IX: A Protected Dataplane Operating System for High Throughput and Low Latency

Adam Belay, George Prekas,
Samuel Grossman, Ana Klimovic,
Christos Kozyrakis, Edouard Bugnion
HW is fast

64-byte TCP Echo:

- **HW Limit**
- **Linux**
- **IX**
HW is fast, but SW is a Bottleneck

64-byte TCP Echo:

- HW Limit
- Linux
- IX

4.8x Gap

8.8x Gap
IX Closes the SW Performance Gap

64-byte TCP Echo:

- HW Limit
- Linux
- IX

[Graph showing Microseconds and Requests per Second]
Two Contributions

#1: Protection and direct HW access through virtualization

#2: Execution model for low latency and high throughput
Why is SW Slow?

Complex Interface

Code Paths Convoluted by Interrupts and Scheduling

Created by: Arnout Vandecappelle
http://www.linuxfoundation.org/collaborate/workgroups/networking/kernel_flow
Problem: 1980s Software Architecture

• Berkeley sockets, designed for CPU time sharing
• Today’s large-scale datacenter workloads:

  **Hardware: Dense Multicore + 10 GbE (soon 40)**
  - API scalability critical!
  - Gap between compute and RAM -> Cache behavior matters
  - Packet inter-arrival times of 50 ns

  **Scale out access patterns**
  - Fan-in -> Large connection counts, high request rates
  - Fan-out -> Tail latency matters!
Conventional Wisdom

• Bypass the kernel
  – Move TCP to user-space (Onload, mTCP, Sandstorm)
  – Move TCP to hardware (TOE)

• Avoid the connection scalability bottleneck
  – Use datagrams instead of connections (DIY congestion management)
  – Use proxies at the expense of latency

• Replace classic Ethernet
  – Use a lossless fabric (Infiniband)
  – Offload memory access (rDMA)

• Common thread: Give up on systems software
Our Approach

- **Bypass the kernel**
  - Move TCP to user space (Onload, mTCP, Sandstorm)
  - Move TCP to hardware (TOE)

- **Avoid the connection scalability bottleneck**
  - Use datagrams instead of connections (DIY congestion control)
  - Use proxies at the expense of latency

- **Replace classic Ethernet**
  - Use a lossless fabric (Infiniband)
  - Offload memory access (rDMA)

- **Tackle the problem head on...**

Robust Protection Between App and Netstack
Connection Scalability
Commodity 10Gb Ethernet
Separation of Control and Data Plane

- CP
- DP
- DP
- Userspace
- Kernelspace
  - Host Kernel
  - C
  - C
  - C
  - C
  - C
Separation of Control and Data Plane

Userspace

Kernelspace

Host Kernel

CP

DP

RX
TX

RX
TX

RX
TX

RX
TX

C

C

C

C

C

C
Separation of Control and Data Plane

Ring 3
IX CP

Guest
Ring 0
IX DP

Host
Ring 0
Host Kernel
IX DP
RX TX
RX TX
RX TX
RX TX

C
C
C
C
C
C
Separation of Control and Data Plane

Ring 3
IX CP

Guest Ring 0
IX DP

Host Ring 0
Linux kernel

Dune

IX DP
IX DP

RX TX
RX TX
RX TX
RX TX

C C C C C
Separation of Control and Data Plane

Ring 3
IX CP

Guest Ring 0
IX
libIX
Memcached

Host Ring 0
IX
libIX
HTTPd
Linux kernel
Dune
RX
TX
RX
TX
RX
TX
RX
TX
C
C
C
C
C
C
The IX Execution Pipeline

- Event-driven app
- libIX
- Event Conditions
- Batched Syscalls
- Guest Ring 0
- RX FIFO
- TCP/IP
- RX
- TX
- TCP/IP
- Timer
- Ring 3
Design (1): Run to Completion

Event-Driven App

libIX

TCP/IP

Batched Syscalls

RX FIFO

RX

TX

Improves Data-Cache Locality
Removes Scheduling Unpredictably
Design (2): Adaptive Batching

- Event-Driven App
- libIX
- TCP/IP
- Batched Syscalls

Improves Instruction-Cache Locality and Prefetching
See the Paper for more Details

• Design (3): Flow consistent hashing
  – Synchronization & coherence free operation
• Design (4): Native zero-copy API
  – Flow control exposed to application
• Libix: Libevent-like event-based programming
• IX prototype implementation
  – Dune, DPDK, LWIP, ~40K SLOC of kernel code
Evaluation

- Comparison IX to Linux and mTCP [NSDI ’14]
- TCP microbenchmarks and Memcached

![Diagram of network setup with IX and 10GbE Switches]
TCP Netpipe

- Half Bandwidth @ 20 KB
- Half Bandwidth @ 135 KB

5.7 us 1/2 RTT

Goodput (Gbps)

Message Size (KB)

IX-IX
Linux-Linux
mTCP-mTCP
TCP Echo: Multicore Scalability for Short Connections

![Graph showing the scalability of different networking configurations. The x-axis represents the number of CPU cores, and the y-axis represents messages per second (x 10^6). The graph includes data for IX 10GbE, IX 4x10GbE, Linux 10GbE, Linux 4x10GbE, and mTCP 10GbE. There is a notable saturation point at 1x10GbE.]
Connection Scalability

Messages/sec (x 10^6) vs Connection Count (log scale)

- IX-40Gbps
- IX-10Gbps
- Linux-40Gbps
- Linux-10Gbps

~10,000 Connections Limited by L3
Memcached over TCP

Latency (µs)

- IX (p99)
- IX (avg)
- Linux (p99)
- Linux (avg)

USR: Throughput (RPS x 10³)

2x Less Tail Latency

3.6x More RPS

6x Less Tail Latency With IX clients

SLA
IX Conclusion

• A protected dataplane OS for datacenter applications with an event-driven model and demanding connection scalability requirements

• Efficient access to HW, without sacrificing security, through virtualization

• High throughput and low latency enabled by a dataplane execution model