USTC related

ONGOING COLLABORATION

Wei Yue
Cloud Lab, Futurewei Technologies
11/2021

Around xdp/eBPF based service framework R&D ideas

- Application centric networking
    deep dive into useful scenario, e.g. AI apps, streaming apps, to explore benefits of different approaches;
- Smarter Observability
    e.g. live detect whether noted services goes through unwanted path, or snapshots for specific service;
- Anomaly detection and reaction: define anomaly signature and follow up ebpf function(tail-call) for reaction, collect useful scenario
- Deep dive into eBPF verification engine(more focused scope with clearer goals)
    1. https://dl.acm.org/doi/pdf/10.1145/3452296.3472937 (Aquila - verification system for programmable data plane)
    2. https://dl.acm.org/doi/pdf/10.1145/3452296.3472929 (K2 - Synthesizing Safe and Efficient Kernel Extensions for Packet Processing)
    3. https://linuxplumbersconf.org/event/11/contributions/944/attachments/893/1707/2021-09-23-lpc21.pdf (A proof-carrying approach to building correct and flexible in-kernel verifiers)
- Continuing XDP/eBPF offloading technics, DPU(Nvidia), define use cases.
- Faster OVS datapath with XDP
    - https://legacy.netdevconf.info/0x14/pub/papers/41/0x14-paper41-talk-paper.pdf
    - https://dl.acm.org/doi/pdf/10.1145/3452296.3472914
- DSL/Yang/P4 as intent, converts and deployed to dataplane via xdp/eBPF, etc. some e2e use case poc
- eBPF collecting/signing and verification mechanism

Other references
https://sebymiano.github.io/publication/2021-polycube/2021-polycube.pdf
https://webthesis.biblio.polito.it/15302/1/tesi.pdf

NAI READING LIST (10/21/2021)

Joint Network and Application Control

- *Towards an API for the Path-Aware Internet*
- *Tighter Application-Network Interfacing to Drive Innovation in Networked Systems*
- *Leveraging In-network Application Awareness*
- *Towards a Software-Defined, Fine-Grained QoS Framework for 5G and Beyond Networks*

QoE and QoS Interaction

- *Characterizing the Relationship Between Application QoE and Network QoS for Real-Time Services*
- *IntQOE: Integrated End-to-end QoE Optimization for Edge Computing Enabled Web Application*

OBSERVABILITY (10/21/2021)

- Open Telemetry

    https://opentelemetry.io/

    https://github.com/Flowmill/flowmill-collector

- In-band Telemetry

    https://github.com/intel/host-int

    https://github.com/intel/host-int/blob/main/docs/host-int-project.pptx

- Hubble

    https://github.com/cilium/hubble

- Falco

    https://github.com/falcosecurity/falco

eBPF in Microservices Observability(AWS)

Challenges in microservices

We don't just monitor VMs or processes.
We monitor critical paths.

Context matters.
Downstream stack don't have context.

We initially debug RPCs.
We debug functions or syscalls secondarily.

Too much data.
Need runtime controls to modify the collection.

Instrumentation is a two-year roadmap.
Data is not consistent.

Networking observability is core.
Out of the box instrumentation is essential.
Extensibility in runtime is critical.
Decoration and enrichment is needed.

eBPF help in Microservices Observability(AWS)

Network Diagnostics

TCP, UDP, HTTP, gRPC metrics
Inspect protocols (MySQL, Postgres, ...)

