G-Lab - an Experimental Facility for Future Internet Research and its International Context

Paul Müller, Dennis Schwerdel
Integrated Communication Systems Lab
University of Kaiserslautern
Paul-Ehrlich-Straße, Gebäude 34
67663 Kaiserslautern
pmueller, schwerdel@informatik.uni-kl.de

Abstract: The G-Lab project aims to investigate new networking paradigms and algorithms for a future internetworking architecture in an experimental manner. Thus the G-Lab project consists of a spiral process of two major fields of activities: research studies of future network components and their experimental design within an experimental facility. Both activities are controlled by the same community to ensure that the experimental facility fits to the demand of researchers. Researchers gain access to virtualized resources or may also gain exclusive access to resources if necessary. This paper presents the current setup of the G-Lab experimental facility, and puts the platform into an international context.

1 Introduction

Today's Internet has a large economic influence but is based on mechanisms and algorithms from the 70ies and 80ies. The ever changing requirements of applications and capabilities of the transport technologies demands for changes even of the core technologies of the Internet. Thus several research efforts worldwide currently investigate concepts and technologies for new future internetworking architectures [RM08]. The goal of the G-Lab project is to foster experimentally driven research in this field.

The G-Lab project[1] has started in 2008 as a BMBF[2] funded distributed joint research and experimentation project for Future Internet studies and development between six German universities: Würzburg, Kaiserslautern, Berlin, München, Karlsruhe, and Darmstadt. G-Lab can be divided in two major interacting tasks, the Future Internet research projects and the experimental facility. That means that the G-Lab project is not limited to explore only theoretical possibilities and novel ideas but also to use experimental approaches to verify the derived results while using the experimental facility. To investigate the functional aspects of novel internetworking architecture approaches (like routing, addressing, control, monitoring & management) and their interaction with each other is such an intricate task that could not be validated only in an analytical way.

[1] http://www.german-lab.de
[2] German Federal Ministry of Education and Research, "Bundesministerium für Bildung und Forschung"

The project itself is composed in diverse working groups that are dedicated to different aspects of future Internet research, ranging from architecture to mobility and management. A special working group deals with a distributed experimental facility consisting of wired and wireless hardware with over 185 nodes, which are fully controllable by the G-Lab partners. This infrastructure provides the experimental facility to the G-Lab working groups to test their proposed approaches and ideas for the future Internet architecture. The whole network of the platform is distributed into individual clusters at the six different locations within Germany with Kaiserslautern as the main site. The first version of platform was available in March 2009 and first experiments took place at the commencement of April.

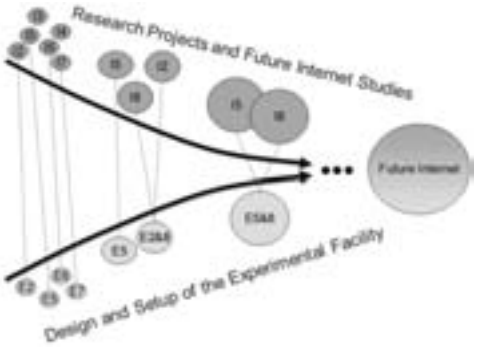


Figure 1: German-Lab philosophy

The overall goal of the G-Lab project is that theoretical research and the experimental facility will converge into a new future internetworking architecture as depicted in Figure 1. Thus it is important that the experimental facility is flexible enough to adapt to the needs of the experiments and ultimately become a research field in itself. With this interwoven approach G-Lab avoids the situation that the platform providers offer their services but nobody is going to use it.

2 Experimental Facility Design

In the design of the G-Lab experimental facility [SGH+10] it has been an important point to use existing solutions, adapt them if needed and integrate them. Thus it was possible to build up a running testbed very quickly. The usage of free and mostly open source software solutions allowed to use the full budget for hardware equipment and also makes it easy to adapt the used software.

2.1 Hardware Layout

The hardware equipment consists of three types of nodes and one switch per site. The nodes can be classified in the following category types:

NormalNode: This is the standard type of node, which can be used to run networking tests and computations. These nodes contain two Xeon Quadcore Processors with 16 GB of RAM, 500 GB of HDD and four gigabit Ethernet interfaces.

NetworkNode: The second node type is designated for special networking tests requir-

ing more network interfaces. To facilitate Emulab-like experiments, the network nodes differ from the normal nodes in the sense that there are eight gigabit Ethernet interfaces.

