

Enabling Tenant-driven Network Debugging with Partial Packet Provenance

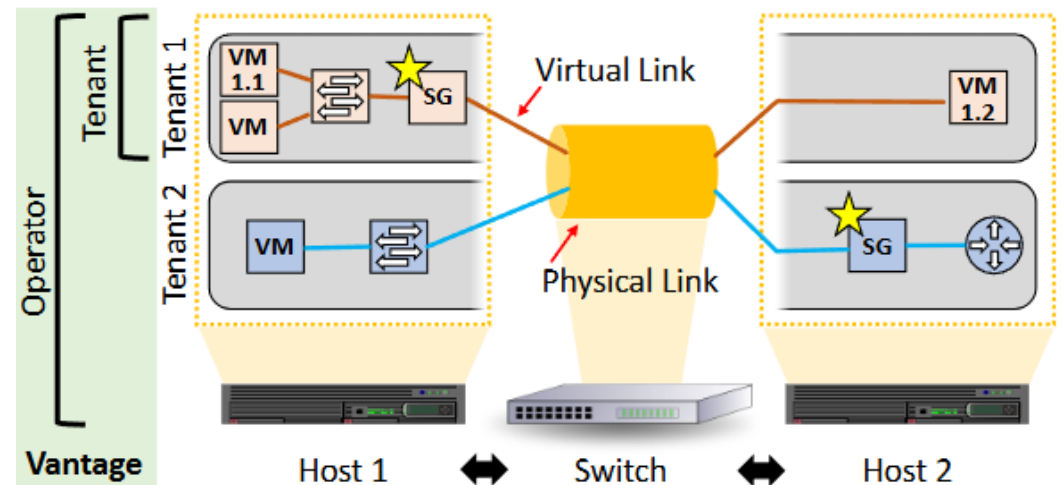
Alexander Wolosewicz (IL Tech), Raffay Atiq (SRI), Nishanth Shyamkumar (IL Tech),
Vinod Yegneswaran (SRI), Ashish Gehani (SRI), Nik Sultana (IL Tech)

Short Bio

- 2nd year Ph.D. student.
- Bachelor's in CS.
- Working on this project since ~September 2024.

Introduction

- Virtualization is present in many modern networks: Cloud, campus, telecom, testbed.
- Problems can manifest both in the virtual (**tenant**) network and in the physical network (**underlay**).
- When an issue occurs, *vantage points* enable diagnosing and debugging.
 - Network operators have many vantage points.
 - Tenants have significantly less.



Examples of Network Problems

- Packet loss and reachability.
- Performance anomalies – staying within SLOs.
- Correct forwarding, loop detection, path conformance.
- Causes include hardware/software faults, misconfigurations.
 - ByteDance: 35.4% of outages misconfigurations (Liu et al. HotNets 24)

Research Problem

- Tenants encounter issues in their virtual networks.
 - May be self-induced, may be virtual bug, may be underlay.
- Limited vantage point is an impediment.
- When in doubt, escalate to operator.
 - Operator support is a scarce resource.
- **Motivation:** improving tenant visibility, conserve operator time.
- **Research Question:** How can tenants diagnose/debug without operator support, preserving virtualization and permissions?

Existing Work

- In the literature: operator-level, non-generic, or non-comprehensive.
 - Netsight/ndb: comprehensive (packet histories), non-generic.
 - VND/NetHint: tailored to virtual environments, but underlay-embedded (VND, needs operator work) or non-comprehensive (NetHint, congestion-only).
- In the field: AWS, Oracle Cloud have static configuration checkers and flow-level logging.
 - Static checks are non-comprehensive.
 - Flow-level logging nice but potentially tedious or coverage-lacking.

