Network Metrology:
Traffic measurements

This class

- Motivation and definitions
  - Why measure traffic?
  - Which traffic properties to measure?
- Tools for measuring traffic
  - Packet capture
  - Interface counts
  - Flow capture

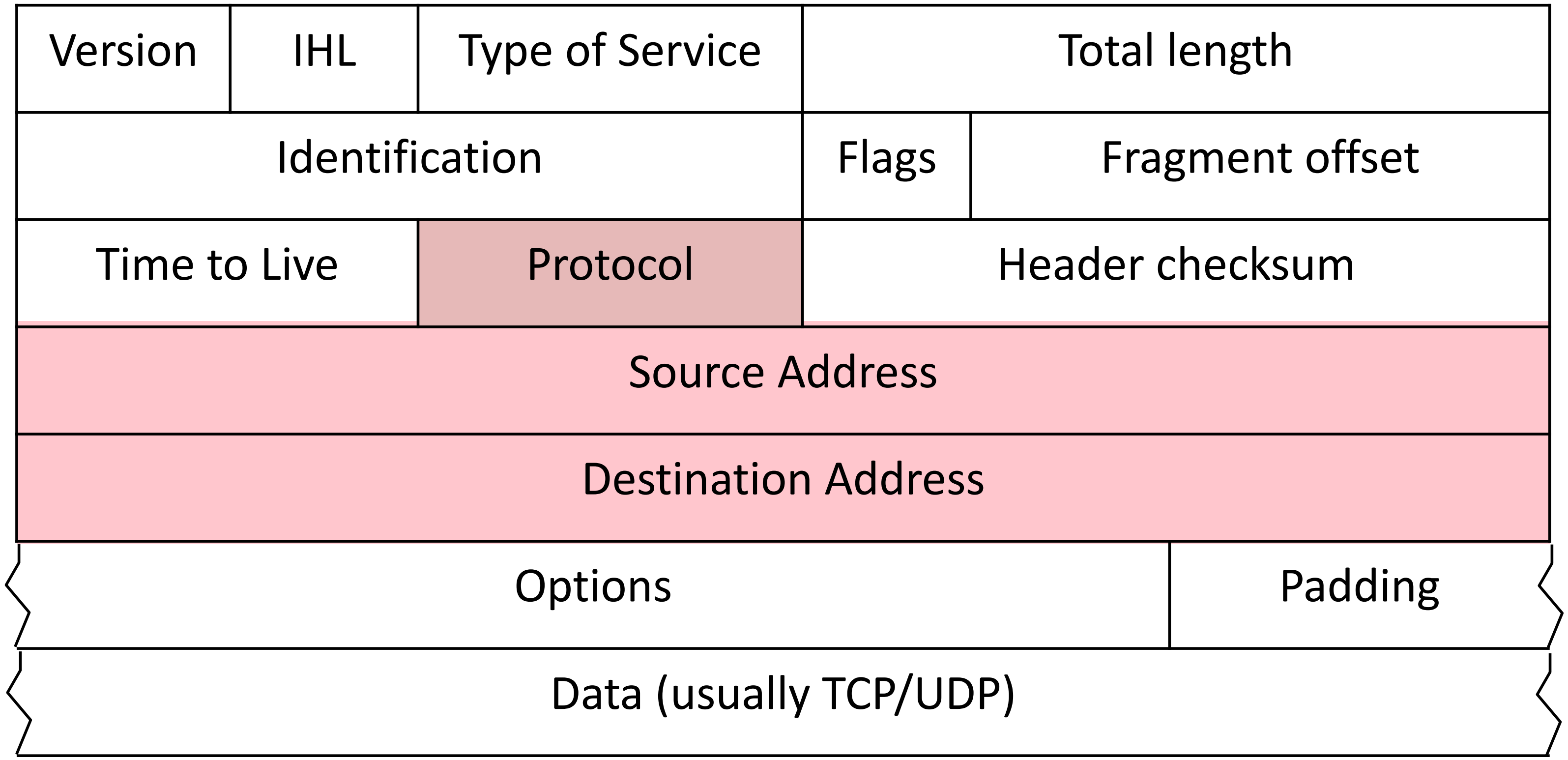
Why measure traffic?

