OpenAirInterface Virtual Network Simulator

Daniel Câmara, Hicham Anouar, Cristian Bonnet, Fethi Filali, Raymond Knopp
Institut Eurecom
2229, route des Crêtes
BP 193 06904
Sophia-Antipolis, France
+33 4 93 00 81 00
{daniel.camara, hicham.anouar, christian.bonnet, fethi.filali, raymond.knopp}@eurecom.fr

ABSTRACT
This paper describes the OpenAirInterface Virtual Network Simulator (OVIS) tool. This tool offers the possibility of deploying in one single computer a complete wireless network scenario. It is designed for protocol and application designers who need to have their software running in a controlled environment as close as possible to the real one.

OVIS intends to decrease the cost and complexity of performing software measures and evaluations for real applications and protocols. Each node in the simulated network is a complete machine instance with a set of real software and a whole network stack.

Categories and Subject Descriptors
I.6.7 [**Simulation and Modeling**]: Simulation Support Systems – *Environments.*

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design – *Wireless communication.*

D.4.8 [**Operating Systems**]: Performance – *Measurements, Monitors, Simulation.*

General Terms
Measurement, Performance

Keywords
Virtualization, simulation, wireless networks.

1. INTRODUCTION
The available tools for testing and simulating protocols for wireless networks rely, normally, on simplification and abstraction of the communication layers. Tools like NS-2 [1] and Qualnet [2], allow the simulation of large and complex networks, however, they need to abstract some parts of the network stack to make the simulations feasible. The problem with this approach is that if one does not use a real implementation, real problems in

the algorithms may pass unnoticed until their implementation and deployement in a real wireless environment.

The creation of small testbeds is another technique extensively used to validate protocols and applications for wireless networks. However the creation of testbeds, even small ones, presents a considerable cost. Furthermore, evaluating the results of a distributed protocol in a testbed may be a challenging and error prone activity. In this case, first, the network deployment needs to be done carefully, if one needs to test specific topologies. Additionally, repeating experiments with the same parameters may be impossible, once the wireless medium is unstable.

The ideal environment for testing a piece of software for wireless networks should enable a simple and inexpensive way to create and manage wireless network topologies, where each node use its own real network stacks in a controlled environment. The OVIS proposal is exactly this one. The possibility of, in a simulator, use the same network stack real computers do, with an easy and flexible configuration system. To be able to do that OVIS relies on the OpenAirInterface project implementation [3]:.

The OpenAir is an open-source hardware/software development platform for collaborative innovation in the area of digital radio communications. It implements in software the PHY (Physical) and MAC (Medium-access) / RLC (Radio Link Control) / RRC (Radio Resource Control) layers for wireless communications as well as providing a IPv4/IPv6/MPLS network device interface.

The remaining of this paper is organized as follows; next section presents the proposed simulator architecture. Section 3 presents the tool used to manage simulations and Section 4 presents the conclusions and next steps for this project.

2. ARCHITECTURE
The OpenAirInterface Virtual Network Simulator (OVIS) tool, using the structure provided by the OpenAirInterface project, creates a series of virtual machines (VMs) inside the real one. From each node point of view it is a completely autonomous and separate entity, using its own resources and network stack. However, in truth, it is a virtual instance of an operating system inside a real machine. The main components of the project are:

- *OpenAirInterface* [3]: to provide the functionality of the layers PHY + MAC/RLC/RCC

- *OpenVZ* [4]: to provide the virtualization platform to instantiate the virtual nodes

- *Linux* [5]: is used as the virtual machines operating system.

It provides also the network upper layers and applications.

Intentionally, all pieces of software used in this project are open source and freely available. The OVIS intends is to provide a completely free and open alternative for testing software for wireless new generation networks.

