Measurements of Quality Differentiation

Jari Huttunen
Helsinki University of Technology
Networking laboratory
Contents

• Network prototype
  – Topology and system configuration
• Traffic generation
  – Traffic generators and traffic tracing
• Applications
• Theory
  – PHBs
    • Assured Forwarding
    • Expedited Forwarding
• Test cases
  – Best Effort (BE)
  – Expedited Forwarding (EF)
  – Assured Forwarding (AF)
  – EF + AF
  – AF: Bad provisioning
• Summary
• Conclusions
• Future work
Network prototype

- Standard PC-hardware
  - AMD 1300 MHz/256 MB
  - 4 * 3Com 10/100 Ethernet NICs
- 3 core and 5 edge routers
- Dummynet network emulator
  - 30 ms extra delay (low and high delay paths)
- Several traffic generators
System configuration

- FreeBSD OS with ALTQ-package
  - QoS mechanisms (queueing, scheduling, shaping, metering, marking)
- All NICs configured to 10Base-Tx full duplex
- CBQ is used in traffic shaping
  - WRR as a general scheduler
- Static provisioning, no borrowing between classes
  - Capacity differentiation (a predefined amount of link capacity for each traffic class)
- Queue management: tail-drop, RED, RIO
- Token Bucket (TB) and two rate Three Color Marker (TrTCM) used for metering/marking (color blind)
Traffic generation

• Applications with different properties
  – Real time / non-real time
  – Bandwidth sensitive / delay sensitive
  – TCP / UDP based
  – Constant bitrate / varying bitrate
  – Long “friendly” TCP flows / short “aggressive” flows

• How to carry all this traffic in a single network and same time provide quality of service?
Traffic generation (cont.)

• Traffic generators
  – SmartBits 600
  – PC hardware

• SmartBits 600
  – SmartMetrics 10/100 BaseT Ethernet module
  – SmartVoIPQoS
    • A test application to stress and analyze the network’s ability to carry voice and data traffic simultaneously
    • Can generate multiple IP flows and simulate VoIP gateways and phones
    • Measure delay, jitter, throughput and packet loss (+define overall voice quality)

• PC hardware
  – Linux / FreeBSD TCP stack implementation
  – Synchronized by using Network Time Protocol (NTP)
Traffic tracing

- SmartBits 600 can trace the traffic it sends and give aggregated results
  - Transmit flows → Collect data from cards → Display results

- We perform also packet capturing using Tcpdump at client and server side
  - Analysis of TCP traffic
    - RTT, throughput etc.
  - Packet captures are analyzed with Tcptrace tool
    (http://irg.cs.ohiou.edu/software/tcptrace/idex.html)
Applications

• VoIP
  – SmartBits 600 and SmartVoIPQoS software
  – G.711 μ-law voice coding with 20 ms framing
    • Packets size of 218 Bytes
  – We apply 20 flows per client, bi-directional
    • =40 VoIP flows
  – Silence detection is not being modelled
    • Constant bitrate application
  – Mapping to PSQM (Perceptual Speech Quality Measure) voice scoring system (0.4 → 6.5)
    • Comparing measurement results to a matrix that includes mappings between PSQM scores and the effect of impairments (jitter, packet loss)
    • Low PSQM value indicates high voice quality
    • PSQM values are affected by:
      – Frame loss
      – Jitter
      – Type of codec used
Applications (cont.)

• Video streaming
  – Rude/Crude UDP traffic generator/receiver
    • Used with video trace files
  – MPEG-4 encoded video stream from a movie
  – 25 frames per second (40 ms interval)
  – Mean bitrate 1.029 Mbps, max 8.797 Mbps
    • Varying bitrate application

![Video stream data rate profile](image)
Applications (cont.)

- **FTP**
  - Client-server transactions
  - A modified version of Markus Peuhkuri’s Kilent-Server application
  - Packets size of 1500 B (MTU)
  - 50 individual file transfers
  - File size modelled by geometric distribution (mean 500 kB)
  - Example of long lasting TCP connections
Applications (cont.)

