240329_105515.txt',
  87548742,
  37677643189,
  407.2076614699332,
  997.59),
 ('glog_files/send_pcap_log_1000_20240331_103106.txt',
  87548742,
  37677643189,
  421.26140311804033,
  991.94),
 ('glog_files/send_pcap_log_1000_20240331_172059.txt',
  77286027,
  33441159880,
  467.07,
  467.07),
 ('glog_files/send_pcap_log_1000_20240331_185113.txt',
  77286027,
  33441159880,
  466.17,
  466.17),
 ('glog_files/send_pcap_log_100_20240328_154020.txt',
  87548742,
  37677643189,
  99.98652561247212,
  99.99),
 ('glog_files/send_pcap_log_5000_20240328_101717.txt',
  87548742,
  37677643189,
  125.83184855233864,
  341.08),
 ('glog_files/send_pcap_log_5000_20240328_113135.txt',
  87548742,
  37677643189,
  129.43340757238315,
  318.63),
 ('glog_files/send_pcap_log_5000_20240331_082427.txt',
  87548742,
  37677643189,
  484.7475501113585,
  1141.53),
 ('glog_files/send_pcap_log_5000_20240331_083441.txt',
  87548742,
  37677643189,
  485.53171492204916,
  1143.27),
 ('glog_files/send_pcap_log_5000_20240331_084454.txt',
  87548742,
  37677643189,
  486.79750556792857,
  1159.8),
 ('glog_files/send_pcap_log_5000_20240331_085506.txt',
  87548742,
  37677643189,
  483.70581291759436,
  1137.5),
 ('glog_files/send_pcap_log_5000_20240331_090521.txt',
  87548742,
  37677643189,
  484.33285077951007,
  1131.75),
 ('glog_files/send_pcap_log_5000_20240331_100049.txt',
  87548742,
  37677643189,
  435.1027394209355,
  1094.3),
 ('glog_files/send_pcap_log_5000_20240331_101601.txt',
  87548742,
  37677643189,
  423.66097995545675,
  1108.72),
 ('glog_files/send_pcap_log_5000_20240331_171118.txt',
  77286027,
  33441159880,
  462.44,
  462.44),
 ('glog_files/send_pcap_log_tk_20240330_211305.txt',
  82830415,
  36927239583,
  500.80415178571496,
  1176.79),
 ('glog_files/send_pcap_log__1000_20240329_102807.txt',
  87548742,
  37677643189,
  431.9058351893101,
  1184.23)]
```

[4]:

```python
import matplotlib.pyplot as plt

#
results_with_duration = []

#
for file_path in file_paths:
    with open(file_path, 'r') as file:
        local_total_duration = 0.0  #
        for line in file:
            if "Actual:" in line:
                parts = line.split()
                #
                duration = float(parts[-2])
                local_total_duration += duration
        #
        results_with_duration.append(local_total_duration)

#
durations = results_with_duration
durations
```

[4]:

```
[598.6069550000001,
 626.9849130000002,
 2135.7295989999993,
 622.635971,
 2325.452149,
 612.5627029999997,
 689.9379140000001,
 692.7334389999996,
 674.4161130000006,
 572.77,
 573.87,
 3012.074848999999,
 2305.761599999997,
 2216.3110740000006,
 579.5905919999994,
 579.3324519999993,
 578.0003509999997,
 581.2322139999998,
 580.4865360000001,
 651.6273850000001,
 670.8160140000001,
 578.5,
 532.8903299999998,
 658.082771]
```

[5]:

```
#
transfers = [file_name.split('_')[-2][6:] + '_' + file_name.split('_')[-1][:6]␣
 ↪for file_name in file_paths]
transfers
```

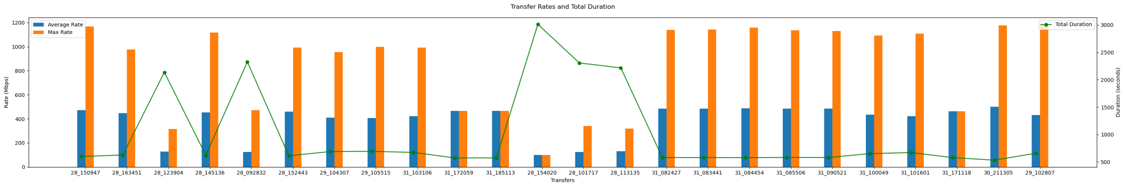
[5]:

```
['28_150947',
 '28_163451',
 '28_123904',
 '28_145136',
 '28_092832',
 '28_152443',
 '29_104307',
 '29_105515',
 '31_103106',
 '31_172059',
 '31_185113',
 '28_154020',
 '28_101717',
 '28_113135',
 '31_082427',
 '31_083441',
 '31_084454',
 '31_085506',
 '31_090521',
 '31_100049',
 '31_101601',
 '31_171118',
 '30_211305',
 '29_102807']
```

[6]:

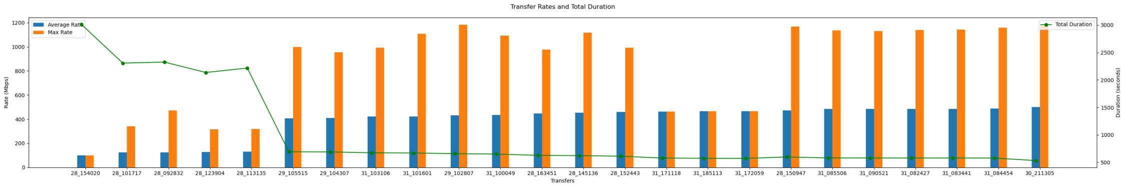
```
#
average_rates = [result[3] for result in results]
max_rates = [result[4] for result in results]

#
index = range(len(transfers))
bar_width = 0.2

#
fig, ax1 = plt.subplots(figsize=(30, 5))

#
ax1.set_xlabel('Transfers')
ax1.set_ylabel('Rate (Mbps)')
ax1.bar(index, average_rates, bar_width, label='Average Rate')
ax1.bar([p + bar_width for p in index], max_rates, bar_width, label='Max Rate')
ax1.set_xticks([p + bar_width / 2 for p in index])
ax1.set_xticklabels(transfers)
ax1.legend(loc='upper left')

#    y
ax2 = ax1.twinx()
ax2.set_ylabel('Duration (seconds)')
ax2.plot(transfers, durations, color='green', marker='o', label='Total␣
 ↪Duration')
```

```
ax2.legend(loc='upper right')

#
fig.suptitle('Transfer Rates and Total Duration')

fig.tight_layout()
plt.show()
```



[7]:
```
#  average_rates, max_rates, transfers, durations

#
combined = list(zip(average_rates, max_rates, transfers, durations))

#          average_rates
#           combined.sort() lambda  lambda x: x[1]
combined_sorted = sorted(combined, key=lambda x: x[0])

#
average_rates, max_rates, transfers, durations = zip(*combined_sorted)

#
index = range(len(transfers))
bar_width = 0.2

fig, ax1 = plt.subplots(figsize=(30, 5))

ax1.set_xlabel('Transfers')
ax1.set_ylabel('Rate (Mbps)')
ax1.bar(index, average_rates, bar_width, label='Average Rate')
ax1.bar([p + bar_width for p in index], max_rates, bar_width, label='Max Rate')
ax1.set_xticks([p + bar_width / 2 for p in index])
ax1.set_xticklabels(transfers)
ax1.legend(loc='upper left')

ax2 = ax1.twinx()
ax2.set_ylabel('Duration (seconds)')
ax2.plot(transfers, durations, color='green', marker='o', label='Total␣
↪Duration')
```

```
ax2.legend(loc='upper right')

fig.suptitle('Transfer Rates and Total Duration')

fig.tight_layout()
plt.show()
```


Transfer Rates and Total Duration
Average Rate
Max Rate
Total Duration
Rate (Mbps)
Duration (seconds)
28_154020
28_101717
28_092832
28_123904
28_113135
29_105515
29_104307
31_103106
31_101601
29_102807
31_100049
28_163451
28_145136
28_152443
31_171118
31_185113
31_172059
28_150947
31_085506
31_090521
31_082427
31_083441
31_084454
30_211305
Transfers