HeadNode: The last type is acting as a head node and manages the local site. This node differs from the other types by a faster CPU and much more disk space.

After vigorous scrutiny, Sun Microsystems and Cisco Systems have been chosen as hardware providers for the facility. All the nodes include a dedicated service processor, i.e. a small computer that allows controlling and monitoring the hardware remotely with a special management network interface. Each site has one head node, two network nodes and a variable amount of normal nodes. The networking equipment consists of a layer-3 switch from Cisco Systems (Catalyst 4500 E Series).

2.2 Network Setup

All nodes of a site are located in one network segment interconnected by the switch, which has been split into two virtual switches using VLANs. The public part contains all interfaces of the normal and network nodes and all except one interface of the head node. The private part contains all management interfaces of the service processors and one normal interface of the head node. Both networks are completely separated and only the public network has an uplink to the Internet. With this separation the access to the service processors can be controlled by the head node.

Public IP addresses are needed for all interfaces of each node (except management interface). The addresses are distributed by the head node using DHCP. Global DNS records are managed by the main site (Kaiserslautern), a site-specific zone is delegated to each site to allow decentralized DNS management.

Some sites have policies denying externally controlled nodes with IP addresses in the address range of that site, because some access rules are based on IP ranges. In this situation special firewall rules have been set up that blocks all communication between the nodes and the rest of the site except a few defined proxy hosts.

2.3 Headnode Structure

In the initial design of the experimental facility the head node has an operating system running directly on the hardware, which has early been recognized as being inflexible. Now the head node has been virtualized and separated in a couple of virtual machines. This has some major advantages:

- Different functionality can be separated into different virtual machines. This even allows for different operating systems (e.g. Fedora Linux and Debian Linux) running on these machines.

- Virtual machines allow easy backups with snapshots of running machines.
- Virtual machines can be cloned and the clone can then be used for development and testing purposes, it can even be sent to other sites.
- The virtualization host provides a remote control (e.g. console login) over the virtual machines which are an additional way of access in case a virtual machine is not working properly.

As a virtualization solution Proxmox VE is being used but other solutions like VMWare, Xen [BDF+03] and VirtualBox[3] are also being examined. Currently the head node in Kaiserslautern (main site) has virtual machines for monitoring (section 2.6), PlanetLab Central (section 2.5), website, user database, a file server, the head node software and various machines for testing purposes.

The headnode software manages and controls all local nodes at a site. It provides the following services:

- Administration of the local network segment using DHCP.
- Provision of boot images for the associated nodes using PXE netboot (see section 2.4).
- Administration of access to the management interfaces of the local nodes via VPN.
- Proxy for monitoring that allows the central monitoring server to monitor the management interfaces (see section 2.6).

This system is provided as a set of Debian packages. So, all sites have the same base system consisting of software from a shared repository.

2.4 Flexible Software Deployment

The headnode software of the local site provides boot images for the nodes via PXE[4] Netboot. Thus any boot image can be booted on any node. In the context of German-Lab we define three categories of boot images:

1. PlanetLab boot image (described in section 2.5): This allows a node to boot the PlanetLab software which is the default. This boot image contains a part that is specific to each node.
2. Virtualization boot image: This kind of boot image provides virtualization with access for all German-Lab users. Thus users can use nodes booted with this image to run custom software images by means of the used virtualization technology. The

[3] http://www.virtualbox.org
[4] Preboot Execution Environment

default virtualization boot image is Proxmox VE. which provides both OpenVZ virtualization and and KVM[5].

3. Custom boot images: This kind of boot image contains a system designed by a user and only allows access to a limited user group specified by the system itself.

There is a clear trade-off between access for more users and more privileges for users. PlanetLab provides a very good virtualization when measured in the number of concurrent users that it allows, but it is very limited in the hardware access it provides (e.g. only TCP and UDP sockets, no raw sockets). Custom boot images can provide full hardware access and also allow for kernel modifications but restrict the number of users that can access the node.

The German-Lab experimental facility allows both, access for all users to almost all nodes (PlanetLab software is the default) and full access to a few nodes if needed. A central management platform for distributing boot images and assigning them to the nodes called "Boot Image Management" (BIM) has been developed.

2.5 PlanetLab Infrastructure within G-Lab

PlanetLab [PBFM06, Fiu06, PR06] is a software environment, that allows to virtualize nodes using the VServer technology and which provides a central managing and control platform. There is also a testbed called PlanetLab (for which the software has been designed) with which we do not currently share resources.