- Performance analysis
- Anomaly and intrusion detection
- Network engineering

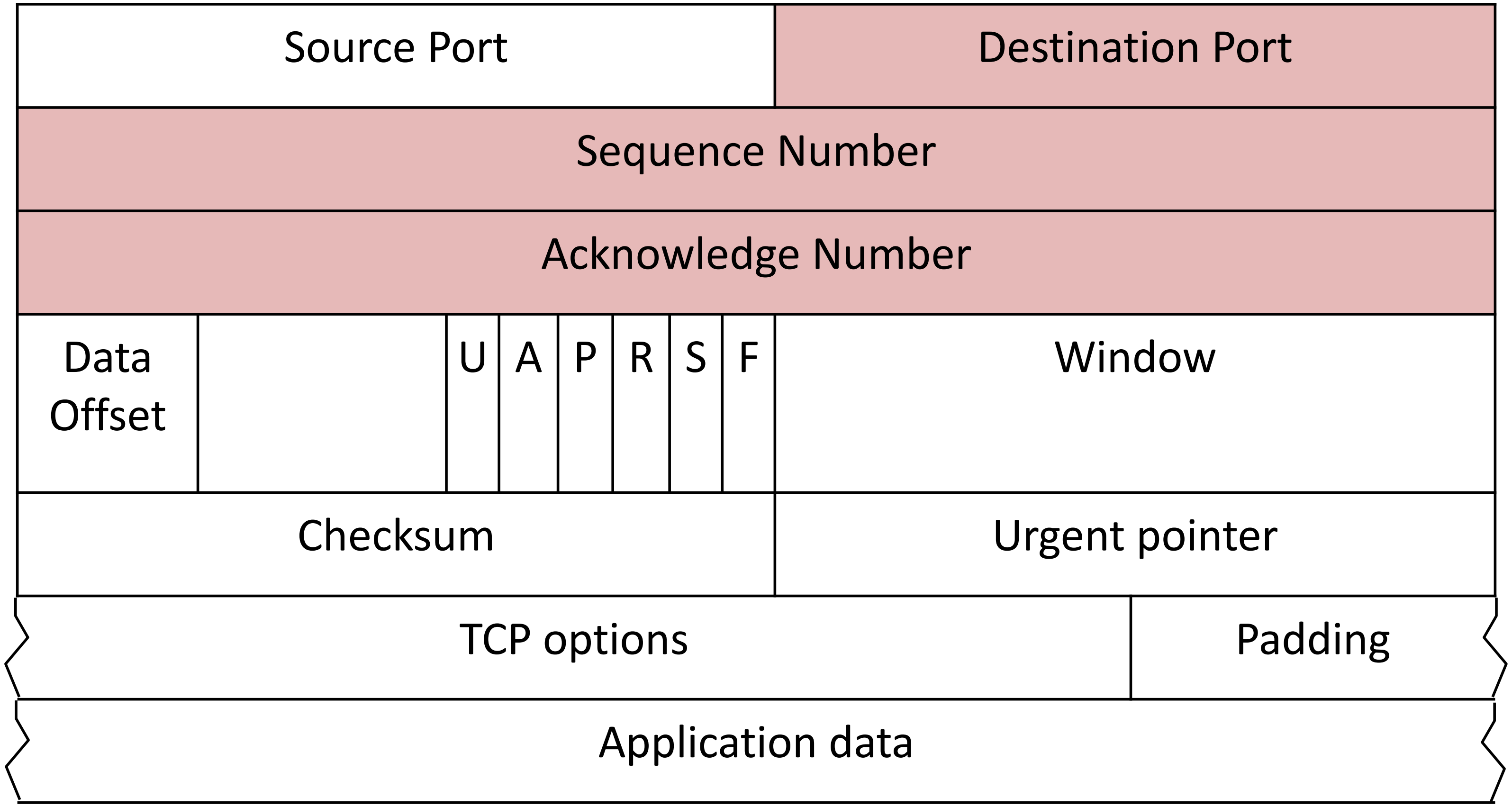
Traffic at different granularities

- IP-level packets
  - Capture per-packet information
- Flows
  - Statistics of packets grouped into flows
- Network interface
  - Statistics of packets that traverse a network interface

