Deploying per-packet telemetry in a long-haul network: the AmLight use case

Jeronimo Bezerra*, Italo Brito*, Arturo Quintana*, Julio Ibarra*, Vasilka Chergarova*,
Renata Frez†, Heidi Morgan‡, Marc LeClerc§, and Arun Paneri§
*Center for Internet Augmented Research and Assessment
Florida International University, Miami, Florida 33199
Email: see https://ciara.fiu.edu/staff.html
†Rede Nacional de Ensino e Pesquisa, Brazil
‡University of Southern California, Los Angeles, California
§NoviFlow, Sunnyvale, California

***Abstract*—Long-haul networks are growing in complexity to address the constant need for more bandwidth, lower latency and jitter, customized traffic prioritization, and SLA-grade network resilience. A more complex infrastructure requires a deeper visibility of the assets to optimize the resource utilization as well as to protect the infrastructure and users connected to it. Leveraging legacy network monitoring technologies, such as SNMP, is not enough, since they do not offer real-time and granular visibility. That's where per-packet monitoring solutions can become a game changer. In-band Network Telemetry (INT) offers per-packet visibility with no impact to the network's forwarding plane. Adding per-packet visibility has the potential to change the network monitoring and operations field, and to redefine how traffic engineering will take place in the future. This paper aims to showcase how INT can dramatically increase network visibility, down to a sub-second scale. Experiments and findings come from using the AmLight long-haul academic network as a use case.**

***Index Terms*—In-band Network Telemetry, INT, Monitoring, Telemetry, Long-haul networks**

I. Introduction

In 2017, AmLight [1] was challenged to provide sub-second network monitoring and performance evaluation metrics to support real-time high-performance SLA-based science applications. At that time, network monitoring and performance measurement technologies were based on one of the following approaches: 1) sampling data from network devices on a regular basis, with protocols such as SNMP [2], 2) network devices pushing packet samples, such as sFlow [3], to a central collector, 3) packet inspection by leveraging port-mirrors and fiber taps to monitor data packets, 4) pushing probes to the network to evaluate the infrastructure, or 5) a combination of approaches. Each technology has pros and cons but, in the end, none of them is accurate or scalable enough to support sub-second network monitoring of a long-haul network infrastructure such as AmLight. In-band Network Telemetry (INT) [4] was identified as the most prominent solution to achieve sub-second network and performance monitoring at scale in a multi-100G network.

INT has the potential to transform how network operators produce and consume network telemetry by enabling per-packet network visibility at scale. To confirm our hypothesis, AmLight's data plane was instrumented to support INT. In 2018, when the project to support INT started, there were no production network devices in the market capable of supporting INT. Moreover, there were no software solutions available to parse per-packet telemetry reports. To enable INT in its data plane, AmLight partnered with NoviFlow[1]. NoviFlow is a networking software company specializing in scaling high performance network infrastructure though Software-Defined Networking (SDN). NoviFlow's network operating system (NOS), called NoviWare, is used for the INT project at AmLight. To collect and parse telemetry reports, AmLight initiated the development of its INT telemetry collector. Together, AmLight and NoviFlow worked on the production and selection of which telemetry data to export, and how to consume and generate useful network telemetry reports. From the network infrastructure perspective, the goal was to create a new infrastructure, capable of matching network flows to receive INT data, exporting and parsing telemetry reports, and generating high-level reports in a sub-second interval. From the network operation perspective, the goal was to evaluate the performance of real-time science applications, as well as validate traffic engineering polices and configurations.

The INT deployment at AmLight started in 2020. By deploying this new INT solution, AmLight became capable of processing per-packet network telemetry reports, generating transient alerts, and notifying the AmLight network orchestrator in less than 200ms. Currently, capable of parsing up to 2,000,000 telemetry reports per second, per telemetry collector node, AmLight's network operation has dramatically enhanced what it previously considered extremely time-consuming and complex activities: detecting microbursts, profiling buffer utilization per node, tracking every packet's path and notifying when there were changes, detecting link-aggregation hash mismatches, mitigating per-packet jitter and sources of packet drops caused by under-provisioned buffers.

