CSE 291: Operating Systems in Datacenters

Amy Ousterhout

Oct. 5, 2023

Agenda for Today

- Reminders
- Introduction to network stacks
- IX discussion
- XDP discussion

Announcements and Reminders

- Sign up to lead a discussion
  - Spreadsheet is linked from Canvas
  - Some papers have multiple co-leads
  - Due Monday 10/9 at 11:59 pm
- Warm-up assignment
  - Posted on Canvas
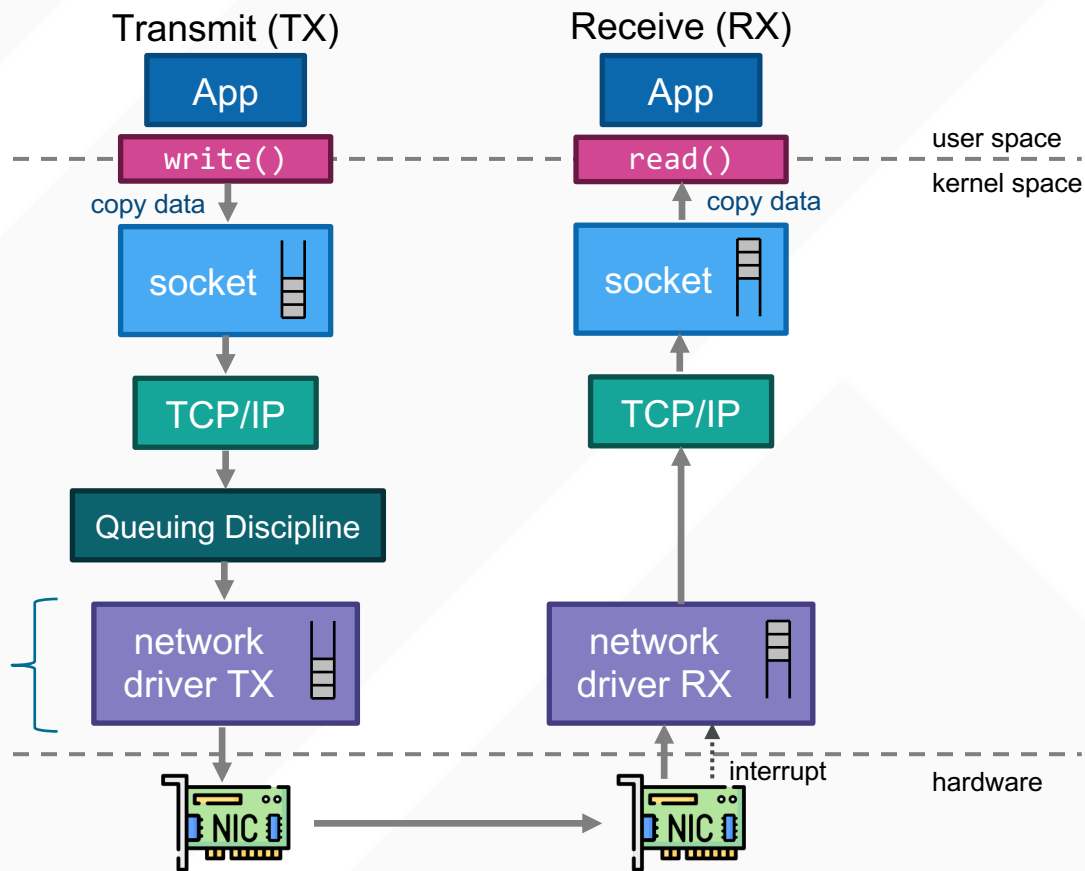 - Due Monday 10/16 at 11:59 pm

Prep for Tuesday

- FaRM discussion
    - Read FaRM and submit review as usual
- Where do research ideas come from? To prepare, consider:
    - What is a systems paper you have really liked?
    - Why did you like it?
    - How do you think the authors came up with the idea?
   - (no need to submit anything)

Network Stacks

Linux's Network Stack

- A simplified version
- Kernel data structure for packets: sk_buff
- Overheads:
  - Copying data
  - Context switches
  - Interrupts
  - Lots of queueing
- For more details, see "Understanding Host Network Stack Overheads" [SIGCOMM '21]

Transmit (TX)

Receive (RX)