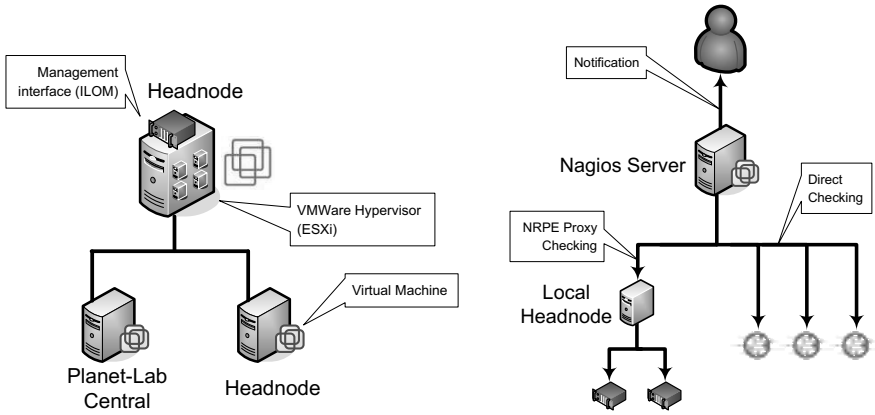
The PlanetLab software consists of a central server called PlanetLab Central (PLC) and a boot image for all nodes. On the PLC all sites, users and nodes can be configured and a custom boot image for each node can be generated.

In German-Lab the PLC runs in a virtual machine on the head node in Kaiserslautern. In the PlanetLab testbed the boot image is booted from a CD or a USB device but in German-Lab that has been modified to be used as a PXE boot image that is provided by the head node software at each site. Figure 2a shows how the PlanetLab software is used in German-Lab. Administrators configure the node on the PLC, which then provides a custom boot image. This boot image is used on the local headnode to boot the node via PXE. Once the node is booted, the node only communicates with the PLC and the users.

2.6 Central Monitoring

The monitoring of the entire infrastructure is also part of the goal. A dedicated virtual server in Kaiserslautern is used for the monitoring infrastructure. The software Nagios[6] is being used to collect monitoring data of individual hosts and services and notify adminis-

[5] http://www.linux-kvm.org
[6] http://www.nagios.org



(a) Planet-Lab usage

(b) Monitoring of the G-Lab Platform

Figure 2: German-Lab Structure

trators by e-mail when problems occur. Information that is currently monitored is:

- Resource usage (CPU, memory, disk, etc.) on all virtual machines
- Hardware health of all nodes (using the service processors)
- Availability of all nodes and service processors

Some of this information is not visible for the monitoring server like resource usage on distant hosts and information of hosts that are not visible from the server like the service processors. To allow the monitoring of these hosts and services the Nagios Remote Plugin Executor (NRPE) software is being used as a proxy. NRPE is a server that allows specified hosts (i.e. the G-Lab monitoring server, see figure 2b) to execute preconfigured commands. With this proxy both internal data and hidden hosts can be checked.

To configure the data for the Nagios software (e.g. hosts, services, check commands, users), Nagios Administrator[7] is used. The monitoring information can be visualized in two ways:

1. A structure diagram gives the current state of each host or host group with green, yellow or red lights. The NagVis[8] software is used for this purpose.
2. Using PNP4Nagios[9] the history of monitored values can be visualized in a time-line graph for each host and each service.

[7] http://www.nagiosadmin.de
[8] http://www.nagvis.org
[9] http://www.pnp4nagios.org

The web-frontends of Nagios, the Nagios Administrator and both visualization tools have been combined in a central website[10]. Of course all monitoring information is also being stored in log files so that future visualization or analysis can work on the history too. The G-Lab monitoring architecture has been valuable since it was deployed and helps to detect and solve problems quickly. Problems that can be fixed without hardware change have frequently been solved within a few hours.

2.7 Design Support for Experiments

A lot of software for experimental facilities has been developed and each one works at a certain level of realism, concurrency and repeatability. The German-Lab experimental facility allows its researchers to choose from various experimental facilities software tools. Within the G-Lab project special software tool for designing and deploying experiments to the experimental facility was developed. This software tool is called Topology Management Tool (ToMaTo)[SHG+11, SRM11b]. ToMaTo allows researchers to create virtual network topologies populated by virtual nodes running standard software. It is a design goal of ToMaTo to overcome limitations found in experimental facility software so that the user has maximal flexibility for his experiments. ToMaTo allows its users to configure and use multiple concurrent network topologies. It also aims to allow lightweight virtualization and full operating system access for the experiments .

