

# The DES-Framework

## Extending a Wireless Multi-Hop Testbed by virtualization and simulation

Mesut Güneş, Oliver Hahm, Kaspar Schleiser  
Distributed Embedded Systems  
Computer Systems and Telematics  
Institute of Computer Science  
Freie Universität Berlin, Germany  
E-Mail: {guenes, hahm, schleiser}@inf.fu-berlin.de

***Index Terms***—Wireless Multi-Hop Networks, Wireless Sensor Networks, Testbed, Virtualization, Simulation

### I. INTRODUCTION

For more than two decades many different algorithms and protocols for Wireless Multi-Hop Networks (WMN) and Wireless Sensor Networks (WSN) have become subject to research. The most common way to study these networks was based on simulations, although currently there is a trend towards testbed-based research. While some characteristics of these networks are hard to model in simulation environments, the experimentation with testbeds is often complicated and very time-consuming.

To mitigate this trade-off we developed the *DES-Testbed* [1] [2] [3], a Wireless Multi-Transceiver Network testbed located on the campus of *Freie Universität Berlin*. Currently the testbed consists of 110 *DES-Nodes* spread over three buildings. Each *DES-Node* consists of a wireless router equipped with three IEEE 802.11a/b/g transceivers and a wireless sensor node. While the wireless transceivers form the Wireless Mesh Network *DES-Mesh*, the sensor nodes establish a Wireless Sensor Network called *DES-WSN*. Thus, a Wireless Mesh and Wireless Sensor Network are operated in parallel, making the testbed one of the largest hybrid networks world-wide.

A centralized management backend, located on the testbed servers, accesses the *DES-Nodes* by an ethernet backbone. In combination with a sophisticated software framework the *DES-Testbed* provides the infrastructure to study WMN and WSN protocols in a real-world environment while having almost the convenience of a simulator. The software framework consists of *DES-Script* [4] to define and describe the experiments, *DES-TBMS* [5] to schedule, manage and execute and them *DES-Vis* to monitor and visualize the current state as well as the the results.

### II. DIFFICULTIES IN DEPLOYING A REAL-WORLD TESTBED

There are several problems and issues that arise from researching on testbeds. Dealing with real-world influences

and real hardware is not an easy challenge. In the *DES-Testbed* several services and tools are set up to lessen most of the problems that, for instance, occur due to hardware errors, misconfiguration, radio noise or interference.

Currently two services exist to monitor the state of the *DES-Nodes* and management network as well as external influences like radio noise or weather conditions. *DES-Mon* collects network and, therefore, experiment relevant data from the *DES-Nodes* via SNMP [6] while the *Mesh-Monitor* collects data about the current state of the individual *DES-Nodes* using collectd [7].

To avoid manual maintenance of every single *DES-Node* we make use of a hardware watchdog in combination with the corresponding Linux Kernel daemon. Thus, even systems that are not responding due to network problems or kernel crashes will recover automatically as soon as possible without the need for manual intervention. Moreover, the *DES-Nodes* boot the Linux Kernel via TFTP and then mount the whole root file system using NFS [8]. This allows changing the software and data for all the nodes at the same time. Due to these mechanisms physical access to the *DES-Nodes* has become redundant.

For the unlikely case where a system will not recover or some serious problems occur at the management backbone, a Nagios [9] service is running on the testbed servers. By Nagios not only the *DES-Nodes*, but also the testbed servers themselves are being monitored. In this manner serious problems are discovered immediately.

### III. EXTENDING THE TESTBED

One of the drawbacks of using a testbed for research on WMN in comparison to simulation-based approaches is the total amount of participating nodes. Even in an advanced testbed setup, as introduced for the *DES-Testbed* in the last section, numbers of one thousand or more nodes would be hard to handle. Beside the challenges of managing such a huge testbed there is also the cost factor of actual hardware.

Virtualization is one approach to mitigate some of the difficulties mentioned above. Whereby researching on networks

in the range of about one thousand nodes is possible. Using virtualized nodes also allows the creation of any arbitrary topology. Simulators even go one step further and make networks in the range of ten thousand nodes feasible. On the other hand simulations require migrating the original software into the simulator while virtualized nodes can run exactly the same system.

Having all three testing facilities (testbed, virtualization and simulation) using the same platform for experiment description, execution and evaluation enables the comparison of results. In this way we are able to make inferences between these facilities in both directions. In terms of, for instance, scalability of an algorithm this enables new potential (see Figure 1).

