

# eBPF for NVMe

# What is NVMe

“The NVM Express<sup>®</sup> (NVMe<sup>®</sup>) family of specifications define how host software communicates with non-volatile memory across multiple transports like PCI Express<sup>®</sup> (PCIe<sup>®</sup>), RDMA, TCP and more.”

Or in simple words: NVMe it is an industry specification for talking to SSDs directly or over various networking protocols.

# Computational storage

- NVMe is working on a Computational Programs command set
  - Allows to download programs to storage devices to run close to the storage
  - Initially brought into NVMe by AWS and Intel including use of eBPF for programs
- Due to lack of NVMe standard the programs have been deferred and only fixed functions (e.g. compression, encryption) are supported for now

# Adaptive Multipathing

- There can be multiple connections to a storage device
  - (e.g. multiple NICs with different network connections)
- With some backends like distributed storage, the client can do intelligent decisions which one to send commands to
  - Basically application level source routing
- The algorithm is very specific to the implementation of the storage “device”
  - Might require the storage device sending a small bit of code that gets executed on the client
  - **→ very interesting security implications that require strong verification of the code!**

# Pointer chasing optimizations

- eBPF could be used to chase pointers in btrees or similar data structures in the kernel and eventually the storage device.
  - So far only an academic paper:  
<https://dl.acm.org/doi/pdf/10.1145/3458336.3465290>
  - Some interest from hyperscalers in this approach.