Debugging Requirements

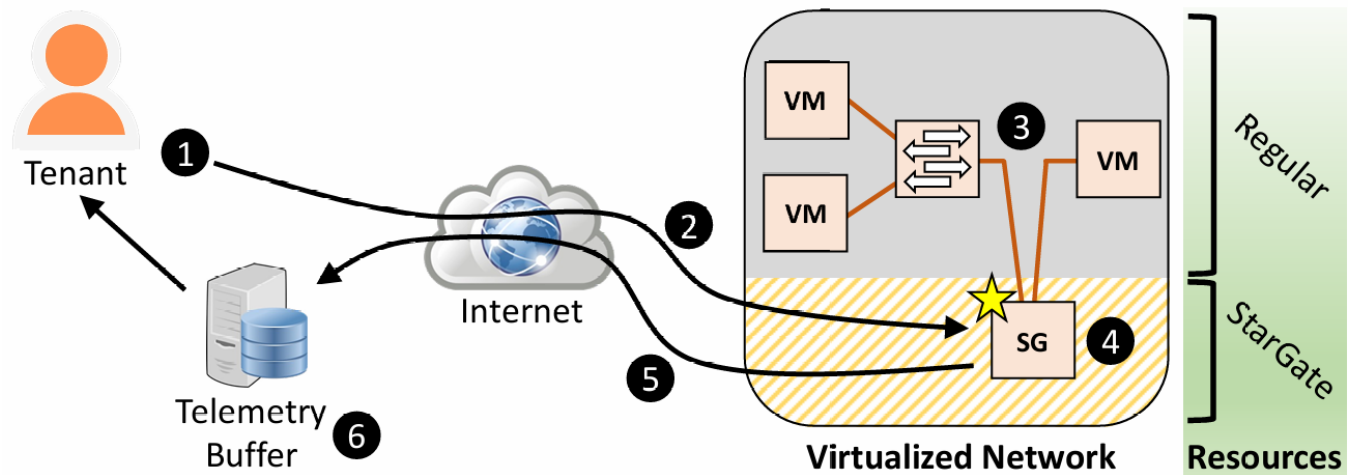
- **Secure and private:** a tenant's debugger is contained to the tenant.
- **Modification-free:** a debugger functions on production traffic without modification to workloads, topology, VM images, software.
- **Scalable:** a debugger scales to large networks and concurrent tenant usage.
- **Flexible:** a debugger covers a full range of network problems.
- **Measurement-capable:** can report on the performance (and culpability) of the underlay.

Stargate

- A debugging architecture for tenants which is flexible and general.
 - No reliance on specific hardware or protocols.
 - Provides comprehensive network debugging.
- Formal design for reasoning about network faults with per-packet provenance.
- Working, user-evaluated implementation run on FABRIC.
 - Virtual network testbed we do not operate.

Design of Stargate

1. Tenants access their resources over the Internet.
2. Tenants use an API to create, manage, query SG.
3. SG nodes are placed to bisect virtual links.
4. SG nodes are custom lightweight VMs that capture information needed for debugging.
5. Debugging information is streamed to a telemetry buffer.
6. The buffer is queried by the tenant.



Formalism

- Partial Packet Provenance (3P): histories of packets.

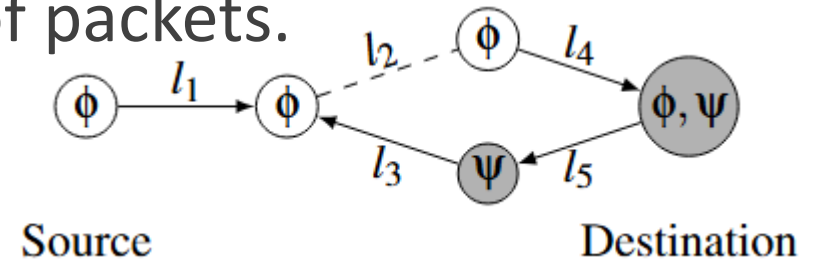
- Partial – does not require full visibility.

- Relation P : frames (ϕ) over links (l).

- Reach: where a ϕ is present.

- Repeat: same ϕ and l , different time.

- Duplicate: ϕ leaves more times on egress links L' than it entered on l .



$$\text{Reach}(\phi, l) \stackrel{\text{def}}{=} \{l : (_, _, l) \in \text{hist}(\phi)\}$$

$$\text{Repeat}(\phi, l) \stackrel{\text{def}}{=} |\text{Reach}(\phi, l)| > 0$$

$$\text{Duplicate}(\phi, l, L') \stackrel{\text{def}}{=} |\text{Reach}(\phi, l)| < \sum_{l' \in L'} |\text{Reach}(\phi, l')|$$

Debugging with 3P Primitives

- Reach covers drops (negation) as well as intended and unintended reachability; packet goes where it should, and not where it shouldn't.
- Reach can reveal loops (all instances where duplication not present).
- Duplication strengthens the two to handle the edge cases frame duplication can introduce.