IP header


Version
IHL
Type of Service
Total length
Identification
Flags
Fragment offset
Time to Live
Protocol
Header checksum
Source Address
Destination Address
Options
Padding
Data (usually TCP/UDP)

TCP header


Source Port
Destination Port
Sequence Number
Acknowledge Number
Data Offset
U
A
P
R
S
F
Window
Checksum
Urgent pointer
TCP options
Padding
Application data

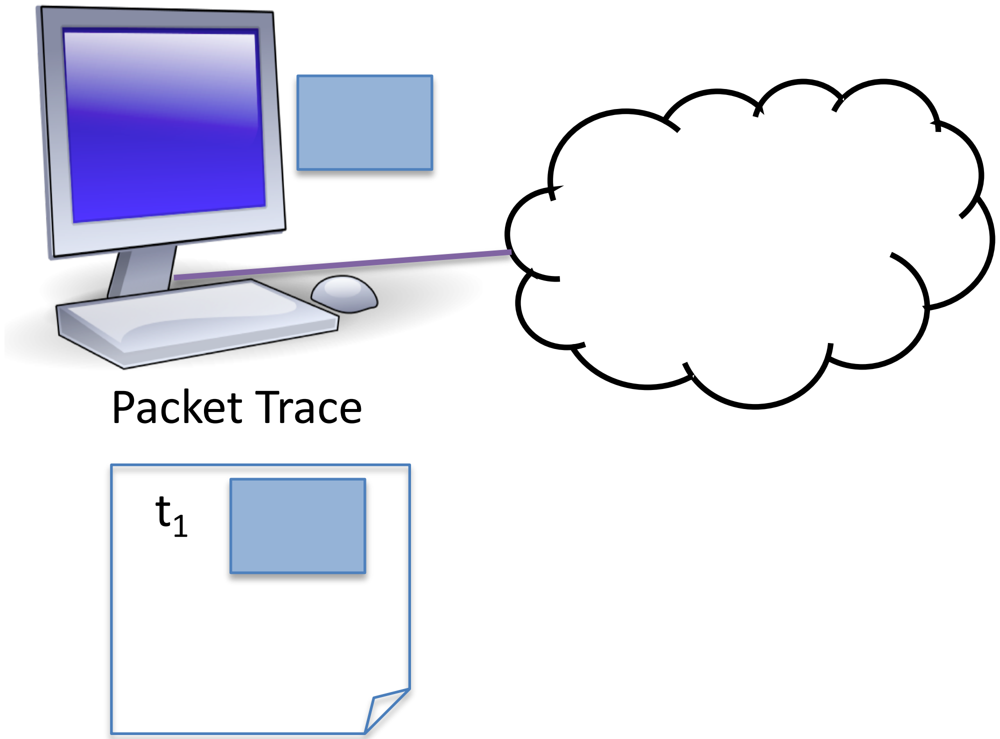
Traffic properties

- Number of packets
- Number of bytes
- Packet arrivals
- Flow arrivals
- Flow duration, size
- Flow performance (e.g., RTT, data rate)

PACKET CAPTURE

Packet capture on end systems

- Basic method
  - Capture and record packets passing through an interface



Tools

- tcpdump
  - Command-line packet capture
- libpcap
  - C/C++ library for packet capture
- Wireshark
  - Packet capture and analysis

Parameters: monitored interfaces

- Example
  - `tcpdump -i eth0`

Parameters: packet filters

- Packets to collect
  - E.g., TCP port 80 packets or packets to a given IP address
- Part of packets
  - E.g., IP/TCP headers