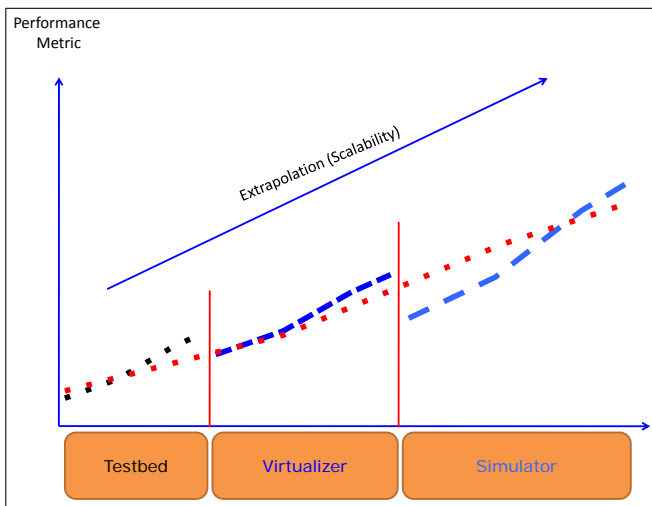


Fig. 1. The testbed can be extended using virtualization and simulation

#### IV. VIRTUALIZATION USING *DES-Virt*

Our virtualization framework *DES-Virt* is based on the Linux Kernel Virtual Machine (KVM) [10]. Currently one virtualization server with 8 CPU cores is used to virtualize up to 1000 routers. In order to make the virtualized routers behave as similar to the real routers as possible, they are booted using PXE and NFS from the same central server using the same kernel and operating system image as the hardware routers. From the testbed management point of view, there is no difference between a virtualized router and a real router.

Virtual wireless interfaces are used to connect the virtual machines to each other or to the real testbed. Based on the `mac80211_hwsim` kernel module, they present themselves like real wireless interfaces to userspace, offering channel and rate selection, ad-hoc and infrastructure modes.

Packets sent over virtual wireless interfaces are repacked into UDP packets and then sent over the management ethernet backbone to a central wireless emulation server. Using simple XML-based topology descriptions, any needed topology can be defined. As of now, it is possible to define a packet loss ratio per link, per data rate.

#### V. CONCLUSION

If necessary, our testbed can easily be extended using virtualized nodes, enabling the researcher to extend his experiments from 100 Nodes to about 1000. As the virtualized nodes behave as much as possible like the hardware routers, seamless scaling of the experiments is made possible. Is a bigger number of nodes necessary for an experiment, our framework offers the simulation of additional nodes, further extending experiments to about 10000 nodes.

Put together, the combination of virtualization and simulation greatly increase the scalability of experiments on our experimentation platform.

#### REFERENCES

- [1] M. Günes, B. Blywis, and F. Juraschek, "Concept and Design of the Hybrid Distributed Embedded Systems Testbed," Freie Universität Berlin, Tech. Rep. TR-B-08-10, 2008.
- [2] M. Günes, B. Blywis, F. Juraschek, and P. Schmidt, "Practical Issues of Implementing a Hybrid Multi-NIC Wireless Mesh-Network," Freie Universität Berlin, Tech. Rep. TR-B-08-11, 2008.
- [3] B. Blywis, M. Günes, F. Juraschek, and J. Schiller, "Trends, advances, and challenges in testbed-based wireless mesh network research," *ACM/Springer Mobile Networks and Applications (MONET)*, February 2010, special Issue on Advances In Wireless Testbeds and Research Infrastructures. [Online]. Available: <http://www.springerlink.com/content/hk25051135m36571/>
- [4] M. Günes, F. Juraschek, B. Blywis, and O. Watteroth, "DES-CRIPT - A Domain Specific Language for Network Experiment Descriptions," in *Proceedings of the International Conference on Next Generation Wireless Systems (NGWS) 2009*, Melbourne, Australia, 12-14, October 2009.
- [5] M. Günes, B. Blywis, F. Juraschek, and O. Watteroth, "Experimentation Made Easy," in *Proceedings of the First International Conference on Ad Hoc Networks, ICST*, Ed., Ontario, Canada, September 2009.
- [6] J. Case, M. Fedor, M. Schoffstall, and J. Davin, "A simple network management protocol (snmp), RFC 1157," <http://tools.ietf.org/html/rfc1157>, May 1990.
- [7] collectd.org, "collectd - The system statistics collection daemon," <http://www.collectd.org/>, 2010. [Online]. Available: <http://www.collectd.org/>
- [8] S. Shepler, B. Callaghan, D. Robinson, R. T. Hurlow, C. Beame, M. Eisler, and D. Noveck, "Network file system (nfs) version 4 protocol, RFC 3530," <http://tools.ietf.org/html/rfc3530>, April 2003.
- [9] nagios.org, "Nagios - The Industry Standard for IT Infrastructure Monitoring," <http://www.nagios.org/>, 2010. [Online]. Available: <http://www.nagios.org/>
- [10] Qumranet, "Kvm: Kernel-based virtualization driver," 2006. [Online]. Available: [http://www.linuxinsight.com/files/kvm\\_whitepaper.pdf](http://www.linuxinsight.com/files/kvm_whitepaper.pdf)