Implementation

- **Virtual network:** the FABRIC national networking testbed.
- **API:** Python 3 extension of FABRIC's API, FABlib.
- **SG nodes:** Ubuntu 22 VMs running a C application using DPDK.
- Buffer uses SPADE as the provenance recorder and database manager (PostgreSQL backend).

SG Nodes

- Receives frame, checks if it has been tagged, tags if not.
 - In-band filtering determines which packets get this treatment – those not meeting the filter are just forwarded.
- Forwards frame (same order as received – goal is transparency).
- Sends frame ID, node info, time, and frame metadata to buffer.

Frame Trailers

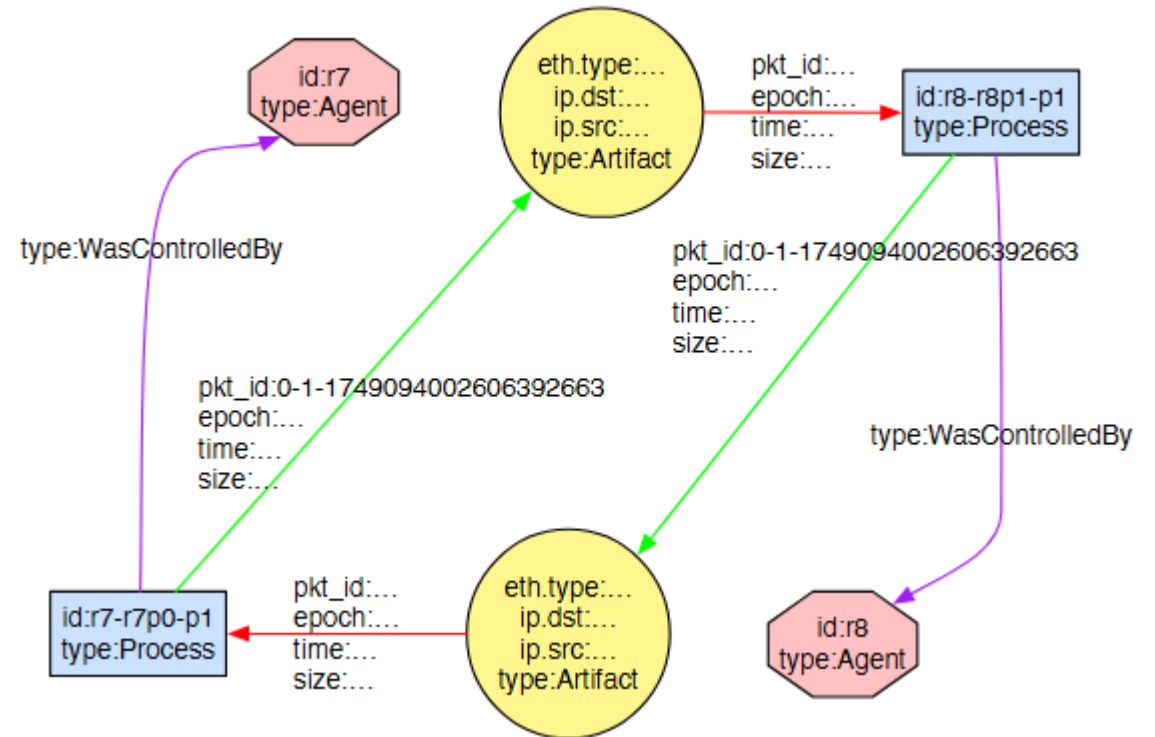
- Per the design, need to uniquely identify frames while maintaining the modification-free property.
- Frame trailers are invisible to process not actively looking for them – no length increment, so appear as padding.

0-1	2-3	4-5	6-7	8-9	10-11	12-13	14-15
Timestamp (ns)				Monitor	Port	Copy 6-7	0x6587

- Limitation: must avoid fragmentation.

