



Emulation in ns

Presented by
Alefiya Hussain

What is Emulation

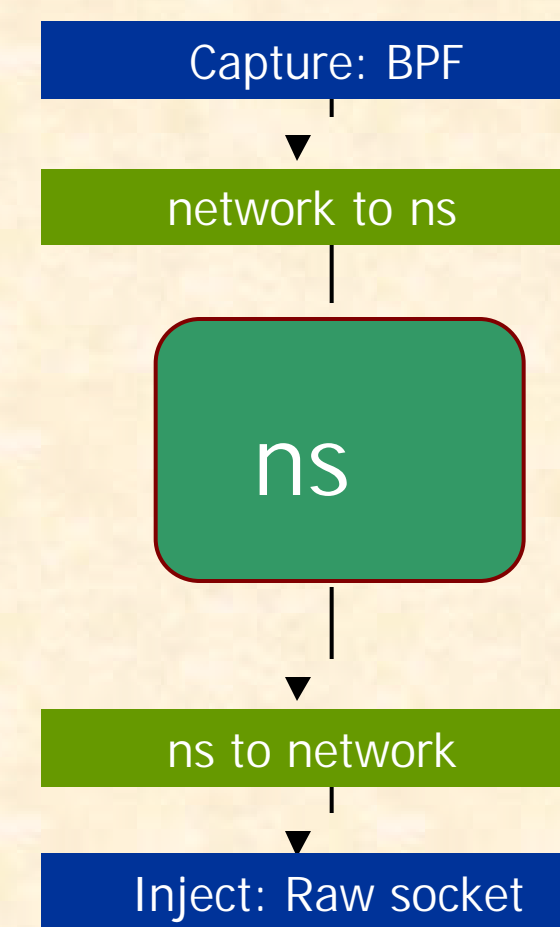
- ◆ Ability to introduce the simulator into a live network
- ◆ Application:
 - Allows testing real-world implementations
 - Allows testing simulated protocols
- ◆ Requirements
 - Scheduler support
 - Packet capture and generation capabilities

Scheduler

- ◆ ns operates in virtual time using event-driven simulation
- ◆ Real-time scheduler
 - Does not advance the virtual clock to next event
 - Dispatches event at real-time

Emulation Objects

- ◆ Interface between ns and network traffic
- ◆ *Network Objects*
Access to live network via BPF and raw sockets
- ◆ *Tap Objects*
Conversion between ns and network packet formats