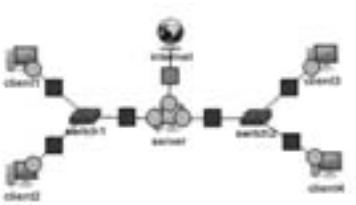


Figure 3: An example topology

The goal of ToMaTo is to enable users to create and use network topologies for their experiments. A network topology consists of two types of components. Devices are active components like computers that run the software of the experiment and are the only sources and sinks of data. Connectors are network components that connect devices and transport their data exhibiting certain configurable characteristics. Figure 3 shows a topology with four client devices, one server device, two switch connectors and one Internet connector.

Figure 4: ToMaTo graphical front end

[10] http://nagios.german-lab.de

ToMaTo uses virtualization technologies to allow experiments to run concurrently in isolated environments spanning parts of the experimental facility. ToMaTo consists of three modules, the host system, the central back-end and the web-based front-end (http://tomato.german-lab.de) as shown in figure 4. The host system runs on all hosts of the experimental facility and offers virtualized resources to be controlled by the central back-end. The host hypervisor consists of a Linux operating system with the following additional components installed:

- PROXMOX VE[11] as virtualization tool for virtual machines
- Tinc[12] as virtualization tool for virtual networks
- TC/Netem as a link emulation tool

2.8 Experiment example

A lot of experiments used or still use the German-Lab experimental facility to develop, test, analyze or evaluate different aspects in the field of future networking. One experiment [SRM11a] that has been run on the experimental facility demonstrates[13] the usage and benefits of ToMaTo in protocol analysis with the scenario of malware analysis. Malware poses a huge security thread on Internet users as it has access to all data on the computer, can record user actions without the knowledge of the user and send this data over the Internet. The most common kind of malware allows the attacker to control the computer remotely and use it to launch other attacks and send spam mails. This way malware is currently responsible for most attacks and spam mails in the Internet.

An analysis of the communication protocol between the malware on the victims computer and the attacker can lead to methods to detect infected computers and quarantine them. Flaws in the communication protocol might offer a way to destroy the overlay network of the infected computers and thus break the control of the attacker. Although only a disinfection of the infected computer can completely remove the malware containment and attacks on the communication infrastructure of the malware network can prevent the disclosure of private user data as well of attacks and spam mails sent by the infected computer.

In this experiment, the communication between the malware instance and its control server could be captured in a secure way. The analysis revieled information like the address of the control server and the protocol that is used to communicate.

[11] PROXMOX VE is a product of Proxmox Server Solutions GmbH, see http://pve.proxmox.com
[12] Tinc is a VPN software project by Tilburg university, see http://tinc-vpn.org
[13] http://dswd.github.com/ToMaTo/presentations/malware_euroview2011.html

2.9 Identity Management

The user management is an important part of the experimental facility and the project because G-Lab is a closed environment in contrast to comparable infrastructures like PlanetLab. While PlanetLab is open to everyone who joins the project with at least two systems, G-Lab is only open to registered users of the G-Lab project. Especially the organization of the identity of a user and his access rights is a critical issue in public available experimental facility design. In case of the G-Lab project the user management is necessary in two different areas, the infrastructure services, and the testbed platform itself.

The infrastructure services consist of the internal and external project documentation area, mailing lists, help desk, and software management. The testbed itself can be divided into a management and experimenter view. The experimenter requires access to the nodes and testbed resources on several layers. As standard software in G-Lab, the PlanetLab environment is used, also for the management of access rights. For deploying and operating specialized images a central account management is provided.

The administration of the users and system resources is done by a distributed administration team organized as a subproject of the overall G-Lab project. Each site might have some equipment, but at least users for the facility equipment. The approach distributes the responsibilities for the users assigned to a specific site to a representative of this site. This procedure requires additional role and access rights assignments for an extended group of identities. For example the headnodes, the node management and monitoring, and the private PlanetLab node administration are typical tasks, which are delegated to site representatives. Also a site representative has to organize the experiments and the resource usage of that site.

