Service-Centric Networking with SCAFFOLD

Written by: Michael J. Freedman, et al.
Presented by: Mert Coskun
The Problem

- Telephone Network
The Problem

0 WWW(1991) : A New Era Begins!
   0 Content

0 Software As a Service (2001)
   0 Service
SCAFFOLD

0 CLEAN SLATE

0 Focus: Replication & Dynamism

0 Solution: Flow-Based Anycast

0 My Opinion: GSLB + IPNL in Network Layer
Architecture
Replication & Dynamism
Three Pillars of SCAFFOLD

0 New Header
   0 Service Identifier + Network Address (LISP/HIP)

0 New Routers
   0 Service and Network Routers (SR = GSLB+R ? /IPNL )

0 New API
   0 Bind to SERVICE
## Application Portability

<table>
<thead>
<tr>
<th>Application</th>
<th>Version</th>
<th>Codebase</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iperf</td>
<td>2.0.0</td>
<td>5,934</td>
<td>240</td>
</tr>
<tr>
<td>TFTP</td>
<td>5.0</td>
<td>3,452</td>
<td>90</td>
</tr>
<tr>
<td>PowerDNS</td>
<td>2.9.17</td>
<td>36,225</td>
<td>160</td>
</tr>
<tr>
<td>Wget</td>
<td>1.12</td>
<td>87,164</td>
<td>207</td>
</tr>
<tr>
<td>Elinks browser</td>
<td>0.11.7</td>
<td>115,224</td>
<td>234</td>
</tr>
<tr>
<td>Mongoose web server</td>
<td>2.10</td>
<td>8,831</td>
<td>425</td>
</tr>
<tr>
<td>Memcached server</td>
<td>1.4.5</td>
<td>8,329</td>
<td>159</td>
</tr>
<tr>
<td>Memcached client</td>
<td>0.40</td>
<td>12,503</td>
<td>184</td>
</tr>
<tr>
<td>Apache Benchmark / APR</td>
<td>1.4.2</td>
<td>55,609</td>
<td>244</td>
</tr>
</tbody>
</table>
## Performance

<table>
<thead>
<tr>
<th>Stack</th>
<th>Mean</th>
<th>Std dev</th>
</tr>
</thead>
<tbody>
<tr>
<td>TCP/IP</td>
<td>929.8 Mbit/s</td>
<td>5.3 Mbit/s</td>
</tr>
<tr>
<td>SCAFFOLD (kernel)</td>
<td>596.6 Mbit/s</td>
<td>17.0 Mbit/s</td>
</tr>
<tr>
<td>SCAFFOLD (user)</td>
<td>110.1 Mbit/s</td>
<td>16.1 Mbit/s</td>
</tr>
<tr>
<td>SCAFFOLD (user with tracing)</td>
<td>82.3 Mbit/s</td>
<td>8.8 Mbit/s</td>
</tr>
<tr>
<td>Router</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Service (Resolution)</td>
<td>12.99 Kpkts/s</td>
<td>0.17 Kpkts/s</td>
</tr>
<tr>
<td>Network (Data forwarding)</td>
<td>13.25 Kpkts/s</td>
<td>1.47 Kpkts/s</td>
</tr>
</tbody>
</table>
The Cool

- **Service Name**
  - Network Layer Security
  - Self-certifying

- **Late Binding**
  - Flexibility (replication)
  - Automatic mobility/adaption

- **API**
  - Address are Transparent to Application
  - Simple Addition / Replacement / Subtraction
The Questionable

- No Connectionless Communication?
- Current System Works!
- The Prototype & Evaluation
- Virtual Machine Migration
  - Remus?
The Bad

- New Routers
  - Change / +1 Hop
  - More Work

- New API

- No backwards compatibility

- No obvious incentive
My Opinion

- Cool **BUT** Too Revolutionary for wide Adoption

- **May be** used within data centres

- P2P Services?

- Motivation for evolutionary research
THANK YOU!

0 Questions?
Load Balancing

**Graph 1:**
- X-axis: Time (s)
- Y-axis: Active Requests
- Legend: Server 1 (red), Server 2 (green), Server 3 (blue)
- Markers: Drain start, Shed start, Shed end, Drain end

**Graph 2:**
- X-axis: Time (s)
- Y-axis: Goodput (Mbps)
- Legend: Client 1 (red), Client 2 (blue)
- Annotations: One server fails