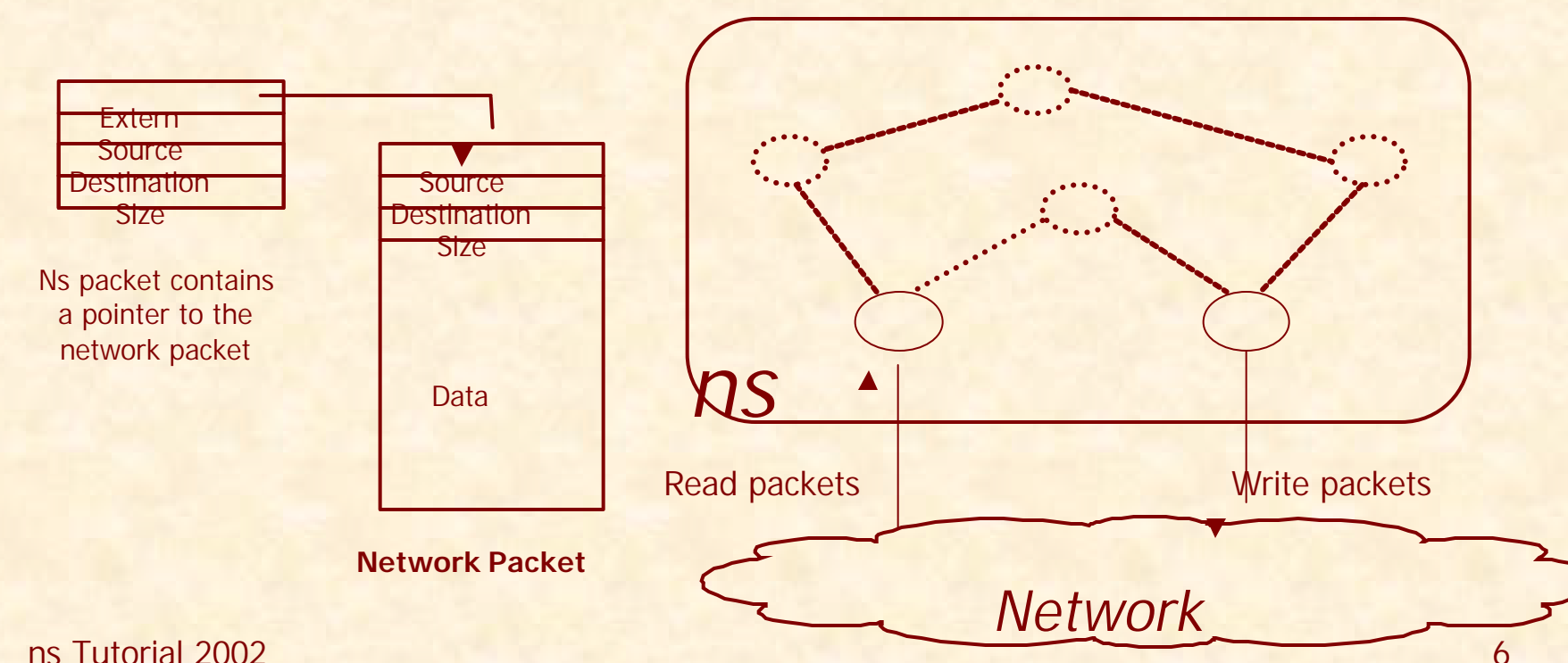


Modes of Operation

- ◆ Packet conversion leads to two modes of operation
- ◆ Opaque Mode
 - ◆ Network packet fields are not interpreted by ns
- ◆ Protocol Mode
 - Network packet is interpreted
 - ◆ TTL values reflect hop count in simulator
 - Network packet fields are generated
 - ◆ Echo responder, TCP application

