Packet Tracking in PlanetLab Europe – A Use Case

Tanja Zseby[1], Christian Henke[2], and Michael Kleis[1]

[1] Fraunhofer Institut FOKUS,
Kaiserin-Augusta-Alle 31, 10589 Berlin
{tanja.zseby,michael.kleis}@fokus.fraunhofer.de
[2] Technical University Berlin
Straße des 17. Juni 135, 10623 Berlin
c.henke@tu-berlin.de

Abstract. Making observations is fundamental for experimental research. Experimental facilities have to provide sophisticated and flexible tools to support scientific experiments by logging experiment results and monitoring environment conditions. Standardized measurements in experimental facilities can also provide consistent input for experiments with adaptive algorithms. In this paper we present a packet tracking architecture for PlanetLab Europe. Packet tracking correlates information from multiple observation points to observe the packet's path and experienced transmission quality. In our approach we use statistical sampling to control the measurement resource consumption and show how data selection processes at multiple observation points can be synchronized. As an example use case we show how the packet tracking architecture is used to support experiments on functional composition.

Keywords: packet tracking, multipoint measurement, hash-based packet selection, IPFIX, PlanetLab Europe.

1 Introduction

The current patchwork of protocols and the increasing demands on performance, flexibility and security in the Internet have led to a variety of programs on Future Internet research worldwide. They encourage, besides incremental solutions, new networking paradigms that follow a "clean state" approach, i.e. disruptive approaches that reconsider the IP protocol stack and try to find new networking designs. Functional Composition approaches revolve around the idea to functionally decompose the protocol stack and reorganize protocol functions in a composition framework in order to simplify the integration of new functionalities. Additional programs supplement theoretical research by large scale experimentation. Examples are the European Future Internet Research & Experimentation (FIRE) program and the US Global Environment for Network Innovations (GENI) program.

Experimental research is an essential building block of scientific work in order to proof a theory, investigate effects, and test new methods under real conditions before applying them in infrastructure and production environments.

One of the main enablers for thorough experimental research is measurements. As in other scientific disciplines, measurement and observation tools are needed to

T. Magedanz et al. (Eds.): TridentCom 2010, LNICST 46, pp. 265–274, 2011.

capture experiment outcome and to log experiment conditions. In [1] scientists from other disciplines strongly advise network researchers to take more care about measurements. They were shocked how limited the capabilities are to measure traffic in the Internet and how little effort is made in providing good observation tools. This advice comes from scientists from disciplines like biology, physics and astronomy who have a much harder time measuring their subjects of research hidden in human organisms, tiny atoms or far away in outer space.

In this paper we present an architecture for packet tracking that we implemented in PlanetLab Europe. We describe its benefits for the Future Internet experimenters in an example of a functional composition approach on application layer.

2 The Need for Measurements in Experimental Facilities

Scientists who perform experiments usually want to prove a theory, investigate a phenomenon or compare their own approaches to others. Observation tools in an experimental facility should be flexible in the sense that experimenters are able to zoom in or out to investigate traffic with different granularities. Furthermore, the tools should provide information about the accuracy of result data to provide comparability with results from others. Measurements in experimental facilities for Internet research can serve three different purposes:

- ***Experiment Supervision:*** Experiment supervision captures the results of the experiments, i.e. the input and output parameters under investigation, and provides them to the experimenter.
- ***Environment Supervision:*** Environment supervision captures further parameters not directly under investigation that may or may not be relevant for the experiment outcome. Especially parameters that cannot be controlled during the experiment but may influence the results need to be monitored.
- ***Measurement Service:*** Besides the capturing of results and conditions, the measurement tools can provide input to algorithms that require measurements for operation (routing, adaptation, learning). Providing a general measurement service allows researchers to use a common basis and eliminates the need to develop and deploy their own proprietary measurement solutions. Common input formats also support the comparability of results from different algorithms.

PlanetLab runs as an overlay over classical Internet connections where traffic from experiments interferes with many other flows. The underlying network, environmental conditions, physical effects, weather, etc. cannot be controlled by the experimenter. It is exactly such real unpredictable conditions, with potential undiscovered side effects, that make experimental research attractive. An ideal controllable environment would be unrealistic; simplified controllable settings would not differ much from a simulation. In uncontrollable environments a good documentation of experiment conditions is crucial. We may not be able to repeat all experiment conditions, but a good documentation helps to investigate observed side effects and analyze potential correlations if results differ from theory.

