Reproducibility by Design: A Family of Testbeds for High-Precision Network Experiments

Position Statement

Sebastian Gallenmüller, Georg Carle *(intended GEFI 2019 participant)*
[gallenmu | carle]@net.in.tum.de, Technical University of Munich, Germany

ABSTRACT

Reproducibilification, i.e., making experiments reproducible, is the ultimate goal for successful scientific experiments. In this work, we identify key challenges for the design of reproducible network experiments. We present our approach for reproducible network research which enforces an experiment workflow leading to inherently replicable network experiments. Our approach realized in our testbed infrastructure combines high-precision measurement tools, full automation, and support for publishing experiment scripts and results.

We further present ongoing work, including extending high precision traffic generation and measurement capabilities for 100G Ethernet. Future plans involve the creation of a multi-site wireless testbed, which connects our testbed infrastructure with different remote testbeds, thereby creating a federated testbed. This federated testbed can be used for scenarios combining 5G Radio Access Network infrastructure with high-performance backbone infrastructure to investigate low-latency communication and edge computing use cases.

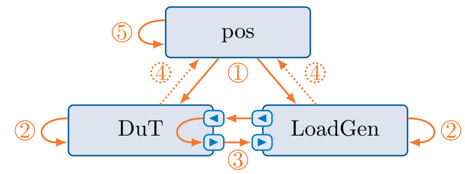

Figure 1: Testbed workflow

(a) Rack-testbed **(b) Wifi-testbed**

Figure 2: Currently available testbed deployments

1 KEY CHALLENGES

Reproducible network research is part of our research agenda for more than 15 years, see for example the 2003 Workshop at ACM SIGCOMM on Models, Methods and Tools for Reproducible Network Research co-organized by Georg Carle [1], in which the community identified a first set of challenges. We also participated at the Reproducibility Workshop at ACM SIGCOMM 2017, and share the view of Nussbaum [8] concerning key challenges for the design and execution of reproducible network experiments.

Precision: The ability to accurately perform detailed observations of elementary packet processing functions with fine time resolution is a key factor when investigating the behavior of network devices. In the past, we investigated the changing behavior of software packet processing systems for different traffic characteristics, with other parameters such as the average packet rate remaining unchanged [4]. It is important that testbed infrastructure offers facilities to combine flexibility in traffic generation with precisely controlling key features of traffic characteristics at a low level. At the same time, the measurement capabilities must be precise to detect the impact of changes. In particular timestamping of packets with ns precision at packet rates of 10 Gbit/s and above is challenging.

Configuration & setup: Modern computer architectures consist of a multitude of inter-operating hardware and software components, each offering a certain level of configurability, e.g. BIOS, OS, NIC hardware, NIC driver, and investigated applications. In case of virtualization, the software stack becomes even more complex. Minor changes to the configuration of a single component may influence experimental results significantly. Therefore, the configuration of all relevant systems of an experimental infrastructure and the documentation is a key challenge for reproducible network experiments.

Publication: Typically, preparation and release of results of experiments can be done with a limited amount of additional effort for the author. However, pure results do not contain all information necessary to fully understand or reproduce an experiment. Therefore, instructions on how to set up and perform an experiment, or auxiliary tools like experiment or plotting scripts are needed additionally. Despite their high value for the scientific community, these additional artifacts are rarely provided. One hindering reason is the time-consuming process for the individual researcher to prepare such artifacts. It is important that testbed infrastructure supports the creation and publication of data and accompanying information in a non-time consuming way.

2 CURRENT STATE

As we did not identify an available testbed that addressed all relevant challenges in a convincing manner, we created our own testbed infrastructure to solve the identified key challenges. Fig. 1 shows a typical workflow of a network experiment in the testbed, consisting of 5 steps. In this example, three servers are involved: the management host pos (plain orchestrating service) coordinating the experiment workflow, the load generator (LoadGen), and the DuT (Device under Test). Step 1 deploys and configures the images, Step 2 configures both experiment hosts, Step 3 runs the actual experiment, Step 4 collects the data, and Step 5 evaluates the data.