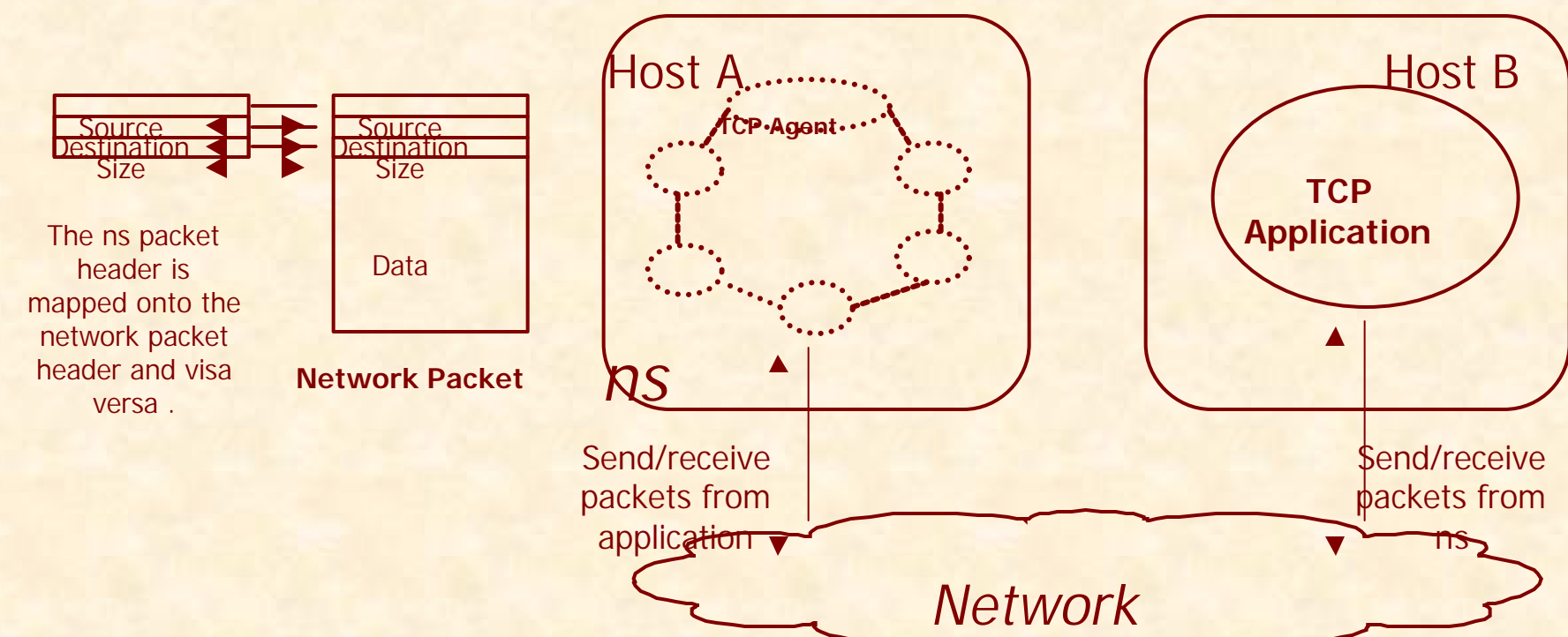
Opaque Mode

- ◆ Network packet is passed unmodified through simulator



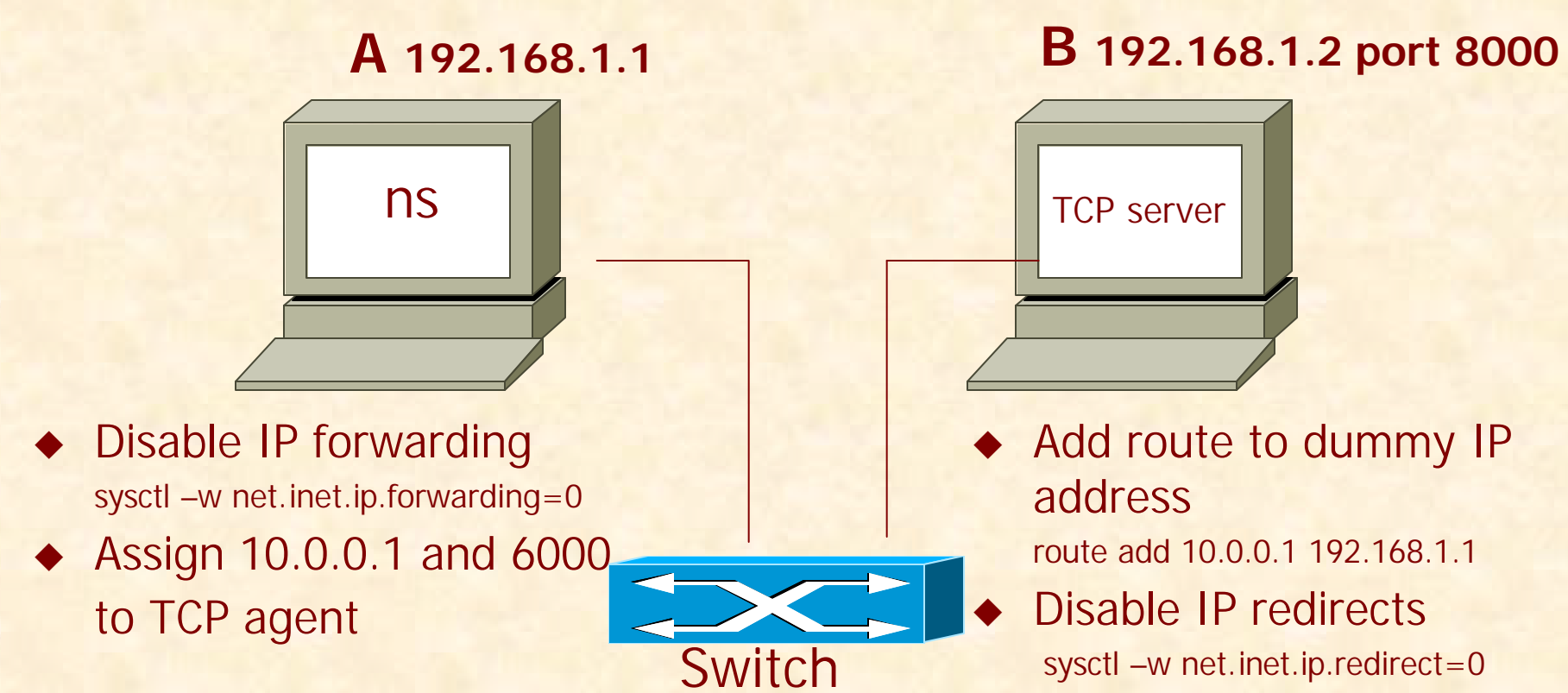
Protocol Mode

- ◆ Network packet fields are generated by the simulator



Example: Setup

Goal: Make a ns TCP agent interact with a TCP server



Example Script

```

set ns [new Simulator]
$ns use-scheduler RealTime

set entry_node [$ns node]
set tcp_node [$ns node]

$ns duplex-link $entry_node \
  $tcp_node 10Mb 1ms DropTail

set tcp [new Agent/TCP/FullTcp]
$ns attach-agent $tcp_node $tcp
  
```

} Activate ns and
Change to real-time
scheduler

} Create topology

} Create TCP Agent



Example Script

```
set capture_tap [new Agent/TCPtap];
set bpf [new Network/Pcap/Live];
set dev [$bpf open readonly eth0]
$bpf filter "src 192.168.1.2 and src port 8000 \
           and dst 10.0.0.1 and dst port 6000"
$capture_tap network $bpf;
$ns attach-agent $entry_node $capture_tap;
$ns simplex-connect $capture_tap $tcp
```

Capture &
convert to
ns format

```
set inject_tap [new Agent/TCPtap];
set ipnet [new Network/IP];
$ipnet open writeonly
$inject_tap network $ipnet;
$inject_tap advertised-window 512
$inject_tap extipaddr "192.168.1.2"
$inject_tap extport 8000
$ns attach-agent $entry_node $inject_tap;
$ns simplex-connect $tcp $inject_tap
```