Provenance

- Conforms to Open Provenance Model, adapted to networking.
- With frame IDs, shows where it went, when, and what it was.
- Sufficient information to debug a complete set of problems.



Evaluation

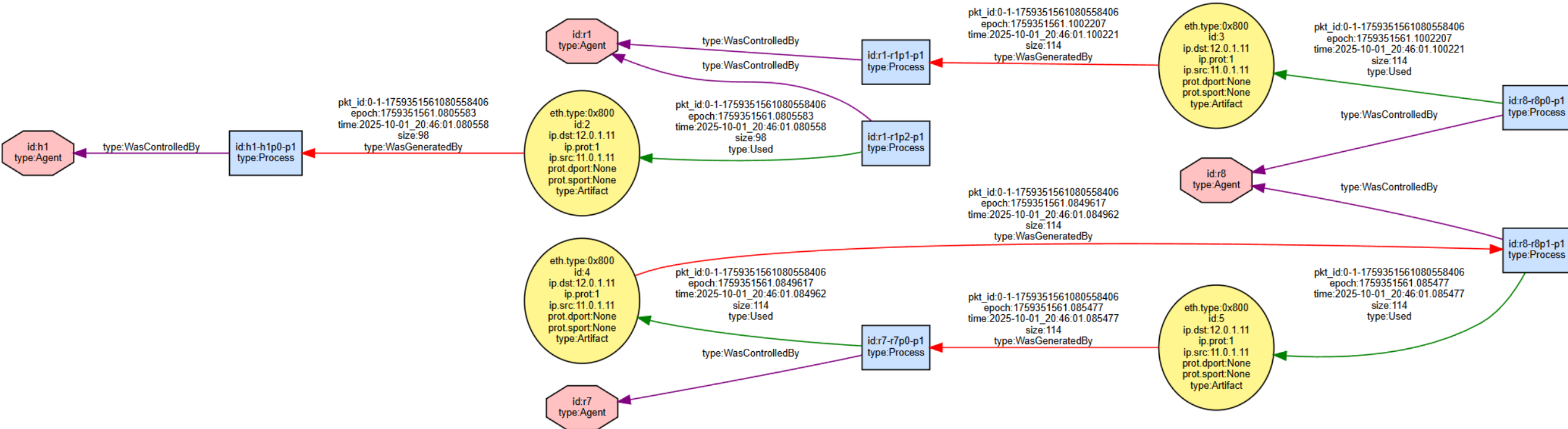
- SG nodes use DPDK version 25.07 (latest at the time), 4 cores (2.4 GHz), 1 GB RAM, Mellanox Connect-X 6 SR-IOV virtual function NICs.
- Buffer: 4 cores, 16 GB RAM (for DB queries).
- FABlib version 1.9.2.

Debugging Problems

- Timestamped packet provenance sufficient to debug latency, reachability, and misconfiguration issues.
- Two example cases: misconfigured virtual switch forwarding and program bug causing header corruption.

Bad forwarding

- r7 receives a packet, reflects back to r8 where it drops.



Tunnelling Error

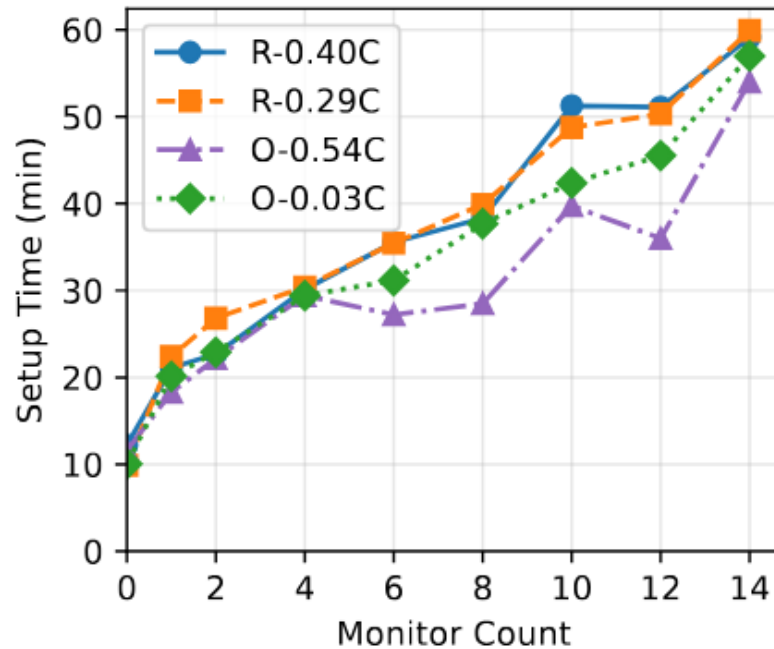
- P4 bug means no header emitted, corrupting downstream parsing.