This paper presents how AmLight's efforts to support INT paid off and how INT will potentially change the way network

[1]NoviFlow Inc.'s website: https://noviflow.com

operation and monitoring will be done in the future. The paper is organized as follows. In Section II we describe INT in detail and the architectural decisions made. Details of how AmLight deployed INT are presented in Section III. The evaluation and benefits of INT at AmLight are presented in Section IV. Finally, we conclude in Section V.

II. In-band Network Telemetry

In-band Network Telemetry (INT) is a P4-based technology that enhances network monitoring and visibility by enabling per-packet network telemetry. The INT specification [4] defines three application modes to support INT: (1) eXport MetaData (INT-XD), (2) EMbed with Instruc(X)ions (INT-MX), and (3) EMbed with MetaData (INT-MD). When supporting INT-XD, each INT node exports telemetry reports directly from the data plane, without changing the original ingress data packets. This application mode simplifies the data plane operations over the packets. However, the INT-XD mode increases the complexity of correlating telemetry data with the other network events at the telemetry collector node, because each INT node will send a separate telemetry report that the telemetry collector processes (for each INT node) to correlate an event. When supporting INT-MX, the INT Source node (first INT node in the path) adds an INT header with the instructions to be performed per INT node in the path. Each instruction describes which telemetry data the INT node should export to the telemetry collector. The data correlation at the telemetry collector node is as complex as it is for INT-XD. However, for the network operator, the activity of defining and configuring the instructions to be performed is simplified and limited to the INT Source nodes. For both INT-XD and INT-MX modes, complexity increase is a result of the INT Telemetry Collector receiving a stream of reports at (almost) the same time and creating a condition where there is no easy way to correlate which reports are related to a specific traffic flow (we found that comparing TCP/IP headers is not scalable). When supporting INT-MD, the INT Source node adds an INT header with instructions AND the telemetry data to the user packet. Each INT node in the path adds telemetry data to the user packet. The INT Sink node (last INT node in the path) extracts all telemetry data added and exports a telemetry report with all telemetry data to a central telemetry collector while forwarding the original user packet to the final destination. INT-MD enables the telemetry collector to receive a telemetry report per packet with network telemetry data from all INT nodes. As a result, there is no need for data reduction or correlation, since each telemetry report will provide node-by-node visibility. Figure 1 shows INT nodes (switches or routers) adding telemetry data across the network topology.

When INT is supported by the INT node's data plane forwarding technology (ASIC, chip, FPGA), for instance, when using the Intel Tofino ASIC [5], all operations for gathering and adding telemetry data to user packets, and generating telemetry reports are performed directly from the data plane without impacting the network device's forwarding functions. When supporting INT-MD, user packets carry per-node network telemetry without impacting the user application, even at 100Gbps. INT-MD is the default application mode when deploying INT, and it was the mode deployed at AmLight.

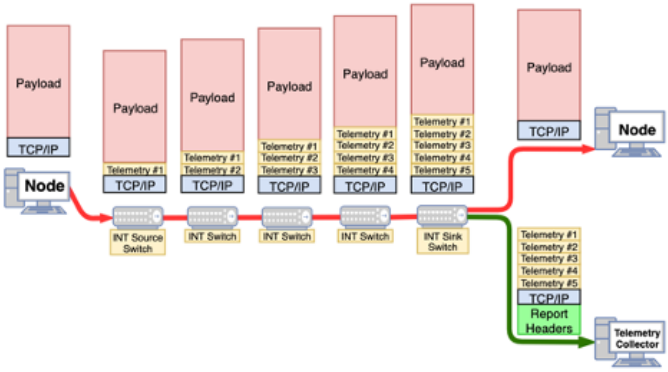

Fig. 1. INT-MD adding telemetry data per INT node.

Each INT network telemetry report provides a set of per INT-node network telemetry data and the user packet's TCP/IP headers. An INT network telemetry report includes the following data: per INT-node incoming and outgoing timestamps with nanoseconds accuracy, identifiers per INT-node incoming and outgoing interfaces, per INT-node outgoing interface's queue identification and the queue occupancy. With the incoming and outgoing timestamps, operators can compute how long a user packet was buffered in the switch before being forwarded, which is called *hop delay*.