- **HTTP**
  - Apache 2.0 www-server
    - Most popular web server in the Internet
  - Siege is used at the client side
    - http testing and benchmarking utility
    - Simulates a predefined number of users
    - Reports of response times, amount of data transferred, etc.
  - 2 clients with high delay paths and 1 client (client #3) with a low delay path
    - Study the effect of dissimilar RTT
  - HTTP 1.0
  - Simulating 165 users
  - Object size modelled by geometric distribution (mean 10 kB)
  - Reading time 12 s
  - Example of short, interactive TCP connections
Test procedure

- **A one minute warm up period before actual measurements**
  - To ensure that network is in stable condition
- **We apply 15 seconds UDP bursts to congest the network**
  - Packets of size 512 B (8 Mbit/s)
    - Does not aim to model any particular application
    - An aggressive, unresponsive source
  - VoIP flows
- **We record all TCP traffic in the network for later analysis**
Per Hop Behavior (PHB)

- Defines the treatment how traffic belonging to a certain behavior group is forwarded at the individual network node
- DiffServ codepoint of a packet (DSCP) is used to select the PHB
- Two standardized PHB groups
  - Assured Forwarding
  - Expedited Forwarding
Assured Forwarding (AF)

- Four independent forwarding classes with three drop precedences per class (RFC 2597)
- The forwarding assurance of an IP packet in a network node is determined by:
  - Resources allocated to the particular AF class
  - The current load of the AF class
  - Drop precedence of a packet
Expedited Forwarding (EF)

- "Leased line emulation"
  - Low loss, low latency and assured bandwidth service
- Defined in RFC 2598
- Strict queueing treatment
- Arrival rate < minimum service rate
  - Requires powerful forwarding from the router
Test cases

- **Case 1:** Best Effort (BE): No differentiation between traffic flows
- **Case 2:** Expedited Forwarding (EF): EF service for VoIP flows.
- **Case 3:** Assured Forwarding (AF): Four independent forwarding classes.
- **Case 4:** EF+AF
- **Case 5:** AF: Different provisioning
Connection pairs

HTTP
FTP
VoIP
Case 1: BE

- Baseline (reference) for our studies
- No differentiation
  - Traffic sources are competing for the resources
  - Situation in today’s Internet
Case 1: BE (cont.)

- TCP is not able to send new data during congestion
- High delay and packet loss

TCP sequence number congestion (FTP connection)

<table>
<thead>
<tr>
<th></th>
<th>Packet loss (%)</th>
<th>PSQM</th>
<th>Avg latency (ms)</th>
<th>Avg jitter (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-&gt;B</td>
<td>14.57</td>
<td>3.16</td>
<td>220.87</td>
<td>1.95</td>
</tr>
<tr>
<td>B-&gt;A</td>
<td>18.15</td>
<td>3.49</td>
<td>231.16</td>
<td>2.82</td>
</tr>
<tr>
<td>Total avg</td>
<td>16.36</td>
<td>3.3</td>
<td>225.56</td>
<td>2.37</td>
</tr>
</tbody>
</table>

VoIP

Video streaming delay
Case 1: BE (cont.)

- Delays not acceptable, heavy packet loss
- http client (client 3) with low delay path is dominating

<table>
<thead>
<tr>
<th>Source</th>
<th>Data transferred (B)</th>
<th>Response time (s)</th>
<th>Throughput (kbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client 1</td>
<td>2751418</td>
<td>3.23</td>
<td>365.33</td>
</tr>
<tr>
<td>Client 2</td>
<td>2616671</td>
<td>2.84</td>
<td>350.17</td>
</tr>
<tr>
<td>Client 3</td>
<td>4272803</td>
<td>0.11</td>
<td>570.75</td>
</tr>
</tbody>
</table>
Case 2: EF

- Two scenarios
  - 20% of the link capacity provisioned for VoIP traffic
  - 30% of the link capacity provisioned for VoIP traffic
- Other traffic sources get BE service
- Max delay set to 20 ms
- Leased line emulation
Case 2: EF (cont.)