Our packet tracking architecture can substantially provide such measurement information as we will show based on the use case in Section 5. Path and delay information is relevant for a variety of experiments in the area of routing, overlay construction and optimization, congestion control and for any experiment with adaptive algorithms that require QoS values as input.

For measurements in federated testbeds it is also desired to send measurement results between different administrative domains. For this it is important that result data transfer uses protocols that support congestion control and encryption.

3 Related Work

PlanetLab is a global experimental platform for large scale experimental research. Slices that allocate resources on Planetab nodes are assigned to researchers who can use them for their experiments. The status of PlanetLab is reported by several programs and tools that view different aspects of the nodes and their connectivity.

Everstats [14] collects information from the slicestat [15] program running on each PlanetLab node which provides slice-level resource consumption information on each node. **CoMon** [16] provides the status of all PlanetLab nodes. It supports node-centric view statistics and a slice-centric view to see how a slice is consuming resources. **PlanetFlow** [20] is a flow measurement tool that logs the outbound network activity of all nodes. It is based on fprobe, which is a data collector using the Netflow format. For the ORBIT (wireless) testbed a framework is provided for control, measurement and resource management. For this the **ORBIT Management Framework** (**OMF**) uses the ORBIT Measurement Library (**OML**) to collect any type of measurement into a database. Everstat, CoMon and PlanetFlow trace activities and collect information on PlanetLab nodes. They provide user defined views but not user defined measurements. GIMS is still in the phase of specification. OMF already proposes measurements defined by applications and may be a good candidate for integrating packet tracking functions in wireless networks. To support measurements in GENI it is planned to establish a **GENI Instrumentation and Measurement Service (GIMS)**. The group has identified challenges with regard to data archiving, privacy issues, and the separation of measurement data from experiment data.

PlanetLab Europe is a European part of the worldwide experimental platform PlanetLab administered by the PlanetLab Europe Office in Paris. The European Project OneLab [2] works on the provisioning of highly sophisticated control and monitoring functions to support the needs of experimental-driven research in PlanetLab Europe. The project integrates further technologies like wireless testbeds and platforms for disruptive Future Internet research like autonomic communication, delay tolerant networks and pocket switched networks. The project is also working on solutions to federate with other experimental facilities.

The OneLab project has developed a measurement solution that integrates passive and active measurements. The **Advanced Network Monitoring Equipment (ANME)** is specified in [10]. For passive measurements it uses the Continuous Monitoring (CoMo) platform [11], [12] which allows several experimenters simultaneously to perform arbitrary traffic queries on the data streams on their slices. CoMo supports user defined modules and resource control for measurement tasks. The mapping between PlanetLab slices and their traffic, and access to their relevant measurement data is controlled via a proxy described in [13]. The proxy provides a clear separation of experiments and ensures that experimenters can only access results of their own experiments. The deployment of the ANME and CoMo software in PlanetLab Europe is currently in progress. The service can be accessed at www.packet-tracking.de.

The presented packet tracking solution will be integrated as modules in the CoMo framework in order to use the CoMo and ANME control functions for experiment control, result data access and GPS-based time synchronization.

4 PlanetLab Europe Packet Tracking Architecture

4.1 Multipoint Packet Tracking Architecture

Following packets on their path requires a passive multipoint measurement architecture. This architecture consists of multiple measurement probes and a collector that correlates the probes' measurements. The measurement probes are located at observation points which can be any device that has a connection to the shared medium, such as router or network card. The probes export a packet ID and either the TTL or an arrival timestamp for each observed packet to the collector. Based on the packet ID the collector can correlate the observations and determine the packets' direction by the TTL or timestamp. Packet tracking can also be used for calculating one-way delay between the observation points, but this imposes additional requirements such as a common time base on the architecture. Passive delay measurements will be made available when the architecture is integrated in CoMo in the GPS-enabled ANME boxes.

In the following we present previous results on packet ID generation and hash-based packet selection as these are critical for proper packet correlation and sampling synchronization between measurement points.