Example output: tcpdump

Receive window

Timestamp Source IP Source port Sequence number

```
11:04:57.409674 IP renataceiraair2.62701 >
www.ewr1.nytimes.com.https: Flags [S], seq 631064667, win 65535,
options [mss 1460,nop,wscale 5,nop,nop,TS val 2269480396 ecr
0,sackOK,eol], length 0
```

Destination IP Destination Port TCP Flags

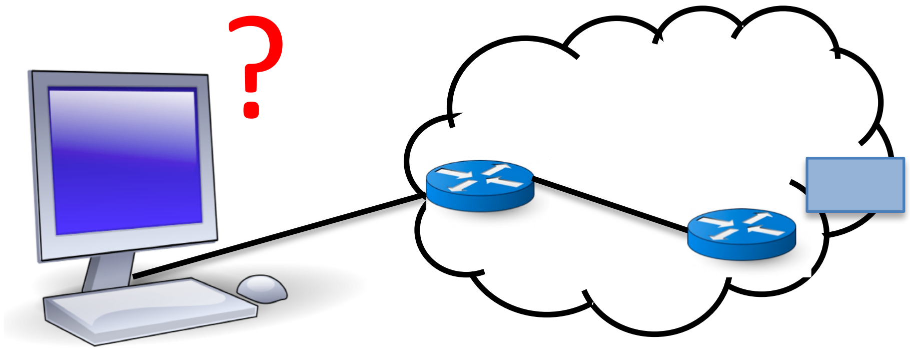
Possible measurement artifacts

- Dropped packets are common under high utilization
  - Inspect report of dropped packets
- Other less frequent artifacts
  - Fail to report drops
  - Falsely report drops
  - Duplicate packets
  - Re-ordered packets
  - Misfilter

How to capture packets of a network?

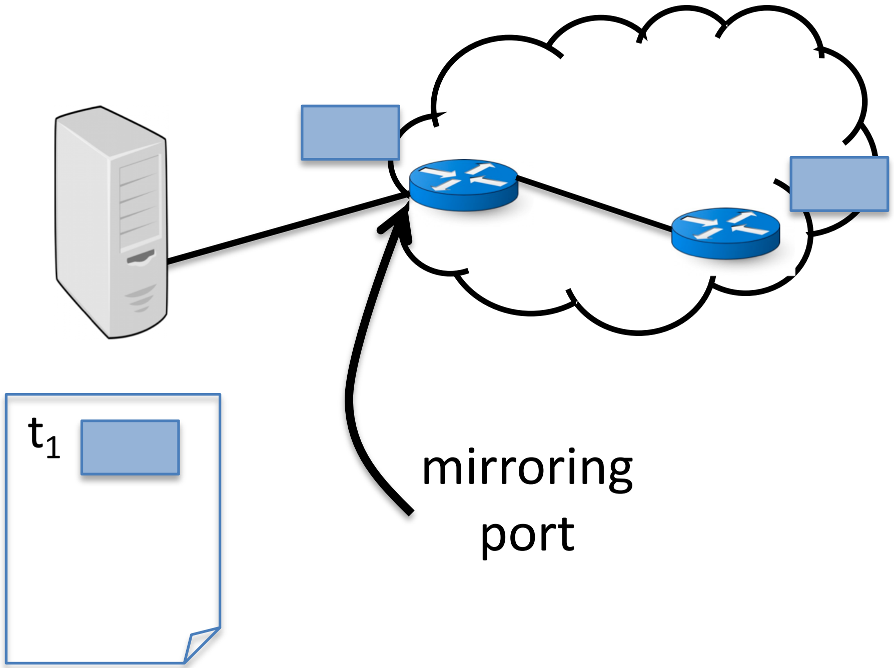
- In broadcast LANs (e.g., WiFi)
  - Set interface in promiscuous mode
  - Then, same as end system packet capture

How to capture packets on point-to-point links?