Out Time: 123144143 ns	
In Time: 123132143 ns	
Queue: 2	Occ: 15MB
Hop Delay: 12 us	
In: Port 1	Out: Port 2
Switch: 1	
Out Time: 124145243 ns	
In Time: 124144143 ns	
Queue: 0	Occ: 10KB
Hop Delay: 1.1 us	
In: Port 1	Out: Port 4
Switch: 2	
Out Time: 125146343 ns	
In Time: 125145243 ns	
Queue: 0	Occ: 10KB
Hop Delay: 1.1 us	
In: Port 31	Out: Port 28
Switch: 3	
Out Time: 126147443 ns	
In Time: 126146343 ns	
Queue: 0	Occ: 10KB
Hop Delay: 1.1 us	
In: Port 12	Out: Port 13
Switch: 4	
Out Time: 127187443 ns	
In Time: 127147443 ns	
Queue: 0	Occ: 21MB
Hop Delay: 40 us	
In: Port 1	Out: Port 7
Switch: 5	

Fig. 2. INT telemetry reports data

With the outgoing interface's queue occupancy, operators can mitigate the packet drops probability and egress buffer utilization. Combining queue occupancy and hop delay, operators can identify microbursts affecting the network performance as well as sources of jitter. As each report provides the incoming and outgoing interface identifiers, network operators can track each packet's physical path throughout the network topology, leading to reliable packet tracing. With the interfaces' identifiers and the IP length provided in the IP headers, instantaneous interface bandwidth utilization can be calculated. Figure 2 represents the INT telemetry reports data stacked in a network with five INT nodes, as illustrated in Figure 1.

III. Enabling INT at AmLight

Enabling INT at AmLight was possible once NoviFlow expanded its NOS to support INT and AmLight developed its telemetry collector solution. As AmLight built its INT solution based on the INT-MD application mode, the first step was to evaluate the impact of adding an INT header to each packet. Field evaluation has shown that adding an INT header is performed by the INT Source node in less than 0.00045 milliseconds by the Intel Tofino ASIC. Since the AmLight network is designed on a multi-millisecond long-haul WAN topology, the 0.00045 milliseconds delay introduced by the INT Source node has no impact on AmLight users. Each INT node stacks up to 24 bytes of telemetry data and the solution was designed to support up to ten INT nodes stacking telemetry data. When ten switches add telemetry data, the user packet increases by 252 bytes (240 for telemetry data plus 12 for the INT header). The Intel Tofino ASIC supports an MTU of 10,000 bytes.

The INT node's NOS supports selective monitoring: the network operator can select which TCP or UDP flow to monitor based on many criteria. The telemetry reports generation is performed directly from the data plane, which results in zero impact to the INT node's host processor. And more importantly, it results in no impact to other network functions performed by the INT node, for instance, flow table updates. The INT Sink node adds to each telemetry report a timestamp, a sequence number, the INT Sink node identifier, and the original user packet's TCP/IP header. It is worth mentioning that the INT Sink node does not include the user's payload in the telemetry report.

The AmLight telemetry collector solution, named INT Collector, is responsible for receiving, parsing, processing, and generating operational reports. The solution was created with performance as the primary requirement. To illustrate the need for performance, a 40Gbps flow using 9,000-byte packets, generates almost 600,000 packets per second (pps). As each user packet triggers a telemetry report, the INT Collector is required to handle 600,000 pps or almost 1.4Gbps of telemetry reports on a topology with six INT nodes stacking telemetry data *(six is the average number of nodes between two points on the AmLight network)*. In a network topology with more than 1.2Tbps of aggregated international capacity and presence in many countries, the AmLight telemetry collector solution was created as a bundle that includes a physical 24-core 3.0GHz CPU and 128GB of RAM server, running GNU/Linux Ubuntu with Kernel version 5.8 leveraging extended Berkeley Packet Filter (eBPF) [6] and eXpress Data Path (XDP) [7], InfluxDB [8], and Grafana [9]. The INT Collector leverages eBPF/XDP due to its simplicity and performance. eBPF/XDP is native to the Linux Kernel, and its execution can be offloaded to the network interface card. InfluxDB and Grafana support the time-series data storage and data visualization, respectively.