20 % provisioning

<table>
<thead>
<tr>
<th></th>
<th>Packet loss (%)</th>
<th>PSQM</th>
<th>Avg latency (ms)</th>
<th>Avg jitter (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-&gt;B</td>
<td>0.080</td>
<td>0.46</td>
<td>109.60</td>
<td>1.019</td>
</tr>
<tr>
<td>B-&gt;A</td>
<td>0.067</td>
<td>0.45</td>
<td>98.56</td>
<td>1.271</td>
</tr>
<tr>
<td>Total avg</td>
<td>0.073</td>
<td>0.46</td>
<td>103.94</td>
<td>1.144</td>
</tr>
</tbody>
</table>

VoIP

30 % provisioning

<table>
<thead>
<tr>
<th></th>
<th>Packet loss (%)</th>
<th>PSQM</th>
<th>Avg latency (ms)</th>
<th>Avg jitter (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-&gt;B</td>
<td>0</td>
<td>0.4</td>
<td>30.870</td>
<td>1.123</td>
</tr>
<tr>
<td>B-&gt;A</td>
<td>0</td>
<td>0.4</td>
<td>33.992</td>
<td>1.572</td>
</tr>
<tr>
<td>Total avg</td>
<td>0</td>
<td>0.4</td>
<td>32.431</td>
<td>1.348</td>
</tr>
</tbody>
</table>

• Increasing provisioning improves the quality of VoIP calls
  – However, there is some oscillation in jitter

20 % provisioning

<table>
<thead>
<tr>
<th></th>
<th>Packet loss (%)</th>
<th>PSQM</th>
<th>Avg latency (ms)</th>
<th>Avg jitter (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-&gt;B</td>
<td>0.080</td>
<td>0.46</td>
<td>109.60</td>
<td>1.019</td>
</tr>
<tr>
<td>B-&gt;A</td>
<td>0.067</td>
<td>0.45</td>
<td>98.56</td>
<td>1.271</td>
</tr>
<tr>
<td>Total avg</td>
<td>0.073</td>
<td>0.46</td>
<td>103.94</td>
<td>1.144</td>
</tr>
</tbody>
</table>
Case 2: EF (cont.)

- Low delay path more dominant when more resources allocated for BE traffic
- EF can be used to provide a leased line emulation

<table>
<thead>
<tr>
<th>Source</th>
<th>Data transferred (B)</th>
<th>Response time (s)</th>
<th>Throughput (kbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client 1</td>
<td>2848275</td>
<td>2.68</td>
<td>380.66</td>
</tr>
<tr>
<td>Client 2</td>
<td>3149460</td>
<td>2.35</td>
<td>421.12</td>
</tr>
<tr>
<td>Client 3</td>
<td>4256773</td>
<td>0.14</td>
<td>565,5</td>
</tr>
</tbody>
</table>

HTTP

80 % for BE service

<table>
<thead>
<tr>
<th>Source</th>
<th>Data transferred (B)</th>
<th>Response time (s)</th>
<th>Throughput (kbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client 1</td>
<td>2814368</td>
<td>2.89</td>
<td>374.56</td>
</tr>
<tr>
<td>Client 2</td>
<td>3352512</td>
<td>1.85</td>
<td>445.37</td>
</tr>
<tr>
<td>Client 3</td>
<td>4088847</td>
<td>0.46</td>
<td>543.01</td>
</tr>
</tbody>
</table>

70 % for BE service
## Case 3: AF

<table>
<thead>
<tr>
<th>Class</th>
<th>Application</th>
<th>Bandwidth %</th>
<th>Buffer size</th>
<th>Conditioner</th>
<th>CIR (Mbps)</th>
<th>Peak rate (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AF1</td>
<td>VoIP+Video</td>
<td>50</td>
<td>20 ms</td>
<td>TB</td>
<td>5</td>
<td>N/A</td>
</tr>
<tr>
<td>AF2</td>
<td>HTTP</td>
<td>20</td>
<td>60 ms</td>
<td>trTCM</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>AF3</td>
<td>FTP</td>
<td>18</td>
<td>120 ms</td>
<td>trTCM</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>AF4</td>
<td>Other</td>
<td>10</td>
<td>360 ms</td>
<td>trTCM</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

- 2% of resources is assigned for the control traffic
- Class AF1 for real time applications
- Class AF2 for interactive TCP traffic (HTTP)
- Class AF3 for non-interactive TCP traffic (FTP)
- Class AF4 for the traffic that does not conform to AF1, AF2 or AF3
Case 3: AF (cont.)