Distributed Traces

Automatically create request span if a trace header is present.

```
GET /users HTTP/1.1
Host: users.service
Accept-Encoding: gzip, deflate
Connection: Keep-Alive
Traceparent: 00-4bf92f3577b34da6a3ce929d0e0e4736-00f067aa0ba902b7-01
```
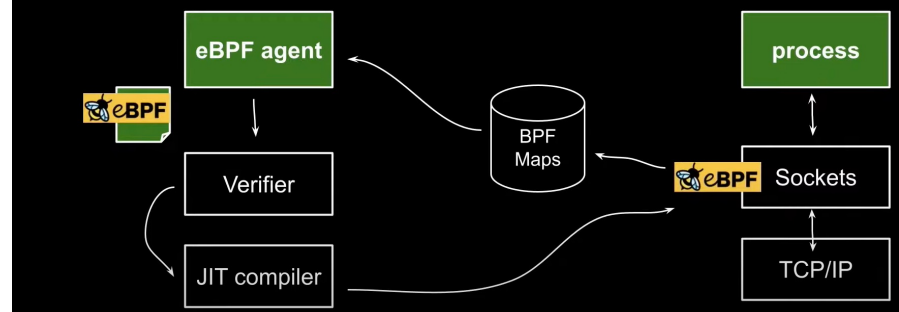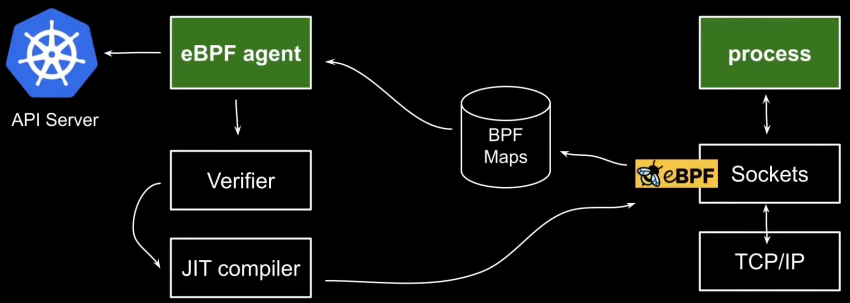
Fleet-wide Profiling

Runtime Extensibility

Decorating with Context

What's next?

- High level language to write probes.
- Make eBPF agents widely available.
- More platforms supporting eBPF.
- Reusable eBPF event processing.

Cloud Transformation

5G pushing forward a cloud-native approach and growing erosion towards hyperscaler clouds.

New Telco and OTT services mostly developed over cloud platforms

Massive Scale

Decentralization and Edge Computing

Evolution towards fine grained application policies, Low latency and localized apps focused.

However ?!!

Commoditization
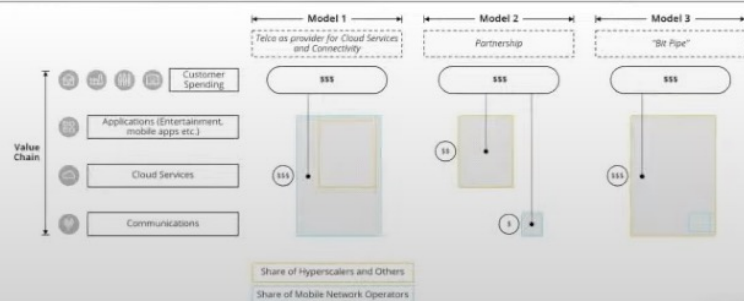
Internet access becoming an "essential service" with decreasing revenues. Value add services moving to OTT space.

Increase Pressure on QoE

Fastest ISPs 2020—Canada (Major ISPs)

Telco's being increasingly measured on Speed, Latency, MOS for various traffic patterns (gaming, cloud, content)

The need for a better handshake

3.2 Gaps & Challenges in Today's Communication Services

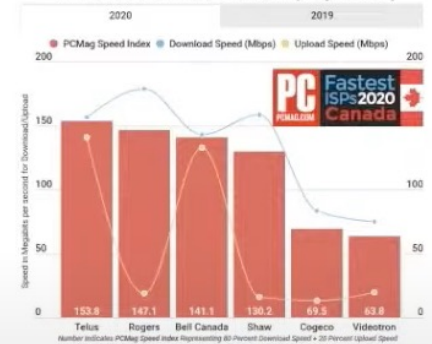The Network 2030 initiative is a structured approach to defining the capabilities of networks and corresponding communication services for the decade following 2030. The goal is to have networks ready for the market verticals that will utilize emerging technologies in 2030. Network 2030 extrapolates from what we know about technologies and develops a vision of the new media, new services, and new infrastructure. For this, we outline a few areas of importance to address towards building this vision.

Lacking service-network interaction: Failure to find an ordered and healthy relationship between the applications and the network has been a sticking point. Connectivity is just one of the workflows involved in application logic. It is a service consumed, with reachability being the only explicit means of setting up the application behaviour in the networks. However, a number of services offered by networks are not obvious to the end user. These include reliability of the network fabric, broadcast or multicast, integrity and security of data delivered, and network level awareness of congestion, capacity, and latency. Accommodating for such capabilities directly in the networks have been much of the focus of industry for the past thirty years. *Due to lack of direct support for such services through proper interfaces to the network, application developers have been left in a limbo, designing for every conceivable possibility of network failures and outages.* Many of these services are controlled over end-to-end interfaces between the endpoints using the transport (TCP) layer which is another example of free-form evolution aiming to solve the problems of the network without sufficient assistance from the networks.