Generating telemetry reports is the responsibility of the INT Sink nodes, since they are the last INT nodes in the path. At sites where AmLight has multiple INT Sink nodes, a 10G+ management network was created (Figure 3). All telemetry reports are redirected to the INT Collector node over the Management Network. This configuration avoids telemetry packets competing against user flows for bandwidth. Figure 3 illustrates how the management network was created in the U.S. and Chile.

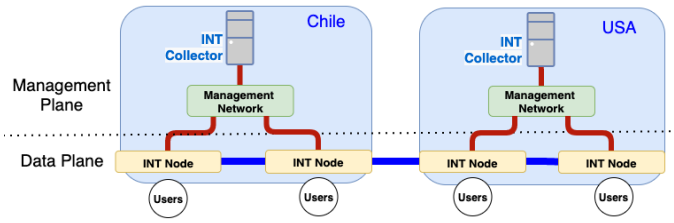

Fig. 3. AmLight management plane.

Section IV shows how the infrastructure presented in Section III is used in AmLight's daily network operation, including some measurements observed that emphasize the benefits of supporting INT in a production network.

IV. How does INT improve the AmLight network operation?

AmLight leverages the granularity and accuracy provided by INT to answer operations questions beyond the scale of for legacy monitoring solutions. As telemetry reports are triggered by user packets in real-time, monitoring the precise bandwidth utilization per interface and per interface's queue becomes possible. This new capability facilitates the identification of microbursts impacting QoS policies and real-time applications (Section IV-A). Leveraging per-packet queue occupancy information, AmLight operators know the current and average buffer utilization per interface's queue (Section IV-B), which allows them to fine-tune traffic engineering policies and configurations. When mitigating packet loss, the first step is to understand what was the path a user packet took during the time frame reported. With INT, monitoring the path taken is enabled per-packet, which helps narrow down the root cause of packet loss (Section IV-C). Each of the following subsections provides context of the current challenges and how AmLight leverages the INT solution to mitigate them.

A. Monitoring Instantaneous Bandwidth Utilization

With INT, AmLight is monitoring instantaneous bandwidth utilization per interface and per interface's queue. Using the telemetry report timestamp, the outgoing interface identification, egress queue id, and the IP length, bandwidth utilization can even be calculated between two packets. The INT Collector tracks the bandwidth utilization, and then saves that information every user-defined time interval, usually in millisecond time scale. At AmLight, the time interval varies from 100ms to 500ms and can be tuned up or down according to operational needs. Figure 4 shows the bandwidth utilization during an interval of 45 seconds. The orange graph shows bursts of traffic lasting less than 5 seconds with high accuracy. The accuracy enabled by INT grants AmLight a mechanism

by which to measure exactly when these bursts started and ended, as well as their total bandwidth utilization. Figure 5 shows in detail a 22-second bandwidth test performed by an AmLight 10G perfSonar [10] node.

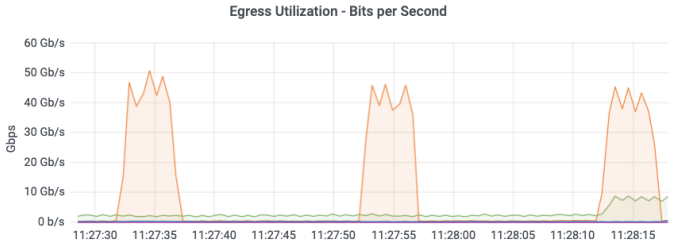

Fig. 4. Interface Egress Bandwidth Utilization.