FTP throughput

- FTP throughput is bounded to about 1.8 Mbit/s (the target rate for class AF3)
Case 3: AF (cont.)

- Fairness between HTTP clients
  - AF helps to diminish the effect of RTT
- Real time applications get decent quality of service
- TCP sources achieve (almost) their target rate

### VoIP

<table>
<thead>
<tr>
<th></th>
<th>Packet loss (%)</th>
<th>PSQM</th>
<th>Avg latency (ms)</th>
<th>Avg jitter (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-&gt;B</td>
<td>1.06</td>
<td>1.02</td>
<td>32.27</td>
<td>2.28</td>
</tr>
<tr>
<td>B-&gt;A</td>
<td>0.0</td>
<td>0.4</td>
<td>34.77</td>
<td>3.91</td>
</tr>
<tr>
<td>Total avg</td>
<td>0.53</td>
<td>0.7</td>
<td>33.48</td>
<td>3.10</td>
</tr>
</tbody>
</table>

### HTTP

<table>
<thead>
<tr>
<th>Source</th>
<th>Data transferred (B)</th>
<th>Response time (s)</th>
<th>Throughput (kbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client 1</td>
<td>3742733</td>
<td>1.26</td>
<td>500.20</td>
</tr>
<tr>
<td>Client 2</td>
<td>3744647</td>
<td>1.27</td>
<td>501.21</td>
</tr>
<tr>
<td>Client 3</td>
<td>3931651</td>
<td>0.79</td>
<td>522.31</td>
</tr>
</tbody>
</table>
Case 4: EF+AF

<table>
<thead>
<tr>
<th>Class</th>
<th>Application</th>
<th>Bandwidth %</th>
<th>Buffer size</th>
</tr>
</thead>
<tbody>
<tr>
<td>EF</td>
<td>VoIP</td>
<td>20</td>
<td>20 ms</td>
</tr>
<tr>
<td>AF1</td>
<td>Video</td>
<td>30</td>
<td>20 ms</td>
</tr>
<tr>
<td>AF2</td>
<td>HTTP</td>
<td>20</td>
<td>60 ms</td>
</tr>
<tr>
<td>AF3</td>
<td>FTP</td>
<td>18</td>
<td>120 ms</td>
</tr>
<tr>
<td>AF4</td>
<td>Other</td>
<td>10</td>
<td>360 ms</td>
</tr>
</tbody>
</table>

- A finer differentiation between real time applications
  - EF PHB for VoIP flows
- Otherwise same as in previous case
Case 4: EF+AF (cont.)

VoIP and HTTP flows experience similar QoS as in AF case.

**HTTP**

<table>
<thead>
<tr>
<th>Source</th>
<th>Data transferred (B)</th>
<th>Response time (s)</th>
<th>Throughput (kbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client 1</td>
<td>3613647</td>
<td>1.45</td>
<td>483.75</td>
</tr>
<tr>
<td>Client 2</td>
<td>3760971</td>
<td>1.29</td>
<td>495.84</td>
</tr>
<tr>
<td>Client 3</td>
<td>3922848</td>
<td>0.82</td>
<td>525.32</td>
</tr>
</tbody>
</table>

**VoIP**

<table>
<thead>
<tr>
<th></th>
<th>Packet loss (%)</th>
<th>PSQM</th>
<th>Avg latency (ms)</th>
<th>Avg jitter (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-&gt;B</td>
<td>0</td>
<td>0.4</td>
<td>32.199</td>
<td>1.807</td>
</tr>
<tr>
<td>B-&gt;A</td>
<td>0.040</td>
<td>0.43</td>
<td>34.386</td>
<td>3.665</td>
</tr>
<tr>
<td>Total avg</td>
<td>0</td>
<td>0.4</td>
<td>33.293</td>
<td>2.736</td>
</tr>
</tbody>
</table>
• Resources are exhausted during the first UDP burst in AF1 class (video streaming)
  – Bad provisioning (overload in the class)
### Case 5: AF different provisioning