As shown at Figure 1 the simulation is done by generating virtual instances of Linux OS inside the real machine. Each node has its own memory space, IP address, OpenAirInterface stack, upper layer stacks and file system. The technique permits the creation of both homogeneous and heterogeneous networks. In an homogeneous network each element has the same operating system and capacities. In an heterogeneous networks the nodes may have different operating systems and capacities.
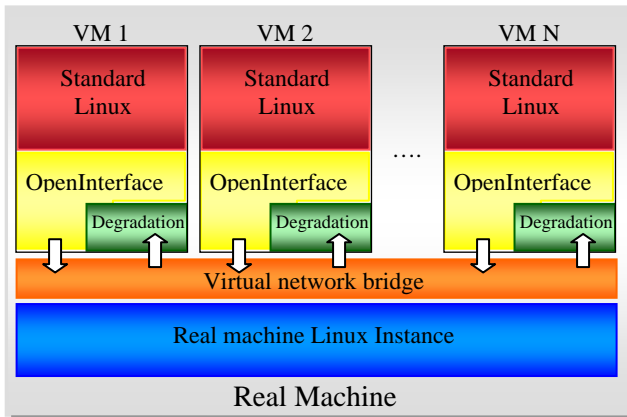

Figure 1 - Virtual Open Interface Simulator architecture

All VMs are "physically" connected through a virtual bridge inside the real machine. Each node has access to the other virtual nodes, as well to the Internet. The OpenAirInterface is also responsible for managing the network topology and degrade the nodes signal. The signal degradation module implements a mathematically defined radio channel model. Such model is built specifically to express, as close as possible, the real radio/medium relation. The degradation module is placed in the receiver, which sends the damaged message to its upper layers, if the signal SINR relation is above a defined threshold. The message, in the upper layers, is treated as any other regular message. Indeed, the signal degradation module is the only simulated process in the whole OVIS structure. All the other processes are the same ones used in real wireless network nodes.

3. VIRTUAL SIMULATOR TOOL

To easily perform the experiments a graphical tool was build. With this tool the user may define the topology scenarios, store the simulation description as XML file, create scripts to run the simulation and follow the simulation runs. The generated shell scripts are responsible for create the virtual network, lunch the structure, run the experiments and stop the virtual network.

On the other hand the XML scenario file provides a desirable degree of flexibility to OVIS. Using XML the scenarios are independent from the GUI tool. In this way the user may easily duplicate scenarios, change parameters directly on the XML file or even use the description as input for other tools. The graphical

tool also permits the user to follow the VMs behavior during the simulation. The OVIS tool process is depicted at Figure 2.

4. CONCLUSIONS AND FUTURE WORK

We truly believe the simulation approach presented here has a series of interesting characteristics may interest to a large range of researchers. The first, and the most important characteristic, is the possibility of evaluate a wireless network using standard stack implementations and applications. This may anticipate many real problems as it brings the simulation results much closer to the reality. Among the other interesting characteristics we may highlight is the existence of a global clock, which makes it easier the job of analyzing and making sense logs. This characteristic also may help the development of newer protocols. In truth, the designer now can count with a global clock to design and analyze his/her distributed system. In addition to this, collecting and processing the logs is also easier. Other interesting asset of OVIS is the possibility of using standard UNIX tools to log the traffic and behavior of the nodes during the experiments.
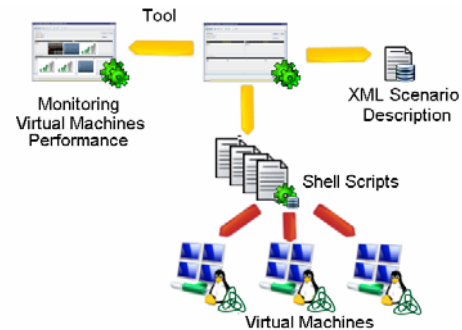

Figure 2 - The tool description

Regarding the topology limits, even though, theoretically, there is no static limit for the number of nodes in the virtual network, the number of nodes instances is bounded by the capacity of the real machine in use. Each new virtual node represents a cost in terms of memory and processing to the real machine.

This work is still in its early stages, but it already presents some good results and a clear indication that OVIS may became a really powerful tool in the next few years. The next steps for the project are, first, the creation of a smarter degradation module, second, update in the capacity of the OpenAirInterface tool to handle more nodes, third, the addition of a mobility mode to the tool.

5. REFERENCES

[1] K. Fall and K. Varadhan, The ns manual, The VINT Project, *UCB,   LBL,   USC/ISI   and   Xerox   PARC,* *http://www.isi.edu/nsnam/ns/doc/, Jan. 2008*

[2] QualNet, http://www.qualnet.com/

[3] OpenAir Interface, http://www.openairinterface.org/

[4] OpenVZ, http://openvz.org/

[5] Linux, http://www.linux.org