4.2 Packet ID Correlation

In order to follow a packet's path through the network we need a unique identifier for each packet. The datagram identification field in the IP packet header was considered as ID in [3], [4] [5], but it is not sufficiently unique because of its limited 16bit size and different handling of the ID from OS (some OSs use an ID of 0). Another possibility is to generate a packet digest over packet header fields and part of the content. The packet ID generation should be done in a way that 1) the resulting ID is small to reduce measurement traffic; 2) the ID is fast to calculate to reduce resource consumption on the node; 3) the probability for collisions (getting the same packet ID for different packets) is low and 4) the digest is calculated over packet parts that do not change on the way.

Fields that are mutable but predictable could also be used for packet ID generation. Furthermore, it is advantageous (but not required) to use an operation that always leads to an ID with the same fixed length. This eases the handling, transmission and the estimation of the overhead caused by measurement export.

The amount of content used for ID generation influences the number of collisions (different packets that map to the same ID) that can occur. It is advantageous to consider fields that are highly variable between different successive packets; fields with low variability (e.g., version field) should only be included as long as there is no significant performance decrease. Investigations into this and comparisons of different packet ID generation methods can be found in [4], [5] and [6].

4.3 Coordinated Sampling by Hash-Based Packet Selection

Network traffic is highly dynamic and hard to predict. General challenges of passive measurements are unforeseen high data rates that can lead to an unexpected exhaustion of measurement resources. Data selection techniques, like sampling and filtering, aim at the reduction of measurement resources (processing, storage, transfer of data). This is achieved by selecting a finite subset of elements (the sample) and estimating the metric of interest from this subset. Especially random packet sampling techniques provide a solution to reduce the measurement traffic while still enabling an accuracy statement about the estimated characteristic. Nevertheless, if we apply random sampling, we randomly select a subset of the packets that traverse the observation point. This is a problem in multipoint measurements, because the random selection processes at the involved observation points would select different packets, making packet matching impossible. So if we want to apply random packet sampling at multiple observation points we have to somehow ensure that the same packets are selected at the involved observation points. We therefore use hash-based packet selection as proposed in [5] and [7] to emulate a random selection and synchronize selection processes at different observation points.

The packet selection is based on a hash function on invariant packet header fields and parts of the payload. If the same hash function and the same hash selection range is used at all involved observation points then packets selected at one observation point are also selected at the other observation points.

A problem with hash-based packet selection is that it is a deterministic function on the packet content which can introduce bias into the selection. In [6] we investigated several hash functions with respect to the achievable random properties and their suitability for multipoint measurements.

Due to the differing requirements for packet ID generation and packet selection [4], [5] and [6] recommend using two different functions for packet ID generation and packet selection. Nevertheless in [7] we could show that for small to medium sized measurement domains it is reasonable to use the same hash function.

Since PlanetLab Europe is a large federated testbed with distributed measurements, we decided to use a CRC32 function for packet ID generation and the BOB function as evaluated in [7] for hash-based packet selection.

4.4 IPFIX and PSAMP

In experimental research, measurement data provides the basis for the formulation of scientific results. It is crucial to agree on standard measurement methods and result data formats to be able to compare the outcome of experiments and to share data with others. Standardized formats simplify the development of tools (e.g., for data analysis) and allows the provisioning of reference data.

In our approach we utilize and exploit approaches for data export and data selection, which are developed with our contribution within the IETF working groups on IP Flow Information Export (IPFIX) and Packet Sampling (PSAMP). IPFIX is a protocol to export flow measurement results. It supports encryption of the transferred data and congestion control. Those features are essential for transferring results among different administrative domains e.g., in federated testbeds. IPFIX defines Information Elements (IE) [8] to report flow information. The PSAMP standard extends these Information Elements with additional fields for packet reporting and sampling. Our group in Fraunhofer FOKUS has developed an IPFIX Library which we used for exporting measurement results in our PlanetLab Europe packet tracking implementation. Since the PSAMP information model is required for the representation of packet IDs in multipoint packet tracking we extended our implementation to support PSAMP information elements.

We expect that in future standard routers will provide IPFIX data. Cisco Systems is working on IPFIX as successor to Cisco NetFlow. As a result of this, data from PlanetLab Europe (PLE) experiments become comparable to data recorded in other networks. The IPFIX group is also working on a file format to store packet and flow data. We contributed to this standardization effort [9] because we believe that this is an important step towards sharing data from experimental research among researchers and provides the right way towards re-usability of analysis tools. We plan to use the file format for storing packet tracking data in PlanetLab Europe.