The fixed-network infrastructure, part of it depicted in Fig. 2a, consists of more than 20 servers equipped with various CPUs, different Intel NICs (X520, X540, X710, XL710, X722), Netronome SmartNICs (10G, 25G), and several switches, including 3 Tofino P4 switches. Network experiments performed in the testbed resulted in a fairly large number of publications, addressing software-defined networking, virtual machines, firewalls, distributed ledger, and secure multiparty computation. The wireless infrastructure (Fig. 2b) is equipped with shielded boxes (2.4 and 5.0 GHz) to perform replicable wireless experiments.

Precise measurements: For solving the problem of flexible and precise traffic generation we developed MoonGen [6]. Relying on DPDK, MoonGen can create data rates of 10 Gbit/s and above per core. It further employs techniques to precisely control the inter packet gap of generated traffic patterns. MoonGen's capabilities exceed the precision of other software packet generators [4]. It employs hardware

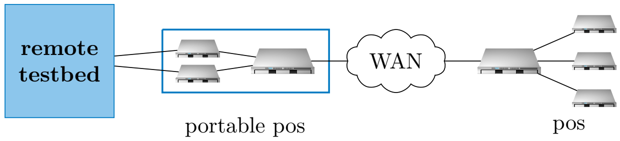

Figure 3: Federated testbed connected via portable pos

timestamping capabilities of modern NICs to timestamp packets with nanosecond accuracy. Our testbed infrastructure is also equipped with passive optical TAPs, used for timestamping every packet in hardware. This allows us to evaluate all packets entering and leaving a device under test.

Automated measurements: Before executing an experiment, all involved devices are rebooted to always start from a well-defined state. The well-defined state is guaranteed as devices in our testbed use Linux live images which do not keep any state between reboots. This prevents residual configuration from previous experiments impacting the current experiment. As the state between two experiments is lost after rebooting, automated configuration becomes a mandatory step in our experiment workflow. Forcing testbed user to script each step of an experiment leads to experiments with a *repeatable* outcome. Providing access to experiment scripts and to the testbed infrastructure simplifies *replicating* results by other researchers. This methodology makes experiments inherently replicable. We call this property *replicability by design*, which we see as one component to foster the way towards reproducible research supporting initiatives such as the ACM reproducibility badges [2]. Publishing data from experiments together with experiment scripts encourages others to *reproduce* one's results. An example of our published reproducible experiments is [7].

3 ONGOING ACTIVITIES

So far, our published results mainly were based on 10 and 40G Ethernet. With the shift of Ethernet towards higher bandwidths, we are currently in the process of creating measurement capabilities of 100G and more in our testbed. The advent of more flexible packet processing hardware with SmartNICs and P4 switches creates new possibilities for creating measurement tools. Projects like HyperGen and FLOWer [5, 9] show how programmable switches can be used to create 100G packet generators with precise timestamping capabilities.

Besides supporting the raw bandwidth requirements of future networks, realistic network behavior is required. We already published Internet-scale data sets in the past [3], which we currently analyze for the relevant statistical properties. Knowledge from both areas—testbed-scale and Internet-scale measurements—are the foundation for testbed tools, currently under development, to be able to replicate the behavior of these data sets in our testbed.

4 PLANNED ACTIVITIES

Our testbed already supports a wide range of devices. By creating a portable version of our testbed, we want to connect our testbed to other testbeds offering novel capabilities, including 5G radio technology, and highly-specialized wireless technology.

Federated testbeds: In its current version our testbed infrastructure is installed at several rooms of our building. Fig. 3 shows our plans for a federated testbed, connecting our testbed on the right to a collaborating testbed on the left of the figure. Our portable testbed infrastructure will be located at the site of collaborating testbeds to act as connecting element between the different testbeds. It creates a tunnel back to our original testbed, making the experiment hosts available to our local infrastructure including all tools that support the experiment workflow and postprocessing.

