Hyperscale NICs: performance and features

Standardizing server NICs at OCP

Willem de Bruijn, willemb @ Google

NETWORKING

NIC features to reach hyperscale

- Stateless offloads
- Tunneling
- Telemetry
- Multi Queue Scaling
- Encryption (OCP Tech Talks '22)
- Traffic Engineering (Global Summit '21)
- Etc.

Targets

- Exclude (for this spec) smartNICs and virtualization
- Hyperscale & similar large deployments
  - High-end servers: 100+ {cores, Gbps, Mpps}
  - Linux

Ambiguity, mistakes, misunderstandings

- Ambiguity: telemetry
- Mistakes: UDP zero checksum
- Misunderstandings: tunnel offload

Why OCP standardization

- Share & codify knowledge about subtler points of the technology
- Common understanding between vendors and users (w/o NDAs)
- Standardize the core features to focus efforts on innovation
- Community developed conformance tests

Core Features

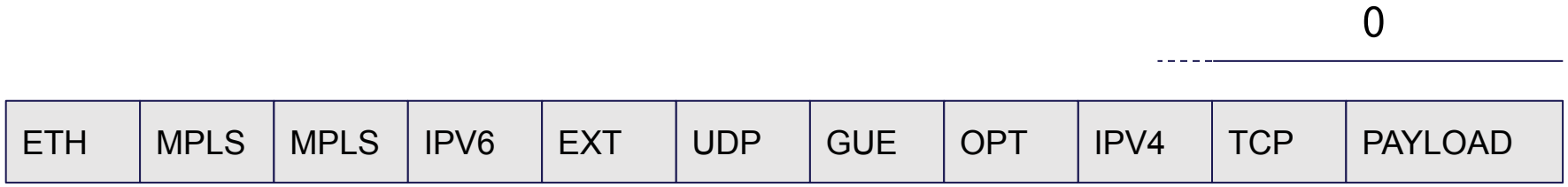
Checksum Offload

- SHOULD: Generic Checksum Offload

Core Features

Checksum Offload

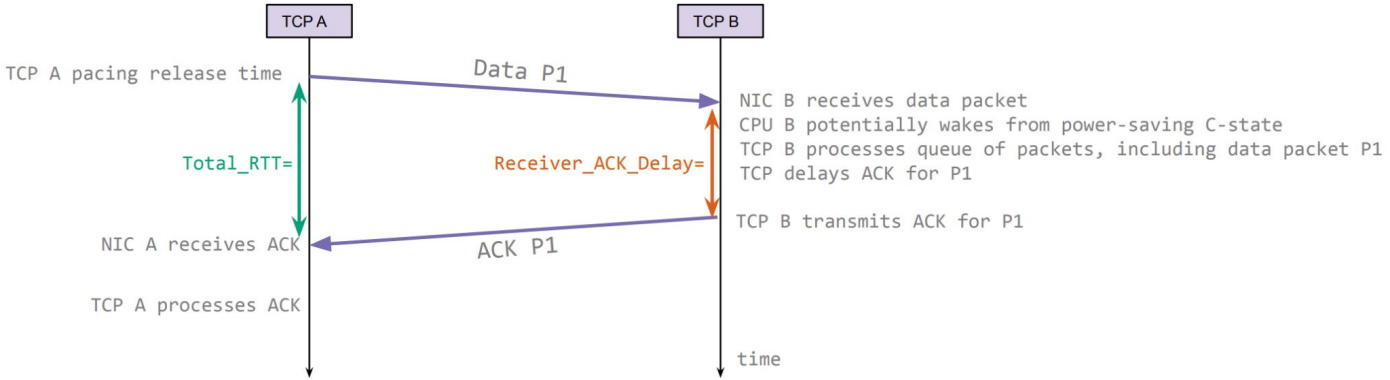
- SHOULD: Generic Checksum Offload



Local Checksum Offload

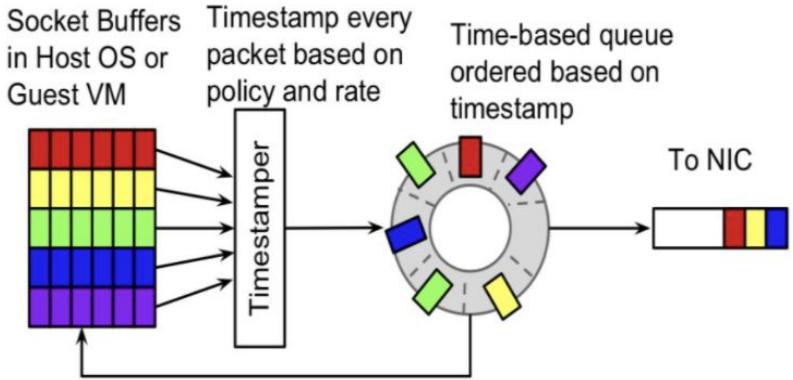
Time: Measurement

- Timestamping + Clock synchronization
- PTP at line rate
- Congestion Control: BBR.Swift @ IETF 109



Time: Delayed Transmission

- TCP pacing: Earliest Departure Time @ netdevconf 0x12
- Carousel @ SIGCOMM'17
- SO_TXTIME



Conformance Testsuite

Linux kernel sources at `tools/testing/selftests/net`

- `gro`
- `timestamping`
- `toeplitz`
- `txtimestamp`
- `so_txtime`
- `udpgso`

Missing

- configuration (ethtool)
- telemetry

Performance Testsuite

Per platform targets

- throughput: BPS, TPS, PPS
- latency (hot and cold system)

- github.com/google/neper
- tcp_stream
  - 1, 10, NR_CPUS, NR_CPUS * M flows
  - 1, 10, NR_CPUS, NR_CPUS * M threads
- tcp_rr_slow, tcp_rr_fast
- tcp_stream bi-directional
- ping_slow, ping_fast

Status and Progress

- Specs in development
  - Core NIC features
  - Inline Crypto Offload
- Possible future specs
  - Time and Timestamping
  - Traffic Engineering
- Champion others

Call to Action

- Have your voice heard
  - Sign effort CLA + help develop existing efforts
  - Suggest new efforts
  - Open up pre-existing specs to the community
- Join the Monthly OCP Networking call, every 1st Monday at 10 AM PT
- Join the OCP Networking mailing list
- More info at opencompute.org/wiki/Networking/NIC_Software
- Timeline
  - Revisions and future extensions