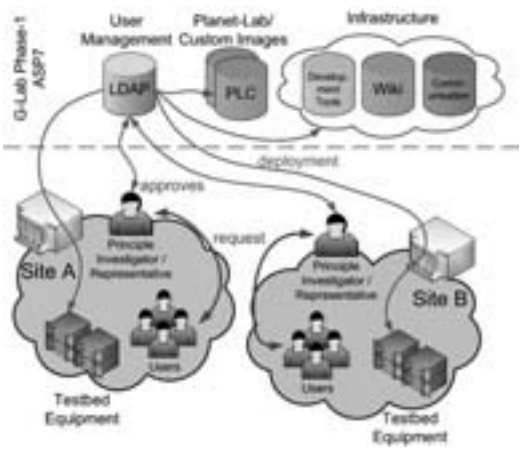


Figure 5: ToMaTo graphical front end

Figure 4 shows an architectural overview of the technical structure of the G-Lab identity and role dependency management. In general a central LDAP server stores the user's identities in a separate subtree, which is suborganized in subtrees containing the users of a specific site. A basic rule is that an identity is not associated with any access rights. This is organized in a separate tree, the so called group tree. Each service is represented by a unique group, which grants its members access to this server. A third separate subtree organizes virtual identities on machine level, so that each site has its own system level access user. This enables a fine grained and easy manageable environment on site level, even in case of changes. For services like the private PlanetLab installation account synchronization will be realized, so that the central LDAP database serves as master environment.

This can easily be extended to future services, if required. The management of the central database is done by a set of scripts, which respect a set of defined default roles for specific tasks. Also these scripts verify the integrity of the stored user data.

3 International Approaches

As mentioned above, the future of the Internet is a global issue which is tackled by activities around the world, for example PlanetLab, GENI (Global Environment for Network Innovations, US) and JGN2plus (Advanced Testbed Network for R & D, Japan), just to name a few. On a European scale, there are several national initiatives and projects which have similar or complementary objectives as G-Lab has. There are projects in France RNRT (French Research Network for Technological Innovation), ICT SHOK (Finnish Future Internet Research Programme), Ambient Sweden, Internet del Futuro (Spain).

But also under the umbrella of the European Commission within the 'Cooperation' program theme Information and Communication Technologies (ICT) of the 7th Framework Programme FP7 (2007 – 2013) of the European Community the "Future Internet Research & Experimentation" FIRE program was launched[14]. The FIRE initiative creates an open research environment, which facilitates strategic research and development on new Internet concepts giving researchers an instrument to carry out large-scale experimentation on new paradigms, across all levels and layers. It is the aim of the FIRE Facility to support Future Internet research in Europe, in a sustainable demand-driven way, which is independent from the program, the research is funded under. The Projects PanLab and OneLab, as well as their successors (OneLab2 and PII) are investigating possible organizational and business models for such a facility. As a result, FIRE will strengthen the competitive position of European research and industry in the important domain of Internet technologies and services. Figure 5 gives an overview about projects under the FIRE umbrella.

Starting from summer 2010, a second wave of projects with a budget of 50 million Euro is significantly expanding the scope of FIRE, moving it in new directions taking on technologies such as sensor networks, clouds and also high level service architectures. A more detailed description of some FIRE projects can be found in [FM10].

Another important project which must be mentioned here is the GENI[15] (Global Environment for Network Innovations) project which is a unique virtual laboratory for at-scale networking experimentation across the US under the auspices of the NSF (National Science Foundation). The GENI mission is to:

open the way for transformative research at the frontiers of network science and engineering; and inspire and accelerate the potential for groundbreaking innovations of significant socio-economic impact.

This project which is more or less an infrastructure oriented project comparable to the FIRE program is accompanied by a more research oriented program from NSF called

[14] http://cordis.europa.eu/fp7/ict/fire/fire-fp7_en.html
[15] http://www.geni.net/

NeTS[16] (Networking Technology and Systems). While GENI is comparable with FIRE and the experimental facility part of G-Lab the NeTS-projects can be compared with the more future internetworking architecture part of G-Lab.

While in the past all these projects were more or less focused on infrastructure and the core mechanisms of the Internet there are recent signs of a focus shift towards applications. In the European context a PPP[17] (public private partnership) program was launched last year (2011) which follows an industry-driven approach and works on research and development in network infrastructures, devices, software, service and media technologies. In parallel, it promotes its own experimentation and validation platform, bringing together demand and supply and involving users early in the research lifecycle. The new platform will thus be used by range of actors, in particular SMEs and Public Administrations, to validate the technologies in the context of smart applications and their ability to support user driven innovation schemes.