TX Host	TX Interface	RX Host	RX Interface	Size	Epoch	Time	Ether Type	IP Protocol	Source IP	Destination IP	Source Port	Destination Port
r1	r1-r1p2-p1	h1	h1-h1p0-p1	90	1760428513.0113745	2025-10-14_07:55:13.011374	0x800	0	0.0.13.123	0.34.0.1	None	None
r3	r3-r3p1-p1	r2	r2-r2p0-p1	126	1760428513.0118923	2025-10-14_07:55:13.011892	0x800	17	15.1.1.12	15.1.1.11	2152	2152
r4	r4-r4p1-p1	r3	r3-r3p0-p1	126	1760428513.019532	2025-10-14_07:55:13.019532	0x800	17	15.1.1.12	15.1.1.11	2152	2152
r2	r2-r2p1-p1	r1	r1-r1p0-p1	126	1760428513.0212333	2025-10-14_07:55:13.021233	0x800	17	15.1.1.12	15.1.1.11	2152	2152
r5	r5-r5p0-p1	r6	r6-r6p1-p1	134	1760428513.0228193	2025-10-14_07:55:13.022819	0x800	17	15.1.1.12	15.1.1.11	2152	2152
r9	r9-r9p1-p1	r4	r4-r4p2-p1	126	1760428513.0263708	2025-10-14_07:55:13.026371	0x800	17	15.1.1.12	15.1.1.11	2152	2152
r6	r6-r6p2-p1	r9	r9-r9p2-p1	134	1760428513.027536	2025-10-14_07:55:13.027536	0x800	17	15.1.1.12	15.1.1.11	2152	2152
h2	h2-h2p0-p1	r5	r5-r5p2-p1	98	1760428513.0340798	2025-10-14_07:55:13.034080	0x800	1	12.0.1.11	11.0.1.11	None	None

User Evaluation

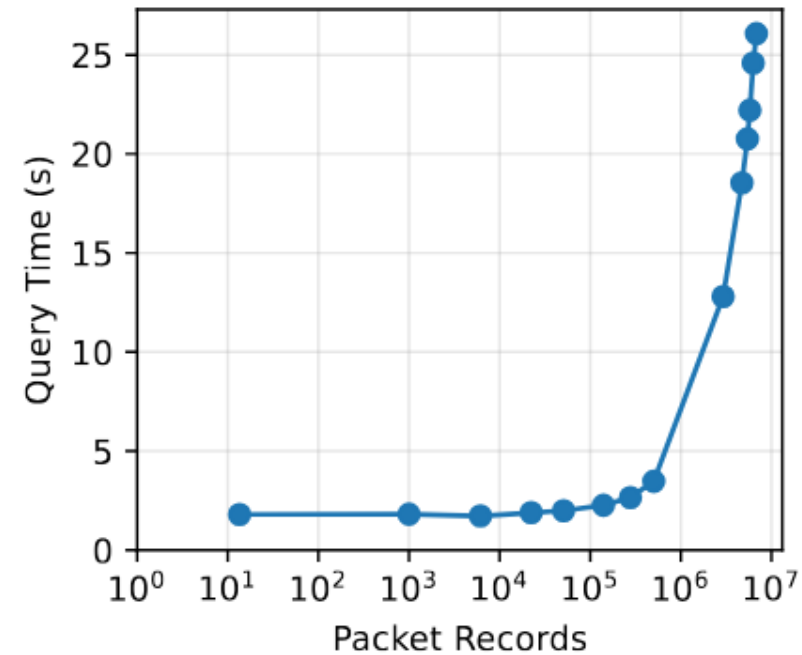
- 11 outside users used SG to debug some networking issues while 11 used common tools like ping and tcpdump.
- Those with medium FABRIC and network debugging experience could debug up to 37.5% faster on the second exercise, with only a 15-minute demo video and a previous exercise (two of three solved with a single API call).
- Results are promising. We'd like to run a larger study.