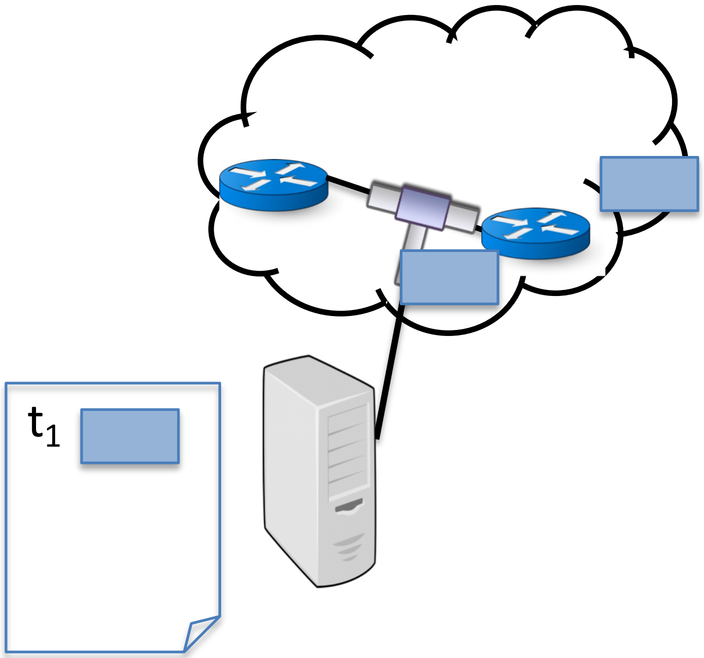
Port mirroring

- Basic method
  - Copies packets from one or more ports to a mirroring port
  - Run packet capturing tool on host connected to mirroring port



Network Tap

- Basic method
  - Electrical or optical splitter on monitored link
  - Monitoring host with specialized network interface and interface driver



Comparison

Port mirroring

- Pros
  - Easy to setup
  - Low cost
- Cons
  - Hardware and media errors are dropped
  - Packets may be dropped at high utilization

Tap

- Pros
  - Monitor all packets
  - Eliminates risk of dropped packets
- Cons
  - Expensive

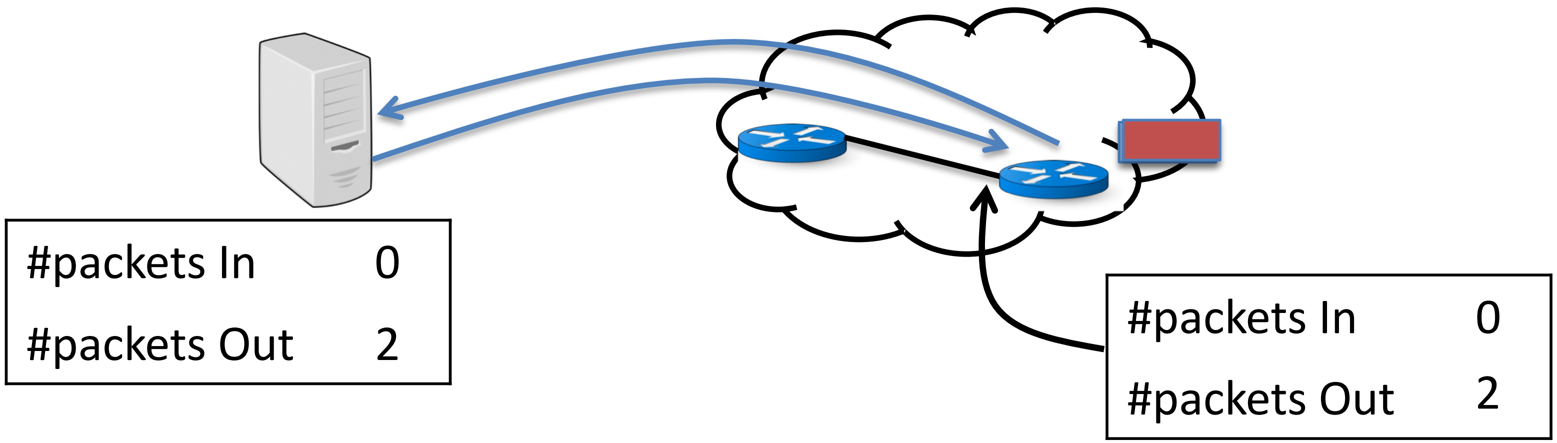
High-speed capture with commodity hardware

