**Motivation**

By using simplified implementations and models of protocols, traditional simulators do not provide functional realism. DCE aims to increase the realism of experimentation results by running existing protocol implementations inside a network simulator.

<table>
<thead>
<tr>
<th>Simulator</th>
<th>Testbeds</th>
<th>Emulators</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functional Realism</td>
<td>???</td>
<td>✓</td>
</tr>
<tr>
<td>Timing Realism</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Topology Flexibility</td>
<td>✓ (limited)</td>
<td>✓</td>
</tr>
<tr>
<td>Easy Replication</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>Easy Debug</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>Experimental Scalability</td>
<td>✓</td>
<td></td>
</tr>
</tbody>
</table>

**What is DCE helpful for?**

- To obtain realistic results from network simulation
- To avoid re-implementing possibly complex protocols for simulation (e.g., OSPF or BGP)
- To debug and test your code within a controlled environment
- To create a model of a large scale network (let’s say, a model of an ISP network) in a single node

**DCE Architecture**

DCE consists of 3 Components:

- **Core layer**
  - virtualization primitives handles stacks, heaps, global memory in a single host process.
- **Kernel layer**
  - as-generative implementation of Linux with glue code
- **POSIX layer**
  - reimplementation of standard socket API

**Supported Applications /Kernels:**

- Linux kernel (2.6.36, 3.4, 3.5, 3.7, and 3.10)
  - IPv4, IPv6, TCP/UDP, DCCP, mptcp, Mobile IPv6, etc.
- Quagga, PARC ccnx, iperf, ip, ping/ping6, umip, bind9, unbound, httpd, BitTorrent

**Demonstration 1**

Highlights

- Realistic performance evaluation of CCNx over MANET
- CCNx data transfer simulation
- Network configuration:
  - WiFi mobile nodes, random direction model (RD)
  - MANET routing such as OLSR, DSR (for comparison with CCNx)
- Traffic: ccnpeek, ccncpoe, iperf

**Demonstration 2**

Highlights

- Realistic simulation (Linux MPTCP)
- Debugging environment (i.e., gdb)

**For further information**

- Project Web page:
- Contact:
  Hajime Tazaki (tazaki at wide.ad.jp)
  Emilio Mancini (emilio.mancini at inria.fr)

**References**