Berlin: The Berlin Experimental Router Laboratory for Innovative Networking

Dan Levin, Andreas Wundsam, Amir Mehmood, and Anja Feldmann

Deutsche Telekom Laboratories, TU Berlin, Berlin, Germany
{dan,andi,amir,anja}@net.t-labs.tu-berlin.de

Abstract. Today's disruptive approaches to rearchitecting the Internet, e.g., *Clean Slate Networking* initiatives require testbeds that present unprecedented flexibility to the experimenter. This poster presents Berlin, a flexible testbed platform designed towards the requirements of Future Internet research. Berlin combines a diverse landscape of network elements, both software-defined and legacy hardware, and unifies them under a common management interface, presenting them as *pluggable services* to the experimenter.

Keywords: future internet, testbed support, experiment services.

1 Introduction

The Internet's core architecture, including its routing and addressing scheme currently aches under the pressure of the continuing rapid expansion of the net, as well as new applications and usage patterns. Useful incremental improvements have not seen wide adoption in recent years (e.g., IPv6, DiffServ, DNSSec). This has sparked renewed interest in disruptive approaches for re-architecting the Internet core, for instance in a multitude of *Clean Slate Networking* initiatives around the globe.

Due to the diversity of these approaches, Future Internet experimentation requires a flexible testbed that caters to a wide range of requirements. Such experiments may require programmable, virtualized and non-virtualized hardware (e.g., PC servers), software-configurable hardware (e.g., NetFPGAs), and flexibly configurable logical and physical topologies. Experimentation also requires a sound infrastructure of support, including the generation of realistic traffic patterns and proper modeling of user expectations and experiences.

This proposed poster introduces Berlin, an experiment platform tailored to the requirements of Future Internet research. Berlin combines a diverse landscape of network elements, both software-defined and legacy hardware, unifies them under a common management interface, and presents them as pluggable services to the experimenter. We first present its architecture and the core services provided, then examine some use-cases of successful Future Internet experimentation in the lab. We believe that our poster can contribute valuable insights to the discussion about the primitives, components, and services required to successfully enable Future Internet experiments.

T. Magedanz et al. (Eds.): TridentCom 2010, LNICST 46, pp. 602–604, 2011.

2 Architecture

BERLIN uses a three-layered architecture with a multitude of different *experimental devices* managed by a unified *management platform*. On this foundation rests the *pluggable service* infrastructure, offering combinations of hardware, software, and configurations as pre-built and customizable services to the user.

BERLIN contains a highly diverse collection of experimental devices including routers, switches, and traffic-generating servers. These devices differ widely in their feature set, performance, and control interfaces. Terminal servers and SNMP controllable power interfaces provide additional *out-of-band management*.

This varied landscape of devices is managed by our custom software management system, the *Labtool*. *Labtool* presents a unified, vendor-agnostic interface to the experimenter for device reservation, configuration, interaction (e.g., console access, power management), and topology management. The Labtool also maintains a complete and historically versioned picture of the physical and logical network testbed topology. Labtool integrates with an automated system configuration and disk imaging tool which allows disk images and router and switch configurations to be deployed quickly onto arbitrary experimentation devices.

This unified management allows BERLIN to offer pre-configured combinations of hardware and software as higher-level *pluggable services* as depicted in Figure 1. These fulfill many common requirements, e.g., traffic generation, monitoring and capturing, network emulation, NetFPGA packet processing, and virtualization services. These pluggable services allow researchers to quickly establish an experimental setup with most of the required services from pre-built components. For instance, an experimenter may want to evaluate a new router primitive implemented as a *NetFPGA* program, then require *self similar background traffic* to be generated and routed through the *NetFPGA*, apply emulated WAN line delay characteristics, and finally capture packet level traces at several points in the experiment.

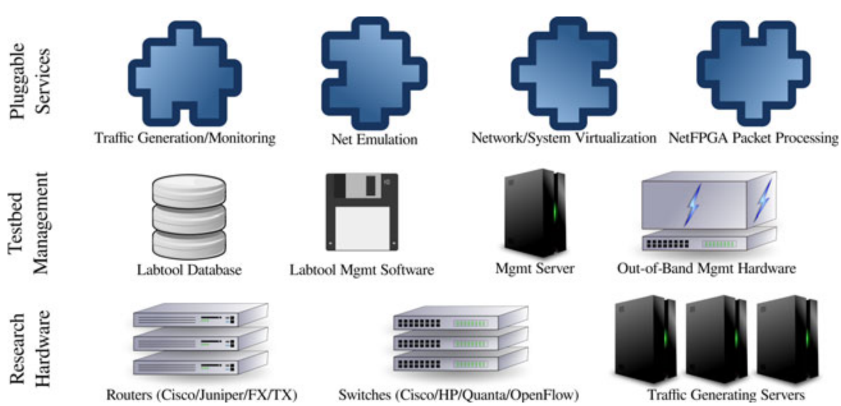

Fig. 1. BERLIN Testbed Architecture

3 Case Studies

We next present cases where Berlin supports Future Internet experimentation.

HAIR, the *Hierarchical Architecture for Internet Routing* addresses the limits of current Internet routing table growth. It better supports mobility, traffic engineering, and multipath routing by separating the *locator and identifier* elements of the addressing scheme. HAIR places routing elements into *hierarchical name-spaces* to localize the effects of routing changes and improve routing table scalability. Berlin enables us to devise an experiment to evaluate the HAIR software-based routing elements, all of which utilize special network packet formats. Virtualized hosts with the the Click modular software router as provided by Berlin can support these special packets. The Berlin topology management enables simple reconfiguration of the underlying IP network onto which HAIR elements are overlaid. Results from the prototype evaluation show the feasibility of hierarchical routing with mobility utilizing this model of split locator and identifier [1].

QoE/Virt Providing good *Quality of Experience*(QoE) for end users remains a challenging problem on the Internet. Especially *virtualized networks* pose novel challenges, as multiple virtual networks may compete for the same physical resources and the quality of the isolation provided by virtualization platforms varies. Accordingly, we devise a combined experiment on our testbed, confronting *multimedia VOIP traffic* with bursty, self-similar *background traffic* as found on the Internet, and routing both over a *virtualized substrate* based on XEN and OpenFlow with advanced debugging and flow management capabilities. Our results indicate that the advanced troubleshooting and management capabilities in the virtualized setup result in improved QoE, even though the level of isolation provided by the virtualization is limited [2].

4 Summary and Outlook

We have presented the architecture of Berlin and some case studies underlining its potential as an experimentation platform. In the future, we plan to further extend Berlin's flexibility, specifically to automatically integrate OpenFlow networks into experiments. To support experiments beyond the size of a single physical testbed we will investigate extending the Berlin testbed architecture to other sites, as well as federating Berlin with other Future Internet testbeds, while maintaining the architecture of pluggable services that is paramount to its flexibility.

References

1. Feldmann, A., Cittadini, L., Mühlbauer, W., Bush, R., Maennel, O.: Hair: hierarchical architecture for internet routing. In: ACM ReArch Workshop (2009)
2. Wundsam, A., Mehmood, A., Feldmann, A., Maennel, O.: Network troubleshooting with shadow vnets. In: Proc. ACM SIGCOMM Demo Session (2009)