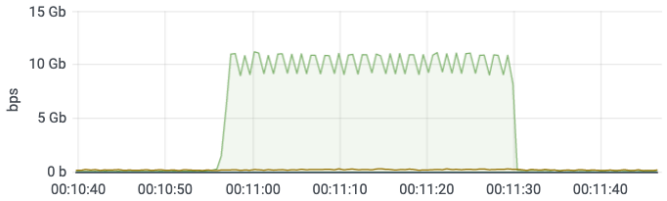

Fig. 5. A perfSonar BWCTL test at AmLight.

Microbursts usually last from milliseconds to a few seconds and can lead to jitter and packet drops. Many research groups have been exploring the challenge of mitigating microbursts by leveraging the new capabilities of programmable ASICs and chips, for instance [11], [12], and [13]. Microbursts are hard to detect when the network monitoring solution is based on polling data from the network devices, since the time intervals between data gathering happens from 30 to 300 seconds on production environments. Figure 6 shows how INT compares to SNMP during a microburst mitigation activity that took place on July 15th, 2021. Figure 6 has two graphs: the top graph shows bandwidth utilization of the INT node interface 11 using INT. The bottom graph shows bandwidth utilization of the neighbor Ethernet switch connected to the INT node on port 11. Both devices are part of the AmLight production network infrastructure. These graphs show interface utilization from July 15th, 11:26:40 AM to 11:28:35 AM, for less than 2 minutes. The SNMP poll interval was set to 15 seconds, since the Ethernet switch only updates the SNMP counters every 14 seconds. The INT graph (top) shows five bursts lasting approximately 5 seconds, each generating from 38 to 50Gbps. The SNMP graph (bottom) shows two bursts lasting 30+ seconds with peaks of utilization of 13Gbps. The INT graph is a more precise representation of network events, because packets are processed 1-to-1, whereas the SNMP graph is less precise, because packets are sampled on a 15 second interval.

B. Monitoring Packet Drop probability and Jitter

Section IV-A showed the benefits of using INT to have an accurate visualization of the bandwidth utilization. Although bandwidth utilization is an important component of the

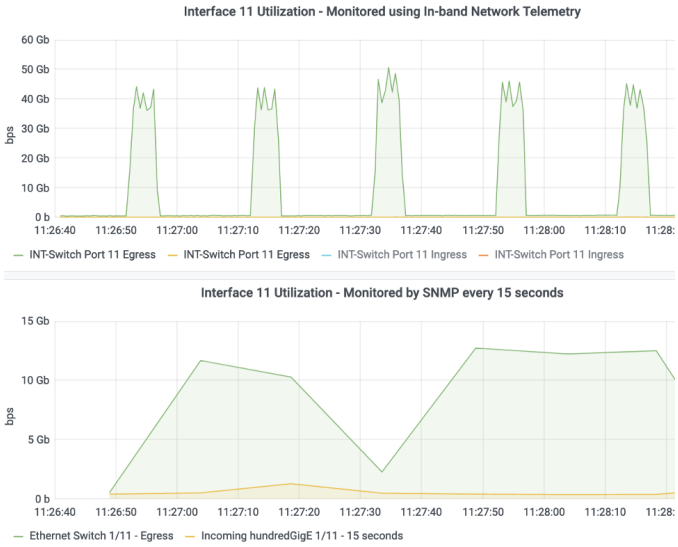

Fig. 6. Detecting burts: INT vs. SNMP.

network monitoring framework, programmable ASICs, such as the Intel Tofino, enable network operators with deeper visibility into components previously considered too complex, or not possible to be monitored, for instance, the egress buffer occupancy [14]. The egress buffer is used by network devices to store packets received from ingress interfaces before they are then sent out of the egress interface. An egress buffer holds the outgoing packets in partitions called *egress queues* and each queue can have a user-defined length depending on the traffic engineering policies. Once an egress queue gets full or close to full utilization, packets are then dropped in a process called *tail drop* [15], impacting the data transfers' performance. Monitoring and defining the length of egress queues is an old but active investigation topic [16] [17], because monitoring the egress queues was not supported by fixed function switch ASICs until recently [14]. With the accuracy provided by INT, egress queue occupancy can be monitored per packet, providing instantaneous reports to support the network operation and traffic engineering policies.