This federated architecture allows for additional experiments not possible with the current infrastructure. A possible scenario for collaboration is to connect a 5G Radio Access Network to our local testbed, which acts as cloud data center infrastructure. Our highly precise timestamping capabilities—in the portable testbed and the "cloud" testbed—facilitates 5G Ultra-Reliable Low-Latency Communication experiments. The portable equipment also can be used as edge computing infrastructure, thereby allowing to investigate edge-to-cloud data center deployment strategies. Reproducibility is especially challenging for wireless network experiments. Our wireless infrastructure includes RF-shielded boxes for eliminating the environment from Wi-Fi experiments, to always create a replicable scenario. Pairing this highly predictable RF-shielded environment with attenuators and a software-defined radio (SDR) allows for replicable experiment series, ranging from best-case to impaired scenarios, with the SDR acting as a deterministic replicable source of interference.

Virtual testbeds: We are convinced of the usefulness of approaches that provide quick and easy access to experimental facilities. To support further take-up of our workflow and tools by others, we are in the process of creating an open-source virtual clone of a testbed, which can act as a low-effort entry playground for scientists to perform reproducible experiments with a fully-featured testbed, without actually requiring all the resources on a physical testbed. From the software perspective, the goal is that the virtual testbeds behave as similar to the original testbed as possible (but of course with scaled-down performance). Ideally, this similarity creates experiment results comparable to the ones in the real testbeds. We want to investigate to which extent it is possible to create transfer models that describe the relation between experiments conducted on a real testbed and its virtual twin.

5 COLLABORATION EFFORTS

We are strongly interested in collaborations, offering our expertise for creating replicable measurements and sharing our knowledge on high-precision measurements. We intend to use our highly flexible portable testbed approach to federate our experiment infrastructure with other testbeds. Our interest is not limited to the scenarios described above, but we are also open to new ideas and approaches to be explored using such federated testbeds. Our second activity involving virtual testbeds aims to combine knowledge from different backgrounds. It supports collaboration by different research groups for linking knowledge in modeling, virtualization, and low-level testbed architecture.

REFERENCES

[1] ACM. 2003. Workshop on Models, Methods and Tools for Reproducible Network Research, organized by Georg Carle, Hartmut Ritter and Klaus Wehrle. http://conferences.sigcomm.org/sigcomm/2003/workshop/mometools/. (2003).

[2] ACM. 2018. Artifact Review and Badging. (2018). https://www.acm.org/publications/policies/artifact-review-badging.

[3] J. Amann, O. Gasser, Q. Scheitle, L. Brent, G. Carle, and R. Holz. 2017. Mission Accomplished? HTTPS Security after DigiNotar. In *Internet Measurement Conference (IMC), IMC Community Contribution Award, IRTF Applied Networking Research Prize (ANRP) 2018*. London. raw data: https://mediatum.ub.tum.de/1377982.

[4] P. Emmerich, S. Gallenmüller, G. Antichi, A. W. Moore, and G. Carle. 2017. Mind the Gap – A Comparison of Software Packet Generators. In *ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS 2017)*. Beijing, China.

[5] P. Emmerich, S. Gallenmüller, and G. Carle. 2016. FLOWer—Device benchmarking beyond 100 Gbit/s. In *IFIP Networking Conference (IFIP Networking)*. IEEE, 109–116.

[6] P. Emmerich, S. Gallenmüller, D. Raumer, F. Wohlfart, and G. Carle. 2015. MoonGen: A Scriptable High-Speed Packet Generator. In *Internet Measurement Conference (IMC), IRTF Applied Networking Research Prize 2017*. Tokyo, Japan.

[7] S. Gallenmüller, J. Naab, I. Adam, and G. Carle. 2019. Reproducibility example. (2019). https://gallenmu.github.io/low-latency/.

[8] L. Nussbaum. 2017. Testbeds support for reproducible research. In *Proceedings of the reproducibility workshop*. ACM, 24–26.

[9] Z. Xi, Y. Zhou, D. Zhang, J. Wang, S. Chen, Y. Wang, X. Li, H. Wang, and J. Wu. 2019. HyperGen: High-Performance Flexible Packet Generator Using Programmable Switching ASIC. In *ACM SIGCOMM Conference*. ACM, 42–44.