VirtualMesh: An Emulation Framework for Wireless Mesh Networks in OMNeT++

Thomas Staub, Reto Gantenbein
University of Bern, Switzerland
Outline

> Motivation
  — Experiments with simulations and inside test beds
  — Requirements not satisfied by simulation and test beds
> Related work
  — How do Cisco, Tropos, ZyXel etc. test their equipment?
  — ORBIT
  — Wireless emulation platforms
> VirtualMesh
  — Architecture
  — Performance evaluation
> Conclusions / Future work
> Questions
Motivation

> New protocols, concepts and ideas have to be verified
  – Proof-of-Concept
  – Different topologies
  – Different scales (10 - 10‘000 nodes)
  – Different environmental conditions (background interferences)
  – Cross-layer testing

> Development steps
  – Simulation
  – Prototype implementations
  – Testing the prototype within a test bed

> Easy to handle test facilities improve the software evolution
Experiments with Simulations and inside Test Beds

> Simulations
+ Fast prototyping
+ Variable abstraction level
+ Large scale experiments, mobility
+ Repeatable results
  – Everything has to be implemented in the simulation
  – Relevance of results for real world
  – Limited level of detail

> Test Beds
+ Existing software
+ Real protocol / architecture implementation
  – Bad repeatability
  – Effort to test with different topologies, mobility etc.
  – Test beds limited
  – Errors due to implementation or changing environment
Features not Satisfied by Simulation and Real Test Beds

> Enhanced testing facilities for real communication software
  — Repeatability
  — No influence (or controlled) influence of environment
  — Large scale tests
  — Simple tests with mobility
  — Stress testing
  — Cross-layer testing

> Network emulation can provide these features
How do Cisco, Tropos, ZyXel etc. test their equipment?
How do Cisco, Tropos, ZyXel etc. test their equipment?
How do Cisco, Tropos, ZyXel etc. test their equipment?
ORBIT: Open Access Research Testbed for Next Generation Wireless Networks [1]

- orbit-lab.org (5.45 M$ NSF funding 2003 - 2007)
- Rutgers, Columbia, and Princeton University
- Lucent Bell labs, IBM Research and Thomson
- 20 x 20 nodes
Wireless Emulation Platforms

> MobiEmu [2]
  - UML nodes
  - Central component which monitors and controls the network state of all UML instances

  - „Virtualized“ Linux systems on top of L4 microkernel (Fiasco)
  - MobiEmu emulation master

> JiST/MobNet [4]
  - Combined simulation, emulation and real-world testbed for ad hoc networks
  - Java-based
  - Common code base
VirtualMesh

> Key idea
  - Combine the features of simulation and testing on real systems
  - Traffic redirected to a model by a virtual interface
  - Network access layer handled by a simulation model
  - Scalability by virtualization
VirtualMesh: Overview

Virtualized Nodes with Virtual Interfaces: (Emulation)

Real Nodes with Virtual Interfaces

Virtual Interfaces

XEN Supervisor

Communication between Nodes and Model

Model in Network Simulator (Omnet++)

Mesh Routers

Mesh Clients

Wireless Client
Traffic Redirection: Virtual Interface

> Virtual interface
  - TUN device
    - wireless extensions (e.g. channel, rts/cts, tx power)
  - PacketModeller transmits and receives data from WLAN model
Packet Flow

Simulation model in Omnet++
Packet Format (PacketModeller)

> Node data message
  — Wireless parameters have to be transmitted with every packet

<table>
<thead>
<tr>
<th>Type: data</th>
<th>ID</th>
<th>Channel</th>
<th>TxPower</th>
<th>Options</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ethernet Packet</th>
</tr>
</thead>
</table>

> Node register message
  — Nodes have to be registered at the wireless model
  — Model has to know on which IP/port the node is listening

<table>
<thead>
<tr>
<th>Type: register</th>
<th>ID</th>
<th>virtual MAC</th>
<th>host name</th>
<th>IP</th>
<th>Port</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Packet Flow in Simulation Model

Traffic to/from nodes:

- cRAWSocketRTScheduler
- PacketProxy
- NodeManager
- ProxyHost
- WlanNIC
- ProxyNIC

Channel Control:

ID | hostname | IP   | port | virtual MAC
---|----------|------|------|------------

NodeManager:

- ID
- hostname
- IP
- port
- virtual MAC


Simulated host:

- Packet
- Proxy Host

ProxyHost:

- Packet
- WlanNIC
- ProxyNIC
Performance Evaluation: Host Virtualization

RTT between two Linux hosts

- Two physical hosts
- Physical host and paravirtualized host
- Physical host and fully virtualized host
Performance Evaluation: Packet Modeller Overhead

RTT between two virtualized hosts

- Both hosts paravirtualized
- Both hosts fully virtualized
- Both hosts fully virtualized and using virtual network interface
Performance Evaluation: WLAN Model Overhead
Conclusions / Future Work

> Emulation provides a valuable extension of the testing facilities for communication software

> VirtualMesh
  — Host virtualization
  — Traffic interception
  — Integration of OMNeT++

> Future Work
  — Extensive testing of VirtualMesh
  — Improve performance (delays)
VirtualMesh: An Emulation Framework for Wireless Mesh Networks in OMNeT++

References


Questions
VirtualMesh: An Emulation Framework for Wireless Mesh Networks in OMNeT++

THE LAB/OFFICE WHITEBOARD

DO NOT ERASE!!

Leaves here by previous generation of grad students

DO NOT REALLY WHITE ANYMORE

List of supplies to buy: why buy it yourself when you can just write it here and hope it magically appears

WHITEBOARD PRODUCTIVITY:

A BEAUTIFUL MIND

QUID PRO QUO

"HEY, LOOKS LIKE A PARTY"

"CHAO!

PEOPLE IN FRONT OF WHITEBOARD

MARKER ROULETTE:

Only marker that works

Have wrong color cap on

Actually a permanent marker #@$!$

IN CASE OF EMERGENCY

Magnets & Menus from late night take out places

Jorge Cham © 2008

WWW.PHDCOMICS.COM
Example from Our ResearchGroup
Example from Our Research Group