By monitoring the egress queue occupancy, AmLight can measure the packet drops probability and the jitter introduced per node in the path. At AmLight, when a link is operating under 50% of its capacity, the hop delay is less than 1.5 microseconds as it can be seen in Figure 7. Figure 6 and Figure 7 share the same time window (x-axis) correlating the events. Since no concurrent flows were sharing the egress interface, i.e., the bandwidth utilization was below its maximum capacity, it is possible to see that hop delay was not impacted by the microbursts reported in Figure 6. However, if the INT node were forwarding a significant amount of traffic when the microburst happened, the hop delay would have increased accordingly.

Figure 8 provides a visualization of a high queue occupancy due to multiple data transfers sharing the same egress interface and the same egress interface's queue. Before the congestion,

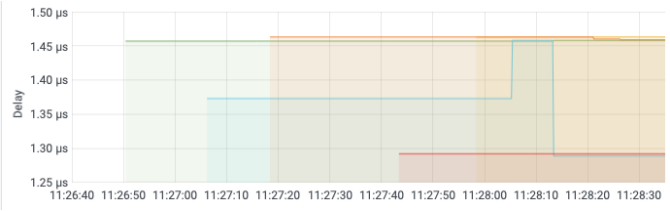

Fig. 7. Hop Delay for INT Node "Novi07" at AmLight.

the queue occupancy was reported as less than 10KB. During the congestion, queue occupancy was reported at 2MB.

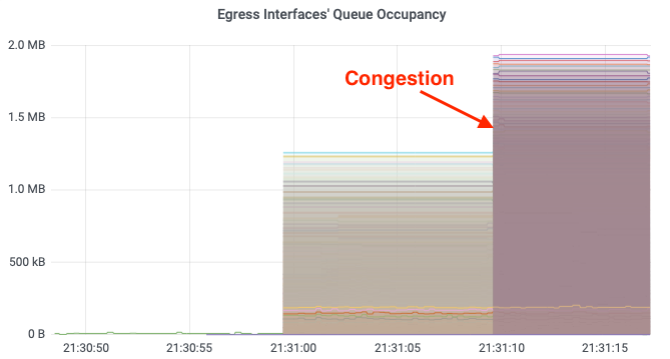

Fig. 8. Queue Occupancy during a congestion.

Figure 9 provides a visualization of the hop delay during the same congestion period: since there is more buffering, the packets spend more time waiting to be serialized out of the egress interface. During the congestion, hop delay increased from less than 2 microseconds to more than 60 milliseconds. This kind of hop delay variation (a.k.a. jitter) creates impact to real-time applications such as Voice over IP (VoIP) and videoconferencing (for instance, Zoom calls) and TCP-based data transfers.

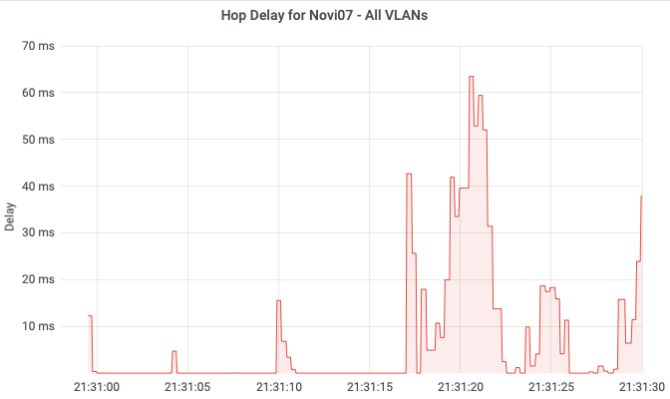

Fig. 9. Hop Delay for INT Node "Novi07" during a congestion.

C. Tracing packets

Certain conditions, such as regulatory obligations or monitoring and troubleshooting scenarios, require network operators to verify that all packets expected to follow a predetermined path in the network are indeed being forwarded across an expected set of nodes, links, and interfaces (including, optionally, the interface's queue). However, depending on the number of redundant paths in the network, fault tolerance strategy, and load balancing or link aggregation approach, it may be unfeasible to confirm that a packet traversed a particular route at a given time. This is because, without INT, data plane forwarding decisions are not reported in a per packet basis to avoid performance degradation.