5 Use Case: Functional Composition on Application Layer

Cooperative Service Provisioning (CSP) [18][19] is a service composition architecture that utilizes a decentralized approach for service path discovery in order to combine services at customer's premises. CSP supports functional composition on the service layer, which we will present as a use for packet tracking.

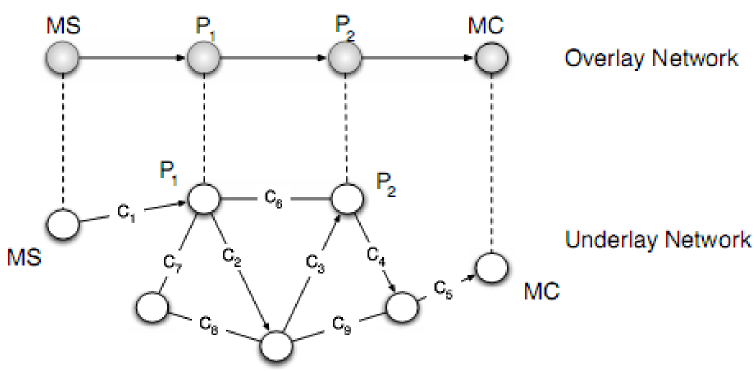

Fig. 1. Overlay Based Service Composition. A media server (MS) publishes a stream that the media client (MC) wants to receive. CSP dynamically creates a service path which satisfies MC's requirements by incorporating services offered by other peers.

In a collaborative P2P environment peers may offer services like caching and transcoding for other peers that do not have the resources to watch a media stream in the originally offered format, e.g., a mobile client peer that wants to watch a high definition video but only features QCIF. CSP organizes the peer nodes and their offered services in a structured distributed hash table (DHT) CAN [18] address space which enables a service path composition that satisfies the clients' requirements (see Figure 1). Because CSP is organized in a DHT it is also failure resilient. When a peer leaves the Overlay, another peer becomes responsible for its share in the CAN.

The research challenge here is the establishment of the function chain. In an environment where multiple nodes offer services that can be included (e.g., caching, transcoding, encryption, etc.) the algorithm needs to find the required functions (based on application demands) and combine them in a reasonable manner.

Since service composition for realtime media services and also for functional composition is QoS sensitive, it has to be ensured that the QoS constrains are not violated by the service composition process. As a consequence service composition resembles a (Multi-) Constraint Routing problem [17] where the construction of the function chain already becomes NP complete.

In previous work we investigated algorithms for the composition [18][19] and made a first simulation implementation on a few nodes. Since results are dependent on characteristics of the underlying network, we planned to test the algorithms in a realistic testbed. We decided to use PlanetLab Europe for the experiments because the traffic runs on real Internet connections and we are able to use nodes that are highly distributed around the globe. The ability to use many highly distributed machines is essential for the investigation of the composition algorithm that depends on the number of available functions and the QoS parameters between nodes.

In our experiments we investigate whether the function chain is correctly established and the packets traverse the network on the expected paths. The packet tracking service here is required for experiment supervision and providing path information. In future packet tracking can also be used to provide input data for overlay establishment and optimization (as a measurement service). It can also log environmental settings because adjacent traffic from other applications in the network will also be measured. In this way one can also examine the influence of different flows within the network.

6 Measurement Setup for Packet Tracking

Basically Planetlab nodes are end hosts which can directly communicate over the Internet. This means that none of the PlantLab nodes serves as a router and will forward packets. Therefore we created an overlay network on nodes in our slice in a way that at least one intermediate node does packet forwarding. We created virtual tap interfaces with their own addressing space upon the original (eth0) interfaces on nodes in our slice. These virtual tap interfaces are interconnected using socat tunnels. We are able to create multiple virtual tap interfaces, but forwarding between those interfaces is not yet supported in PlanetLab, although planned. Until this feature is enabled we are using one intermediate machine in our facilities that connects with the tap interfaces on the PlanetLab machines, creating a star topology with our machine in the middle. Our machine is able to forward packets to the PlanetLab nodes and work as an intermediate router (cf. Figure 2). In addition, we deployed OpenIMP [21] measurement probes on all involved nodes (PlanetLab and our middle machine).