- Key idea
  - Direct access to NIC (i.e., bypass kernel)
  - Parallelism
- Tools
  - TStat
  - ntop
  - WAND

INTERFACE COUNTS

Interface counts

- Basic method
  - Routers log simple statistics (bytes/packets)
    - Total values since interface initialized
  - Request statistics using SNMP (MIB-II MIB)



Example properties

- Number of In/Out bytes (total, unicast, non-unicast)
- Number of In/Out packets (total, unicast, non-unicast)
- Number of In/Out discarded/corrupted packets

Interface counts: Pros and Cons

- Pros
  - Supported on all networking equipment
  - Little performance impact on routers
  - Little storage needs
- Cons
  - Missing data (SNMP uses UDP)
  - Polling makes it hard to synchronize data from multiple interfaces
  - Coarse-grained measurements

FLOW CAPTURE

IP Flows

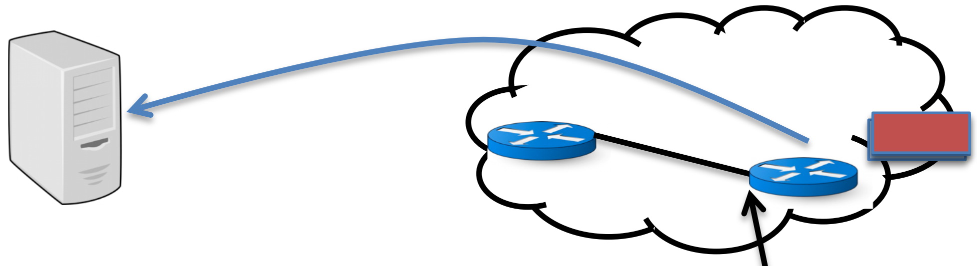
- Set of packets with common properties
  - Definition can vary
    - Traditional 5-tuple: src IP, dst IP, src port, dst port, protocol
    - Packets from one ingress to an egress point
- Packets that are "close" together in time
  - Maximum spacing between packets (e.g., 15 sec, 30 sec)

Flow ≠ application session

- Application session may be composed of multiple flows
- Packets in application session may not follow same links
- Hard to measure application session inside the network

Capturing flow statistics in routers

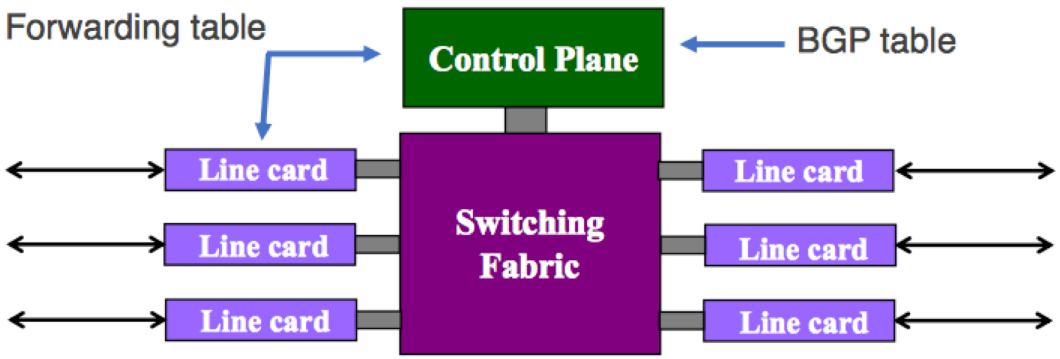
- Basic method
  - Specify set of properties that define a flow
  - Router log statistics per flow (flow records)
  - Push flow records to collecting process (IPFIX)



flow id	#packets
	1
	2

Flow records: Flow identifier

- Packet header information
  - Source and destination IP addresses
  - Source and destination TCP/UDP port numbers
  - Other IP & TCP/UDP header fields (e.g., protocol, ToS bits)
- Routing information
  - Input and output interfaces
  - Source and destination IP prefix (mask length)
  - Source and destination autonomous system numbers