By leveraging INT, AmLight developed an application to analyze the telemetry reports and provide information on the per-packet path. The application is called *proof-of-transit*, and it was named after the IETF Network Working Group effort to standardize mechanisms to securely prove that traffic transited a pre-defined path [18]. Proof-of-transit works by processing the INT reports and recording the path taken by a particular traffic flow whenever the path changes from previous measurements. The traffic flow is characterized by the endpoint's ingress node, ingress interface, and ingress VLAN id. The path taken is obtained by extracting information from the INT reports such as: INT node identifier, ingress and egress interface numbers, ingress/egress VLAN id, and, optionally, egress queue. All those fields combined represent a hop, and a set of hops represent the path taken. Whenever the path taken by a traffic flow changes, the information is recorded into the proof-of-transit database.

The proof-of-transit database provides historical information for all traffic flows transited into the network. Network operators can benefit from such historical information to troubleshoot network issues, produce evidence for regulatory obligations or service level agreements, or even evaluate network policies. More specifically, using the egress queue as part of the proof-of-transit record enables auditing Quality of Service network policies. For example, QoS policies can be validated by verifying if a particular class of service (defined by the traffic flow) took the expected path and the expected interface's queue along the route. Another practical use case is correlating all traffic flows sharing a particular path element in a specific time window, helping the network operator assess bottlenecks, identifying users impacted by a problem, or even narrowing down the root cause of a packet loss.

V. Conclusion and Future Work

In-band Network Telemetry (INT) has been presented as a novel technology with the potential to transform network monitoring by providing network operators with per-packet network telemetry capabilities for deeper visibility into the network. An INT environment was presented with elements, such as INT telemetry data, INT Collector, and telemetry reports, as well as their application to compute useful network monitoring metrics, such as hop delay, packet drop probability and buffer utilization, packet tracing, bandwidth utilization, and jitter. The benefits to the AmLight research and education network were presented, showcasing a new kind of network visibility that will change how traffic engineering and network monitoring will be performed in the future, and how academic communities and their applications will be better supported.

For future work, AmLight will expand the INT Collector to generate high-level network telemetry reports that will feed the AmLight network orchestrator in real-time. The end goal is to leverage high-level network telemetry reports to create a sub-second closed-loop network orchestration and operation that decreases the probability of packet drops caused by under-provisioned buffers. In addition, moving AmLight to a closed-loop network orchestration will change how the network operations team consumes network monitoring and telemetry data, how events exported by multiple sources are correlated, and how to write granular network policies to be consumed by the network to operate itself.

Finally, INT reports, when stored, become excellent datasets for network modeling and capacity planning. Since each telemetry report has the TCP/IP header and telemetry reports are created per packet, not per flow or sample-based, the use of high-performance Machine Learning and other Artificial Intelligence strategies can be employed to detect malicious traffic, such as DDoS attacks, single-packet attacks and scans usually undetectable by sampling approaches, as well as identifying network utilization trends to support capacity planning activities.

VI. Acknowledgments

The AmLight-ExP and AmLight-INT projects are supported by the National Science Foundation (NSF Awards #OAC-1848746 and #OAC-1451018). This research was possible thanks to the collaboration with NoviFlow. NoviFlow is a Network Operating System (NOS) provider for programmable white-box switches and Controller software designed to optimize and scale cybersecurity and SDN solutions.

References

[1] J. Ibarra, J. Bezerra, H. Morgan, L. F. Lopez, D. A. Cox, M. Stanton, I. Machado, and E. Grizendi, "Benefits brought by the use of openflow/sdn on the amlight intercontinental research and education network," in *2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*. IEEE, 2015, pp. 942–947.

[2] D. Harrington, R. Presuhn, and B. Wijnen, "Rfc3411: An architecture for describing simple network management protocol (snmp) management frameworks," 2002.