Comparable to the European PPP program the NSF started the US-IGNITE[18] program which is a Public-Private Partnership for the development of gigabit applications and services in areas of national priority[19] based on ultra-high speed (>100 Mbps symmetric) networks and deeply programmable (allowing new internet architectures and protocols), and sliceable (allowing isolated experiments or services running in parallel) network testbeds.

Last but not least, the AKARI[20] Architecture and Design Project in Japan should be mentioned. The goal of this project is to implement the basic technology of a new generation network by 2015, developing a network architecture and creating a network design based on that architecture. The overall philosophy is to develop an ideal solution by researching new network architectures from a clean slate without being impeded by existing constraints. Based on these new architectural ideas a new network will be designed and a migration path from today's conditions will be considered using these design principles. The overall goal is to create an overarching design of what the entire future network should be. To accomplish this vision of a future network embedded as part of societal infrastructure, each fundamental technology or sub-architecture must be selected and the overall design simplified through integration.

4 Conclusion & Future Work

On a technical level the German-Lab platform can currently be used to run different algorithms either using the PlanetLab software, in a virtualized system or in a custom system directly on the hardware. This provides maximal flexibility for experimenters and thus increases the usage of the platform. In the future the components of the platform will be integrated even more. Currently some efforts are under way to ensure the sustainability

[16] http://www.nsf.gov/funding/pgm_summ.jsp?pims_id=503307

[17] http://www.fi-ppp.eu/

[18] http://www.nsf.gov/cise/usignite/

[19] advanced manufacturing, health, education, energy, economic development, transportation, and public safety/emergency

[20] http://akari-project.nict.go.jp/eng/index2.htm

of the experimental facility which is more than generating some money to keep paying hardware, software and personal expenses. Long-term, organizational sustainability involves four main dimensions, including strategic, programs, personnel and finances. So for the near future we first take attention to the first three dimensions of sustainability, then financial sustainability is much more likely to occur – and much easier to accomplish.

In the past months there were several discussions, especially with industrial partners, in order to clarify whether such a platform could be used under commercial terms and conditions. It has been experienced that manufacturers are interested and forced by quality control services to test and verify their products in a "real" environment before bringing it into the market. This gives G-Lab as a developed platform extra importance in commercial market besides many infrastructure providers also shown the interest to test their product in a "post-IP" environment.

References

[BDF+03] Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T. L., Ho, A., Neugebauer, R., Pratt, I., and Warfield, A. Xen and the art of virtualization. In Scott, M. L. and Peterson, L. L., editors, *SOSP*, pages 164–177. ACM, 2003.

[Fiu06] Fiuczynski, M. E. PlanetLab: overview, history, and future directions. *Operating Systems Review*, 40(1):6–10, 2006.

[FM10] Fischer, S. and Müller, P. Experimentalforschung für das Future Internet - deutsche und europäische Initiativen. In *Springer Verlag*, February 2010.

[PBFM06] Peterson, L. L., Bavier, A. C., Fiuczynski, M. E., and Muir, S. Experiences Building PlanetLab. In *OSDI*, pages 351–366. USENIX Association, 2006.

[PR06] Peterson, L. L. and Roscoe, T. The design principles of PlanetLab. *Operating Systems Review*, 40(1):11–16, 2006.

[RM08] Reuther, B. and Müller, P. Future Internet Architecture - A Service Oriented Approach. In *In it - Information Technology, Volume 50, Number 6, 2008, Oldenbourg Verlag, Munich*, 2008.

[SGH+10] Schwerdel, D., Günther, D., Henjes, R., Reuther, B., and Müller, P. German-Lab Experimental Facility. In *Future Internet - FIS 2010*, 2010.

[SHG+11] Schwerdel, D., Hock, D., Günther, D., Reuther, B., Tran-Gia, P., and Müller, P. ToMaTo - a network experimentation tool. In *7th International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom 2011)*, April 2011.

[SRM11a] Schwerdel, D., Reuther, B., and Müller, P. Malware Analysis in the ToMaTo Testbed. In *Proceedings of EuroView2011*, 2011.

[SRM11b] Schwerdel, D., Reuther, B., and Müller, P. The Topology Management Tool - A demonstration. In *Next Generation Internet (NGI), 2011 7th EURO-NGI Conference on*, pages 1 –2, june 2011.