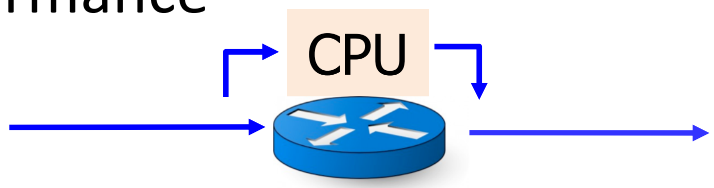


Flow records: Flow properties

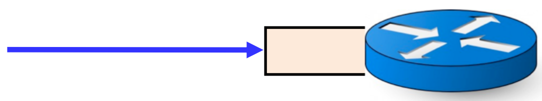
- Aggregate traffic information
  – Start and finish time of the flow (time of first & last packet)
  – Total number of bytes and number of packets in the flow
  – TCP flags (e.g., logical OR over the sequence of packets)

Collecting flow records

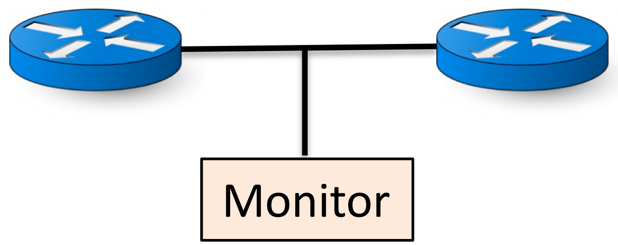
- Route CPU that generates flow records
  - May degrade forwarding performance



- Line card that generates flow records
  - More efficient to support measurement in each line card

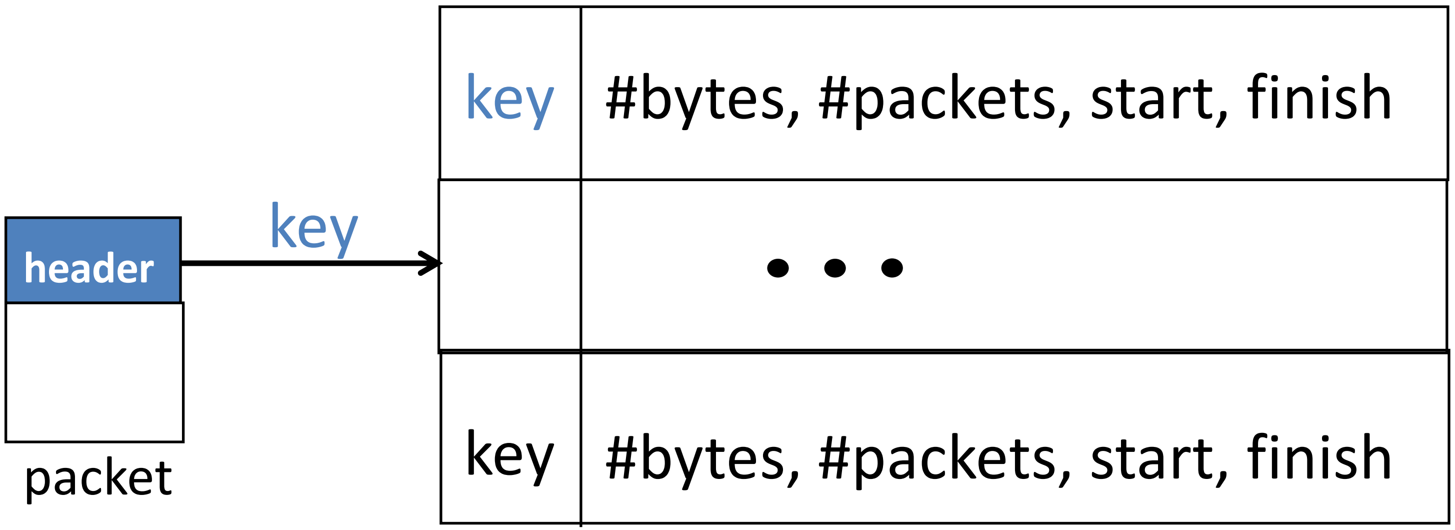


- Packet monitor that generates flow records
  - Often requires third party equipment