[3] M. Wang, B. Li, and Z. Li, "sflow: Towards resource-efficient and agile service federation in service overlay networks," in *24th International Conference on Distributed Computing Systems, 2004. Proceedings.* IEEE, 2004, pp. 628–635.

[4] T. Group *et al.*, "In-band Network Telemetry Dataplane Specification," https://github.com/p4lang/p4-applications/blob/master/docs/INT_v2_1.pdf, 2020, [Online; accessed 10-August-2021].

[5] Intel, "Intel® Tofino™ Series Programmable Ethernet Switch ASIC," https://www.intel.com/content/www/us/en/products/network-io/programmable-ethernet-switch/tofino-series.html, 2021, [Online; accessed 10-August-2021].

[6] D. Scholz, D. Raumer, P. Emmerich, A. Kurtz, K. Lesiak, and G. Carle, "Performance implications of packet filtering with linux ebpf," in *2018 30th International Teletraffic Congress (ITC 30)*, vol. 1. IEEE, 2018, pp. 209–217.

[7] T. Høiland-Jørgensen, J. D. Brouer, D. Borkmann, J. Fastabend, T. Herbert, D. Ahern, and D. Miller, "The express data path: Fast programmable packet processing in the operating system kernel," in *Proceedings of the 14th international conference on emerging networking experiments and technologies*, 2018, pp. 54–66.

[8] S. N. Z. Naqvi, S. Yfantidou, and E. Zimányi, "Time series databases and influxdb," *Studienarbeit, Université Libre de Bruxelles*, vol. 12, 2017.

[9] G. Labs, "Grafana: The open observability platform," https://grafana.com, 2021, [Online; accessed 10-August-2021].

[10] A. Hanemann, J. W. Boote, E. L. Boyd, J. Durand, L. Kudarimoti, R. Łapacz, D. M. Swany, S. Trocha, and J. Zurawski, "Perfsonar: A service oriented architecture for multi-domain network monitoring," in *International conference on service-oriented computing*. Springer, 2005, pp. 241–254.

[11] S. Ibanez, G. Antichi, G. Brebner, and N. McKeown, "Event-driven packet processing," in *Proceedings of the 18th ACM Workshop on Hot Topics in Networks*, 2019, pp. 133–140.

[12] X. Chen, S. L. Feibish, Y. Koral, J. Rexford, and O. Rottenstreich, "Catching the microburst culprits with snappy," in *Proceedings of the Afternoon Workshop on Self-Driving Networks*, 2018, pp. 22–28.

[13] R. Joshi, T. Qu, M. C. Chan, B. Leong, and B. T. Loo, "Burstradar: Practical real-time microburst monitoring for datacenter networks," in *Proceedings of the 9th Asia-Pacific Workshop on Systems*, 2018, pp. 1–8.

[14] S. Arslan and N. McKeown, "Switches know the exact amount of congestion," in *Proceedings of the 2019 Workshop on Buffer Sizing*, 2019, pp. 1–6.

[15] S. Patel, P. Gupta, and G. Singh, "Performance measure of drop tail and red algorithm," in *2010 2nd International Conference on Electronic Computer Technology*. IEEE, 2010, pp. 35–38.

[16] K. Avrachenkov, U. Ayesta, and A. Piunovskiy, "Optimal choice of the buffer size in the internet routers," in *Proceedings of the 44th IEEE Conference on Decision and Control*. IEEE, 2005, pp. 1143–1148.

[17] R. Stanojevic, R. N. Shorten, and C. M. Kellett, "Adaptive tuning of drop-tail buffers for reducing queueing delays," *IEEE Communications Letters*, vol. 10, no. 7, pp. 570–572, 2006.

[18] F. Brockners, S. Bhandari, S. Dara, C. Pignataro, J. Leddy, S. Youell, D. Mozes, and T. Mizrahi, "Proof of transit," Working Draft, IETF Secretariat, Internet-Draft draft-brockners-proof-of-transit-05, May 2018, https://www.ietf.org/archive/id/draft-brockners-proof-of-transit-05.txt. [Online]. Available: https://www.ietf.org/archive/id/draft-brockners-proof-of-transit-05.txt