Network 2030 - A Blueprint of Technology, Applications and Market Drivers Towards the Year 2030 and Beyond

Get the Network Out of the Way

Justin Pietsch Follow
Jan 21 · 15 min read

Getting the network out of the way has been very important for me in my thinking about networks, and is an easy way to help talk about a bunch of very important concepts, decisions, and arguments. This can sound trite, obvious, or insulting. In this post I'll try to describe the concepts I'm talking about and illustrate with some of most important examples in my career.

The more that the network is noticed the worse things are for everyone. Often times, especially when the network is noticed, networking and network engineers are thought of negatively. Instead, if you think of it as a challenge it can help you focus on making a great network. You can think about your goals: how important it is to keep the network working well, to not disrupt the business, and to be able to keep up with any changes that the business needs.

The IETF answer ... APN and PANRG

It should be like this

APN Edge
APP
eBPF

Controller

APN Domain

| Source | APN-Edge | APN-Head | APN-Midpoint | APN-Endpoint | APN-Edge | Destination |

Policy enforcement point x
Policy enforcement point y
Policy enforcement point z

Normal packet (any payload protocol)
Determine APN Attribute

Encapsulate Packet for Tunnel (any tunnelling protocol)

Apply Policy Dependent on APN Attribute applies to the policy enforcement points such as APN-Head, APN-Midpoint, APN-Endpoint within the APN Domain

Decapsulate Packet

Normal packet

An APN Domain may span multiple network domains controlled by the same operator

https://datatracker.ietf.org/wg/apn/about/
https://datatracker.ietf.org/rg/panrg/about/

What other disruption has shown us


```
table srv6_local_sid {
    reads {
        ipv6.dstAddr : lpm;
        ipv6_srh.valid : ternary;
        ipv6_srh.segLeft : ternary;
        ipv6_srh.nextHdr : ternary;
    }
    actions {
        drop_;
        transit;                /* T, T.INSERT, T.ENCAPS
*/
        end;                    /* END */
        end_x;                  /* END.X */
        end_t;                  /* END.T */
        end_dx2;                /* END.DX2 */
        end_dx4;                /* END.DX4 */
        end_dx6;                /* END.DX6 */
        end_dt4;                /* END.DT4 */
        end_dt6;                /* END.DT6 */
        end_b6;                 /* END.B6 */
        end_b6_encaps;          /* END.B6.ENCAPS */
    }
}
```
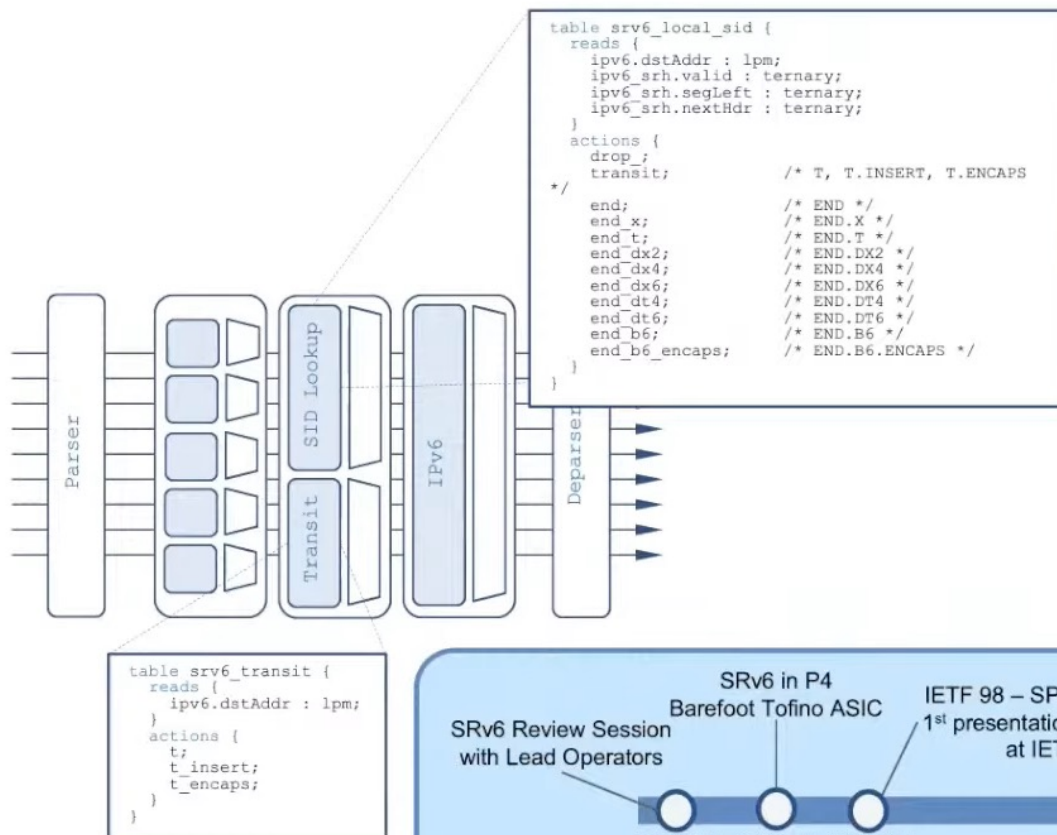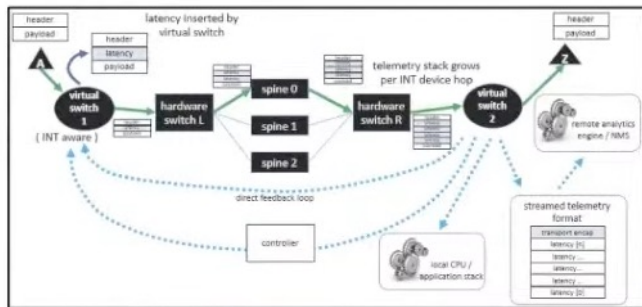
o "Democratization" of ASIC programming allowing for personalization of data plane.

