The New NS2 Emulation Facility
Kevin Fall
kfall@ee.lbl.gov

- NS2 is the simulation vehicle for the VINT project:
  - **USC/ISI**: Deborah Estrin, Mark Handley, John Heideman, Ahmed Helmy, Polly Huang, Satish Kumar, Kannan Varadhan, Daniel Zappala
  - **LBNL**: Kevin Fall, Sally Floyd
  - **UCBerkeley**: Elan Amir, Steve McCanne
  - **Xerox PARC**: Lee Breslau, Scott Shenker

- **VINT** is currently funded by DARPA through mid-1999
ns2 Architecture

- Discrete-event C++ simulation engine
  - scheduling, timers, packets
- Based on ns1 (S. McCanne and V. Jacobson)
- Split Otcl/C++ object “library”
  - protocol agents, links, nodes, classifiers, routing, error
generators, traces, queuing, math support (random
variables, integrals, etc)
- separate visualization tool (“Nam”)
- used on many UNIXes, tested on Windows
Supported Components

• Protocols:
  – tcp (2modes + variants), udp, ip, rtp/rtcp, SRM, 802.3 MAC, 802.11 MAC

• Routing
  – global topology map, classifiers
  – static unicast, dynamic unicast (distance-vector), multicast

• Queueing and packet scheduling
  – FIFO/drop-tail, RED, CBQ, WRR, DRR, SFQ

• Topology: nodes, links Failures: link errors/failures

• Emulation: plug simulator into live network
New Emulation Facility
(note: still under active development)

• Generic which “tap” live networks
  – inject received packets into simulation
  – emit packets on to live network
  – associated with a “network object”

• Network Objects
  – Raw IP and UDP/IP network object
    • send/receive raw IP packets or UDP/IP
    • IP multicast support
  – pcap network object
    • send/receive link-layer frames
    • use tcpdump/pcap filtering language
Sample Environment

Internet

Proxy ARP for Laptop

DHCP for address/routing

Emulation Machine

Laptop
Emulation Machine

Simulator

RAW IP
Sample Script

set ns [new Simulator]
$ns use-scheduler RealTime
set me [exec hostname]
set bpf0 [new Network/Pcap/Live]
$bpf0 set promisc_ true
set ipnet [new Network/IP]
set nd0 [$bpf open readonly fxp0]
$ipnet open writeonly
set filt “(not ip host $me)”
$bpf0 filter $filt
set a0 [new Agent/Tap]
$a0 network $bpf0
set n0 [$ns node]
$ns attach-agent $n0 $a0
TCP Results (10 packet periodic drop)

- Dropped SYN
- ACK of SYN+ACK
- Retransmitted SYN
- Lost
- Dropped SYN
Relevance to TCP-IMPL

• Rich multi-protocol simulation environment
• Modular structure easily extended
• Traffic and error generation
Additional Information

• **Web pages:**
  - http://www-mash.cs.berkeley.edu/ns
  - http://netweb.usc.edu/vint

• **Mailing list:**
  - majordomo@mash.cs.berkeley.edu
  - “subscribe ns-users”