Flow cache

- Maintain a cache of active flows
  - Storage of byte/packet counts, timestamps, etc.
- Compute a key per incoming packet
  - Concatenation of source, destination, port #s, etc.
- Index into the flow cache based on the key
  - Creation or updating of an entry in the flow cache

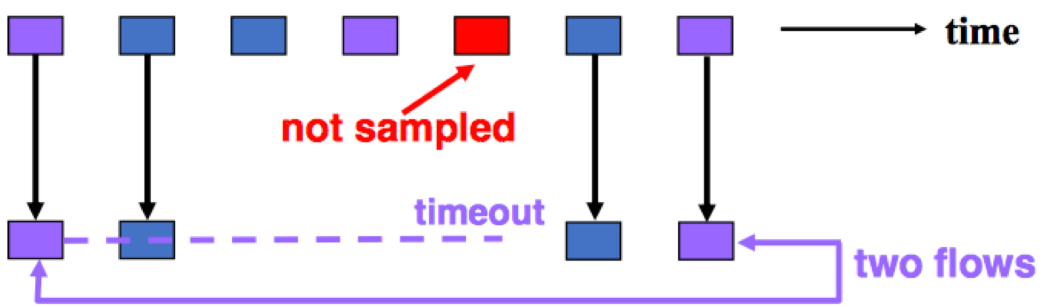


Evicting cache entries

- Flow timeout
  - Remove flows that have not received a packet recently
  - Periodic sequencing through the cache to time out flows
  - New packet triggers the creation of a new flow
- Cache replacement
  - Remove flow(s) when the flow cache is full
  - Evict existing flow(s) upon creating a new cache entry
  - Apply eviction policy (LRU, random flow, etc.)
- Long-lived flows
  - Remove flow(s) that persist for a long time (e.g., 30 min)
  - … otherwise flow statistics don’t become available
  - … and the byte and packet counters might overflow

Packet Sampling

- Packet sampling before flow creation
  - 1-out-of-m sampling of individual packets (e.g., m=100)
  - Creation of flow records over the sampled packets
- Reducing overhead
  - Avoid per-packet overhead on (m-1)/m packets
  - Avoid creating records for a large number of small flows
- Increasing overhead (in some cases)
  - May split some long transfers into multiple flow records



Tools

- In-router capture
  - Cisco NetFlow
  - Juniper JFlow
- Collection and post-processing
  - Flow-tools
  - ntop

Flow monitoring: Pros and Cons

Pros

- More details about traffic compared to counters
- Lower measurement volume than full packet traces
- Available on high-end line cards (Netflow, Jflow)
- Control over overhead via aggregation and sampling

Cons

- Less details than packet capture
  - No individual packet arrival times
  - No information on packet content
- Not uniformly supported (getting better with IPFIX)
- Computation/memory requirements for the flow cache

Using the traffic data in network operations

- Interface counts: everywhere
  – Tracking link utilizations and detecting anomalies
  – Generating bills for traffic on customer links
  – Inference of the offered load (i.e., traffic matrix)
- Packet monitoring: selected locations
  – Analyzing the small time-scale behavior of traffic
  – Troubleshooting specific problems on demand
- Flow monitoring: selective, e.g,. network edge
  – Tracking the application mix
  – Direct computation of the traffic matrix
  – Input to denial-of-service attack detection

Summary

- Packet capture
  - Detailed per-packet measurements
  - High collection overhead
- Interface counts
  - Coarse measurements per link
  - Low overhead, widely available
- Flow capture
  - More details than link counts, less than packet captures
  - Medium collection overhead controlled with sampling

References

- P. Tune, M. Roughan, “Internet Traffic Matrices: A Primer”, in H. Haddadi, O. Bonaventure (Eds.), Recent Advances in Networking, (2013)
  - http://sigcomm.org/education/ebook/SIGCOMMeBook2013v1_chapter3.pdf
- Jennifer Rexford, Network Measurement Lecture Notes, COS-561, Fall 2018.