Scalability

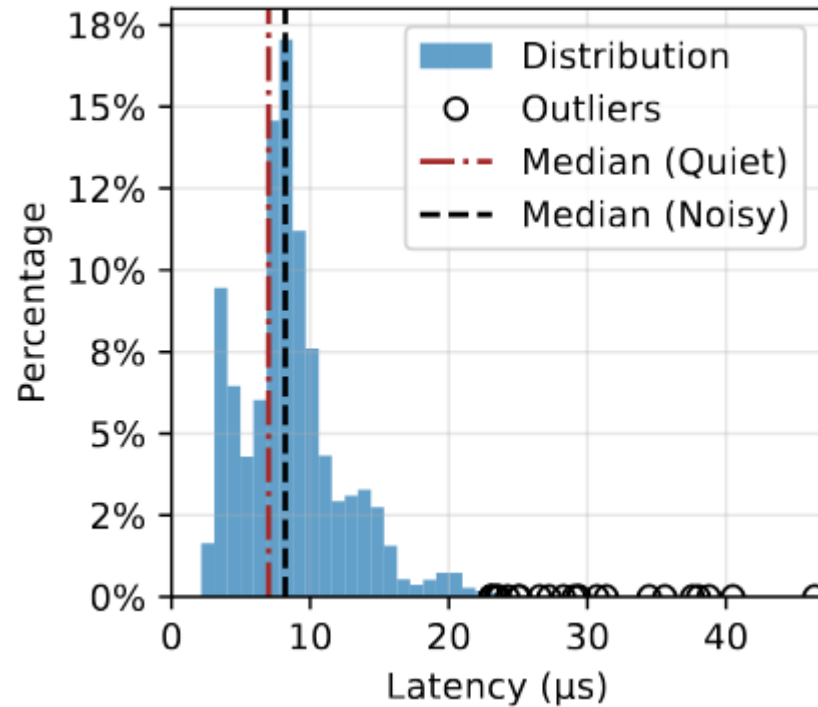


Adding SG nodes to 11-node, 14-link slice on 4 sites
R: on terabit ring vs Off-ring (size of site), C: ratio
avail. CPU (use of site)

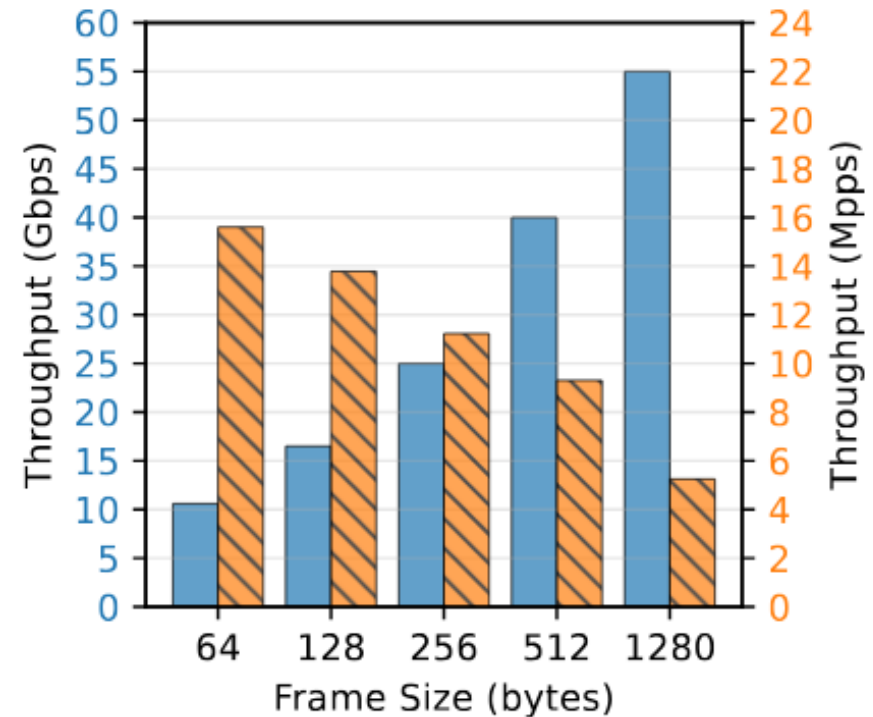
Querying provenance database of 5-hop packet histories for one specific history (5 records/history)



In-Band Performance



Left: Latency distribution in quiet environment (6,692 samples).