Convert to
network
format &
inject

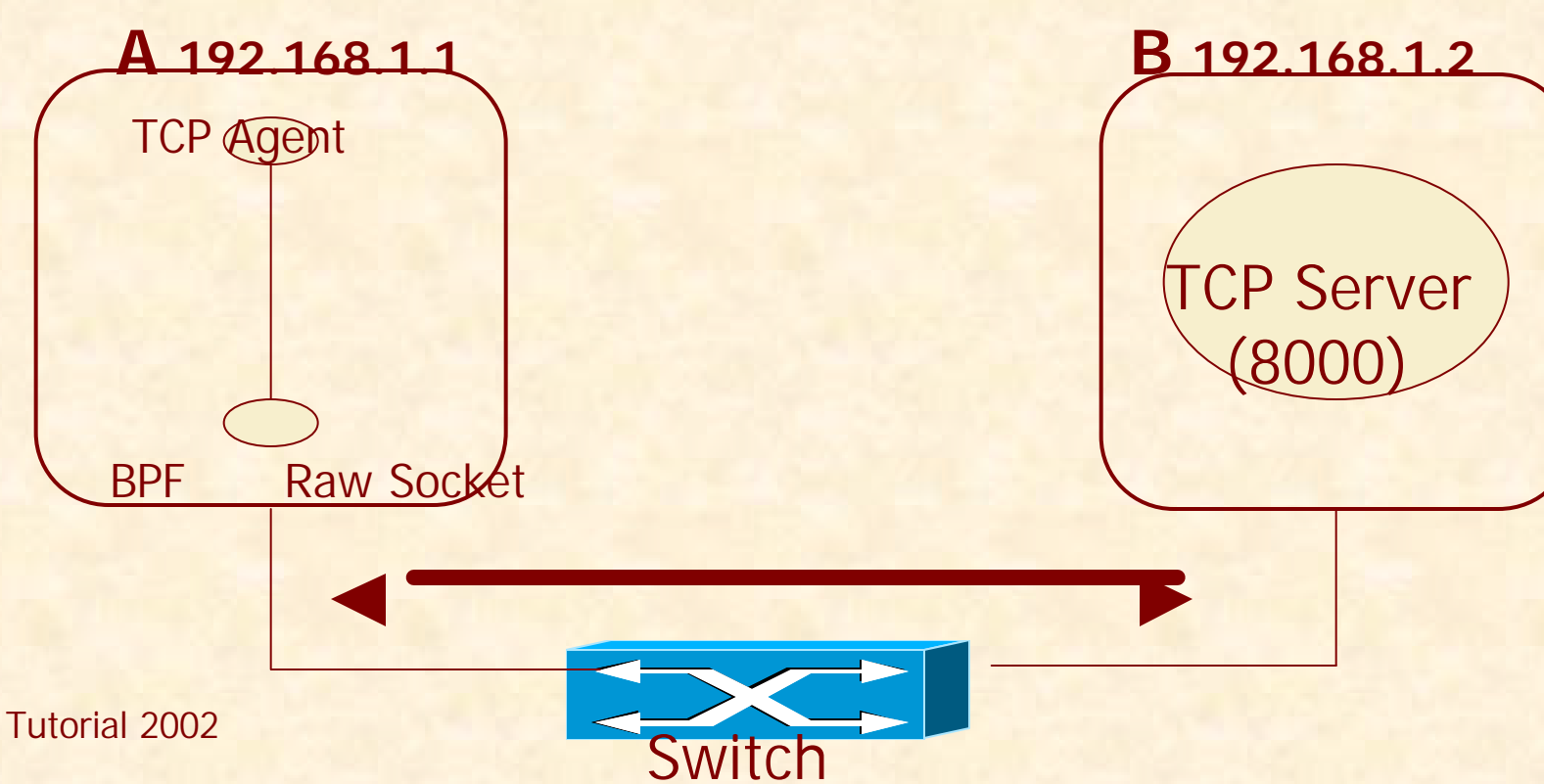
ns 10.0.0.1

10

Example Script

```
$ns at 0.01 "$tcp advance 1"  
$ns at 20.0 "exit 0"  
$ns run
```

} start ns



Applications

- ◆ Opaque Mode
 - Cross-traffic interaction leading to drop, delay and re-ordering of packets
 - End to end application testing
- ◆ Protocol Mode
 - Protocol and conformance testing
 - Evaluate effect of DDoS attacks
 - Wireless networks



Information Sciences Institute



SCHOOL OF
ENGINEERING

Further Information

- ◆ <http://www.isi.edu/nsnam/ns/ns-emulation.html>
- ◆ Scripts ~ ns/emulate
 - Opaque Mode:
 - ◆ em.tcl
 - Protocol Mode
 - ◆ thrutcp.tcl
 - ◆ pingdemo.tcl
 - ◆ tcpemu.tcl
- ◆ Fall K., *Network Emulation in the Vint/NS Simulator*, ISCC Jul 1999