App

App

`write()`

`read()`

user space

kernel space

copy data

copy data

socket

socket

TCP/IP

TCP/IP

Queuing Discipline

Device driver (NIC-specific)

network driver TX
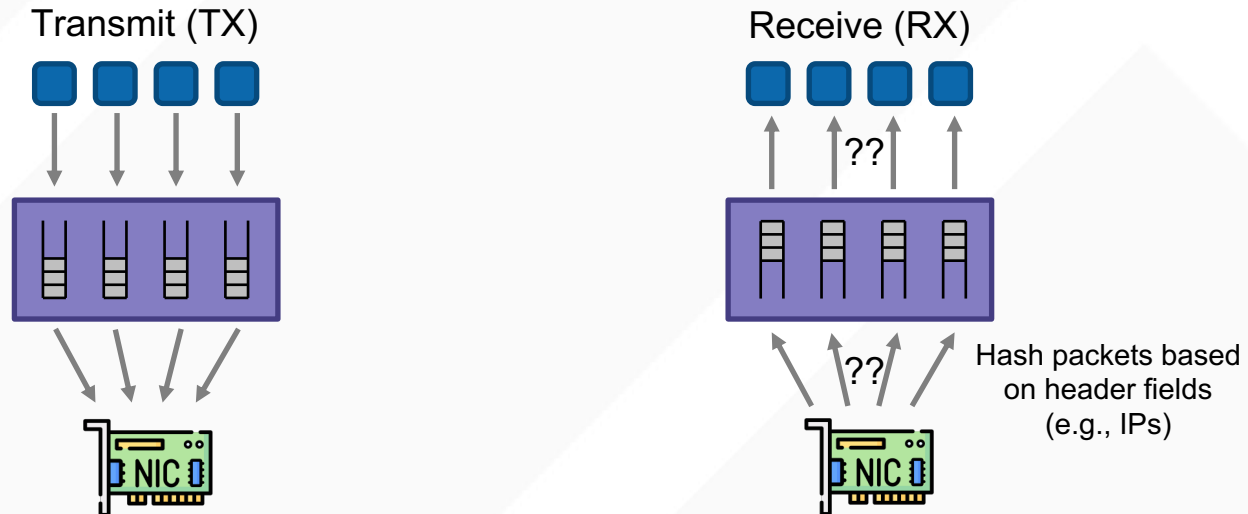
network driver RX

interrupt

hardware

NIC

NIC

Data Plane Development Kit (DPDK)

- One example of kernel bypass
- DPDK data structure for packets: rte_mbuf
- Differences from Linux:
  - Much less functionality
  - No copying
  - Polling instead of interrupts
  - Network stack runs in userspace
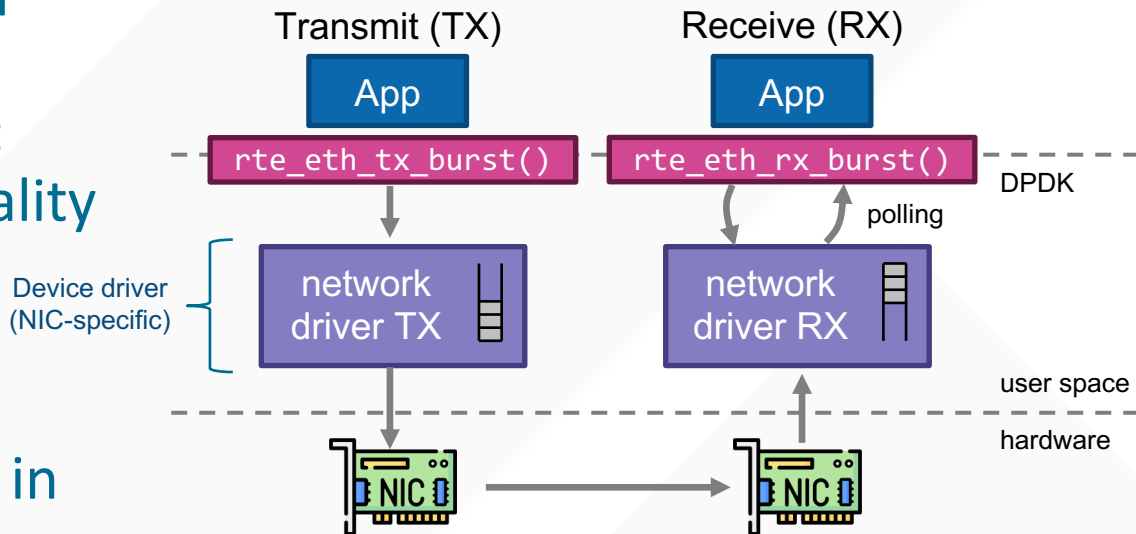
Handling Multicore

- One NIC queue per core
- Transmit path: each core sends to its own queue
- Receive path:
  - Flow-consistent hashing using receive-side scaling (RSS)

Transmit (TX)

Receive (RX)

??

??

Hash packets based
on header fields
(e.g., IPs)

NIC

NIC