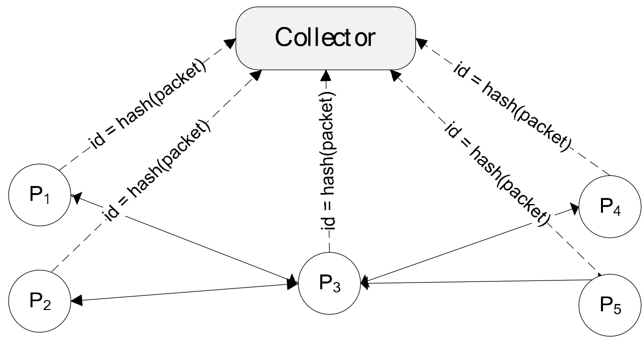

Fig. 2. Each PlanetLab node has a probe that applies hash-based selection, generates a packet ID for each selected packet and sends IPFIX results to a common collector. The nodes are interconnected using socat and a central node that forwards packets.

For performing accurate passive multipoint measurements it is necessary to separate measurement traffic from the measured traffic, i.e., in a packet tracking architecture the IPFIX measurement result traffic should not be observed. Therefore we export our measurement traffic over the original eth0 interface while the probe listens on the virtual interface. This setup is not recommended for delay measurements because the measurement traffic and real traffic is actually exported over the same (real) interface which can distort delay measurement. In a first demo scenario we used iperf to generate traffic between 5 nodes (KTH Stockholm, ELTE Hungary, Quantavis Pisa, Porto and University of Athen) and our node in Berlin in the middle.

7 Packet Tracking Visualization

Packet Tracking is very graspable; everyone can intuitively imagine how a packet traverses its path. We implemented a Java Application which uses OpenStreetMap and an animation framework to visualize the measurement data. The observed packet tracks are aggregated in user-defined intervals and displayed as animated light dots where each dot represents multiple packets. For the path visualization we used cubic splines so that the path from Point A to B does not cover paths from Point A to B to C, i.e. one can follow the packets path from ingress to egress. An example screenshot is shown in Figure 3.

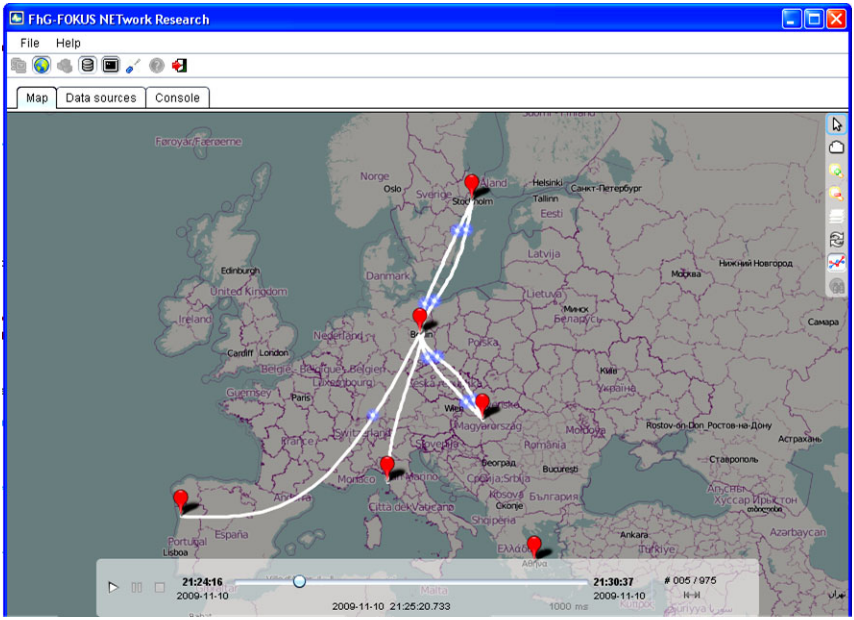

Fig. 3. Screenshot of Visualisation Tool for Packet Tracking in PlanetLab

8 Conclusion and Acknowledgements

We introduced a packet tracking architecture for PlanetLab Europe for experiment and environment supervision and as a general measurement service for algorithms that require such input. We showed how packet tracking with multiple observation points and coordinated data selection techniques can be realized. We presented a typical use case and showed how results can be visualized. We plan to integrate the service with the OneLab Advanced Network Monitoring Equipment (ANME) and to extend the architecture for passive delay measurements and result storage in IPFIX file format. We also plan to integrate packet tracking functions in the recently federated wireless networks of PlanetLab Europe.