<table>
<thead>
<tr>
<th>Class</th>
<th>Application</th>
<th>Bandwidth %</th>
<th>Buffer size</th>
<th>Conditioner</th>
<th>CIR (Mbps)</th>
<th>Peak rate (Mbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>AF1</td>
<td>VoIP+Video</td>
<td>30</td>
<td>20 ms</td>
<td>TB</td>
<td>5</td>
<td>N/A</td>
</tr>
<tr>
<td>AF2</td>
<td>HTTP</td>
<td>40</td>
<td>60 ms</td>
<td>trTCM</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>AF3</td>
<td>FTP</td>
<td>10</td>
<td>120 ms</td>
<td>trTCM</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>AF4</td>
<td>Other</td>
<td>18</td>
<td>360 ms</td>
<td>trTCM</td>
<td>2</td>
<td>3</td>
</tr>
</tbody>
</table>

- More bandwidth allocated for TCP traffic
Case 5: AF different provisioning (cont.)

<table>
<thead>
<tr>
<th>Packet loss (%)</th>
<th>PSQM</th>
<th>Avg latency (ms)</th>
<th>Avg jitter (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td>A-&gt;B</td>
<td>14.62667</td>
<td>3.21</td>
<td>32.026</td>
</tr>
<tr>
<td>B-&gt;A</td>
<td>0</td>
<td>0.4</td>
<td>33.666</td>
</tr>
<tr>
<td>Total avg</td>
<td>0</td>
<td>1.81</td>
<td>33.846</td>
</tr>
</tbody>
</table>

VoIP

<table>
<thead>
<tr>
<th>Source</th>
<th>Data transferred (B)</th>
<th>Response time (s)</th>
<th>Throughput (kbps)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client 1</td>
<td>4148934</td>
<td>0.40</td>
<td>554.67</td>
</tr>
<tr>
<td>Client 2</td>
<td>4136732</td>
<td>0.45</td>
<td>549.64</td>
</tr>
<tr>
<td>Client 3</td>
<td>4251938</td>
<td>0.21</td>
<td>565.70</td>
</tr>
</tbody>
</table>

HTTP
## Summary

<table>
<thead>
<tr>
<th>Case</th>
<th>PSQM</th>
<th>Client 1</th>
<th>Client 2</th>
<th>Client 3</th>
<th>Packet loss %</th>
</tr>
</thead>
<tbody>
<tr>
<td>BE</td>
<td>3.3</td>
<td>3.23</td>
<td>2.84</td>
<td>0.11</td>
<td>15.1</td>
</tr>
<tr>
<td>EF (30%)</td>
<td>0.4</td>
<td>2.89</td>
<td>1.85</td>
<td>0.46</td>
<td>56.3</td>
</tr>
<tr>
<td>EF (20%)</td>
<td>0.46</td>
<td>2.68</td>
<td>2.35</td>
<td>0.14</td>
<td>23.2</td>
</tr>
<tr>
<td>AF</td>
<td>0.7</td>
<td>1.26</td>
<td>1.27</td>
<td>0.79</td>
<td>2.46</td>
</tr>
<tr>
<td>EF+AF</td>
<td>0.4</td>
<td>1.45</td>
<td>1.29</td>
<td>0.82</td>
<td>7.69</td>
</tr>
<tr>
<td>Case 5</td>
<td>1.81</td>
<td>0.40</td>
<td>0.45</td>
<td>0.21</td>
<td>32.2</td>
</tr>
</tbody>
</table>
Conclusions

• In Best Effort network the traffic sources are interfering with each other
• EF can be used to provide premium service
• AF helps to reduce the effect of RTT for TCP connections
  – Better fairness
• CBQ is not a perfect solution
  – Need for adaptive schedulers?
• It’s all about provisioning
  – The problem of provisioning the resources
  – Need for dynamic provisioning
• Tuning the parameters is not easy
Future work

• Test different mechanisms
  – Borrowing, different schedulers, queuing algorithms etc.
• Measurements using dynamic provisioning
• Adding more flows / connections per class
• More realistic traffic distribution
  – More TCP connections
• Deeper analysis on data
• Measurements using Adtech AX4000
• Development of centralized management platform
  – Easier to manage the network and traffic generators
Thank you

Questions ?