Right: SG node throughput at varying frame sizes (AWS: 25 Gbps limit, 5-10 common).

ILLINOIS TECH

College of Computing

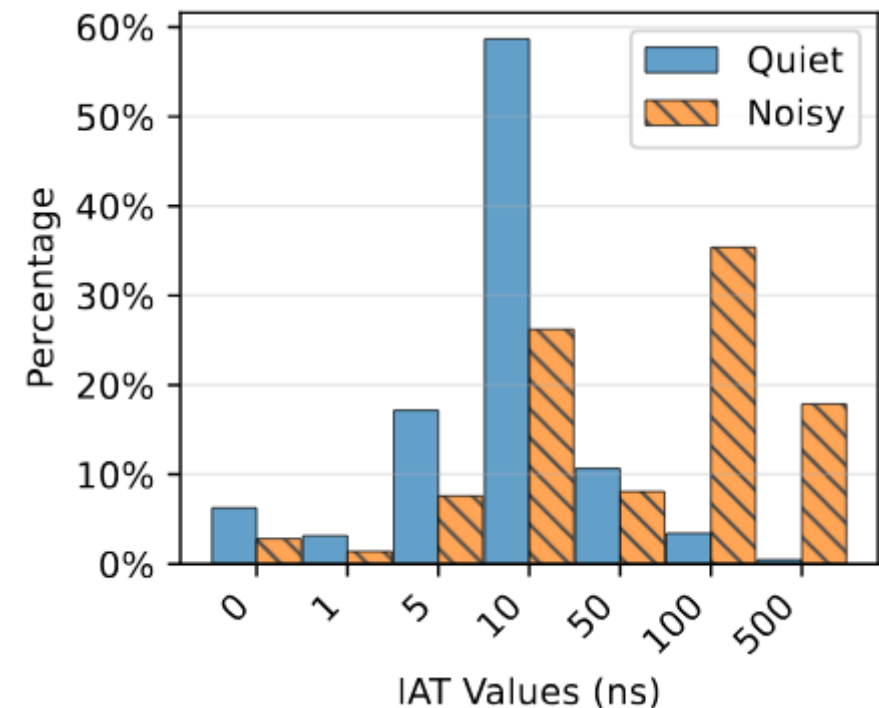
Trailer Preservation

- Preserved in many contexts.
- Linux/BSD IP forwarding rely explicitly on the IP length field.
- Linux 5.15.0-143-generic.
- BSD 14.0-RELEASE-p6.
- OpenVSwitch 2.17.9.

Element	Preserved	Element	Preserved
Linux Bridge	●	FABRIC L2	●
Linux IP Fwd	○	FABRIC L3	●
BSD Bridge	●	FABRIC Fac. Port	●
BSD IP Fwd	○	OpenVSwitch	●
Alveo-to-NIC	●	Tofino2	●

Insight into Underlay

- Added traffic replaying to SG nodes.
- Replay the same traffic multiple times to build a baseline of network consistency.
- Experiment reveals impacts from high co-tenant traffic (overall less than link capacity).



Conclusions

- Stargate provides a highly-general, highly-capable debugging tool for virtual networks.
- Readily available, and successfully used outside our team.
- URL: <https://crease.cs.iit.edu/knit11/artifact>
- Tutorial materials and video: <https://crease.cs.iit.edu/>

Future Work

- **Automation:** SG node placement, deciding traffic filtering criteria, refining queries from more general prompts.
- **Scalability:** boosting monitor throughput through load-balancing. Requires work in coordinating the monitors to ensure overall network consistency.
- **Network consistency:** expanding and refining metrics to quantify the consistency measurement of the network (visibility into underlay).