This work was partly funded by the EU project OneLab2 under the FIRE program. We thank Joachim Kaeber for the PlanetLab Europe setup and Tacio Santos for implementing the visualization. Special thanks go to the PlanetLab Central and Europe support team Guthemberg Da Silva Silvestre, Anil Kumar Vengalil, Benjamin Quétier, Sapan Bhatia and Thom Haddow for quick responses and technical support.

References

1. Committee on Research Horizons in Networking, Computer Science and Telecommunications Board (CSTB). Looking Over the Fence at Networks: A Neighbor's View of Networking Research (2001). National Academy Press, Washington, D.C (2001)
2. OneLab, `http://www.onelab.eu/`

3. Graham, I.D., et al.: Nonintrusive and accurate measurement of unidirectional delay and delay variation on the Internet, INET (1998)
4. Zseby, T., Zander, S., Carle, G.: Evaluation of building blocks for passive one-way-delay measurements (April 2001)
5. Duffield, N.G., Grossglauser, M.: Trajectory sampling for direct traffic observation. IEEE/ACM Transaction on Networking 9(3), 280–292 (2001)
6. Henke, C., Schmoll, C., Zseby, T.: Empirical Evaluation of Hash Functions for Multipoint Measurements. ACM SIGCOMM Computer Communication Review (CCR) 38(3), 39–50 (2008)
7. Henke, C., Schmoll, C., Zseby, T.: Empirical Evaluation of Hash Functions for Packet ID Generation in Sampled Multipoint Measurements. In: Moon, S.B., Teixeira, R., Uhlig, S. (eds.) PAM 2009. LNCS, vol. 5448, pp. 197–206. Springer, Heidelberg (2009)
8. Boschi, E., Trammell, B., Mark, L., Zseby, T.: Exporting Type Information for IPFIX Information Elements, RC5610 (July 2009)
9. Trammell, B., Boschi, E., Mark, L., Zseby, T., Wagner, A.: An IPFIX-Based File Format. RFC5655 (October 2009)
10. Vattay, G., Fekete, A., Hága, P., Stéger, J., Aracil, J.: Giuseppe Iannaccone. Specifications for advanced monitoring boxes, OneLab Deliverable D4.2 (November 2008), http://www.onelab.eu/index.php/results/deliverables.html
11. Iannaccone, G., Niccolini, L., Duca, A.L., Canonico, R., Zseby, T., Witaszek, D., Kaeber, J.: Packet tracking architecture, OneLab Deliverable D5.1 (May 2009), http://www.onelab.eu/index.php/results/deliverables.html
12. CoMo Project, http://como.sourceforge.org
13. Iannaccone, G.: Fast Prototyping of Network Data Mining Applications. In: Proceedings of PAM (March 2006)
14. Everstat, http://everlab.cs.huji.ac.il/planetstats
15. Slicestat, http://codeen.cs.princeton.edu/slicestat/
16. CoMon, http://comon.cs.princeton.edu/
17. Korkmaz, T., Krunz, M.: Multi-Constrained Optimal Path Selection. In: Proceedings of IEEE INFOCOM, Anchorage, Alaska, USA (2001)
18. Kleis, M., Radier, B., Elmoumouhi, S., Carle, G., Salaun, M.: A decentralised service composition approach for peer-to-peer video delivery. Journal Peer-to-Peer Networking and Applications (2009)
19. Kleis, M.: CSP, Cooperative Service Provisioning using Peer-to-Peer Principles, PhD Thesis Chair for Network Architectures and Services Department of Computer Science, Technical University Munich (2009) ISBN 3-937201-05-X
20. PlanetFlow, http://www.cs.princeton.edu/~sapanb/planetflow2/
21. OpenIMP, http://www.ip-measurement.org/index.php?id=8&Itemid=8