o Benefits of building a strong open community merging industry and academia.

o Rapid implementation and testing even if reference standard is not fully baked.

o Ideally same flexibility would apply for Linux data plane development

```
table srv6_transit {
    reads {
        ipv6.dstAddr : lpm;
    }
    actions {
        t;
        t_insert;
        t_encaps;
    }
}
```

SRv6 Review Session with Lead Operators — 01/2017

SRv6 in P4 Barefoot Tofino ASIC — 02/2017

IETF 98 – SPRING WG 1st presentation of SRv6 at IETF — 03/2017

rfc8754
rfc8986 — 03/2021
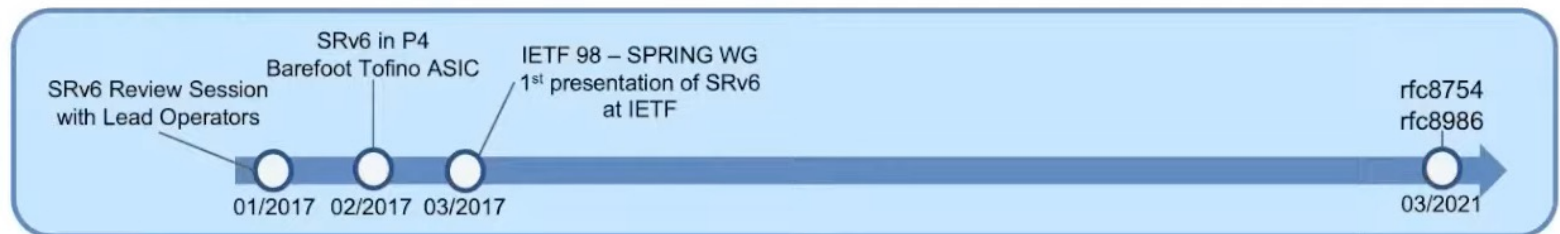
End-to-End In-band Telemetry

o Extend observability across "inband" telemetry collection
o Leveraging eBPF to implement Host-Based INT (https://github.com/intel/host-int)

https://github.com/p4lang/p4-
applications/blob/master/telemetry/specs/INT.mdk