Effectiveness of Redundancy Elimination of Modern Cellular Traffic

KyoungSoo Park

Department of Electrical Engineering, KAIST

Shinae Woo, Eunyoung Jeong, Shinjo Park

Jongmin Lee[†], Sunghwan Ihm[‡]

KAIST [†]SK Telecom, Inc. [‡]Princeton University

Increasing Demand for Cellular Traffic

Mobile devices

Faster cellular network speed

Deloitte: U.S. Could See $53 Billion in 4G Network Investments by 2016, Creating 771,000 New Jobs

New 4G products and services could drive additional growth

SK Telecom to invest W2.3 tril. for smartphones

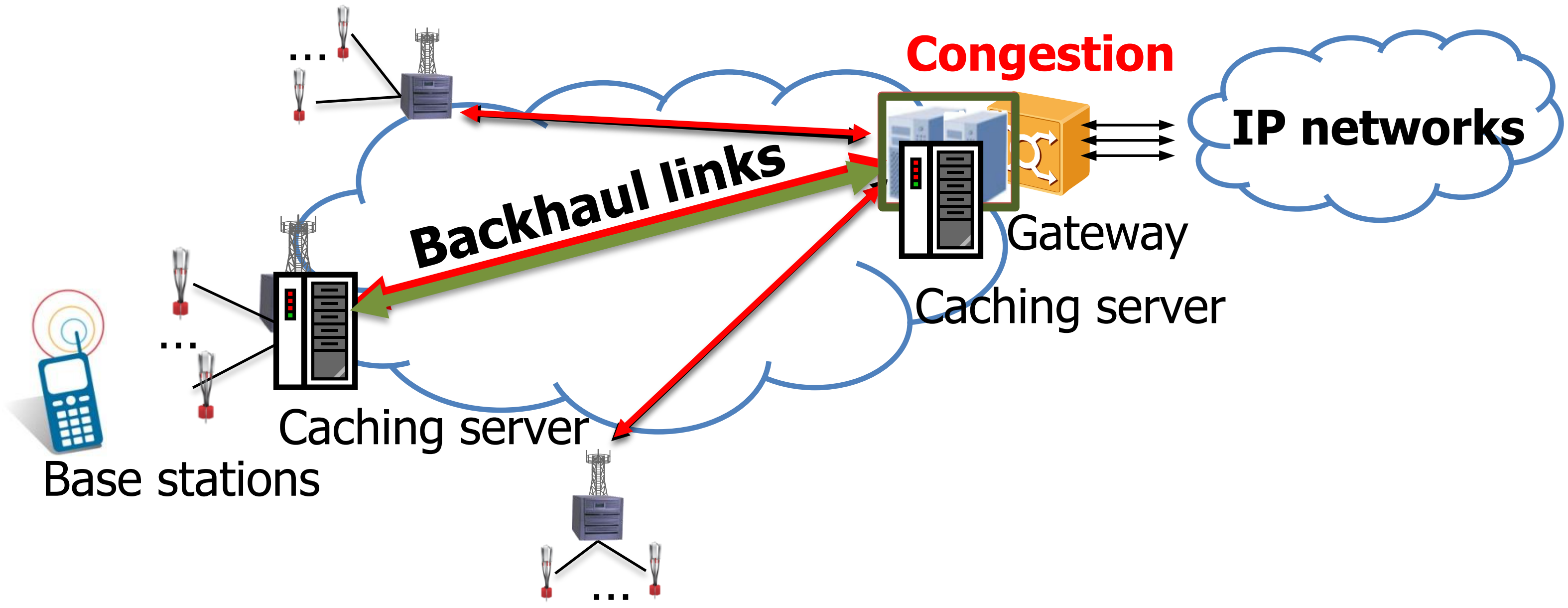
By Yoon Ja-young

RIL plans to invest $10 billion on 4G network: vendor

Press Trust of India | Updated On: December 20, 2012 08:32 (IST)

Caching over Cellular Architecture

- Centralized architecture of cellular networks
  - ✖ Possible congestion point
  - ✔ Aggregated traffic → caching benefit



Benefit of Caching for Cellular Network

- Previous works on caching in the wired Internet
  - 39% additional redundancy after Web caching applied[Spring2000]
  - 59% redundancy in small enterprise network[Anand2009]
  - 42~51% redundancy over wired Web traffic[Ihm2001]
- Few works on effectiveness of caching in cellular networks
  - Not much comparison between caching strategies
  - Not for the traffic over cellular backhaul traffic

Research Goals

- Characterization of modern cellular traffic
  - Characterization of traffic which impact on caching strategies
  - Characterization of redundancy
- Effectiveness of available caching strategies
  - Web caching / prefix-based Web caching
  - TCP flow-based redundancy elimination

Monitoring System Challenges

	Hardware-based solutions	Software-based solutions
▪ High performance • 10 Gbps cellular backhaul links • No single packet drop	✓	✗
▪ Real time flow reconstruction • Creating logs for TCP/HTTP • Creating logs for caching analysis	✗	✓
▪ Inexpensive solution • No expensive specialized H/W	✗	✓

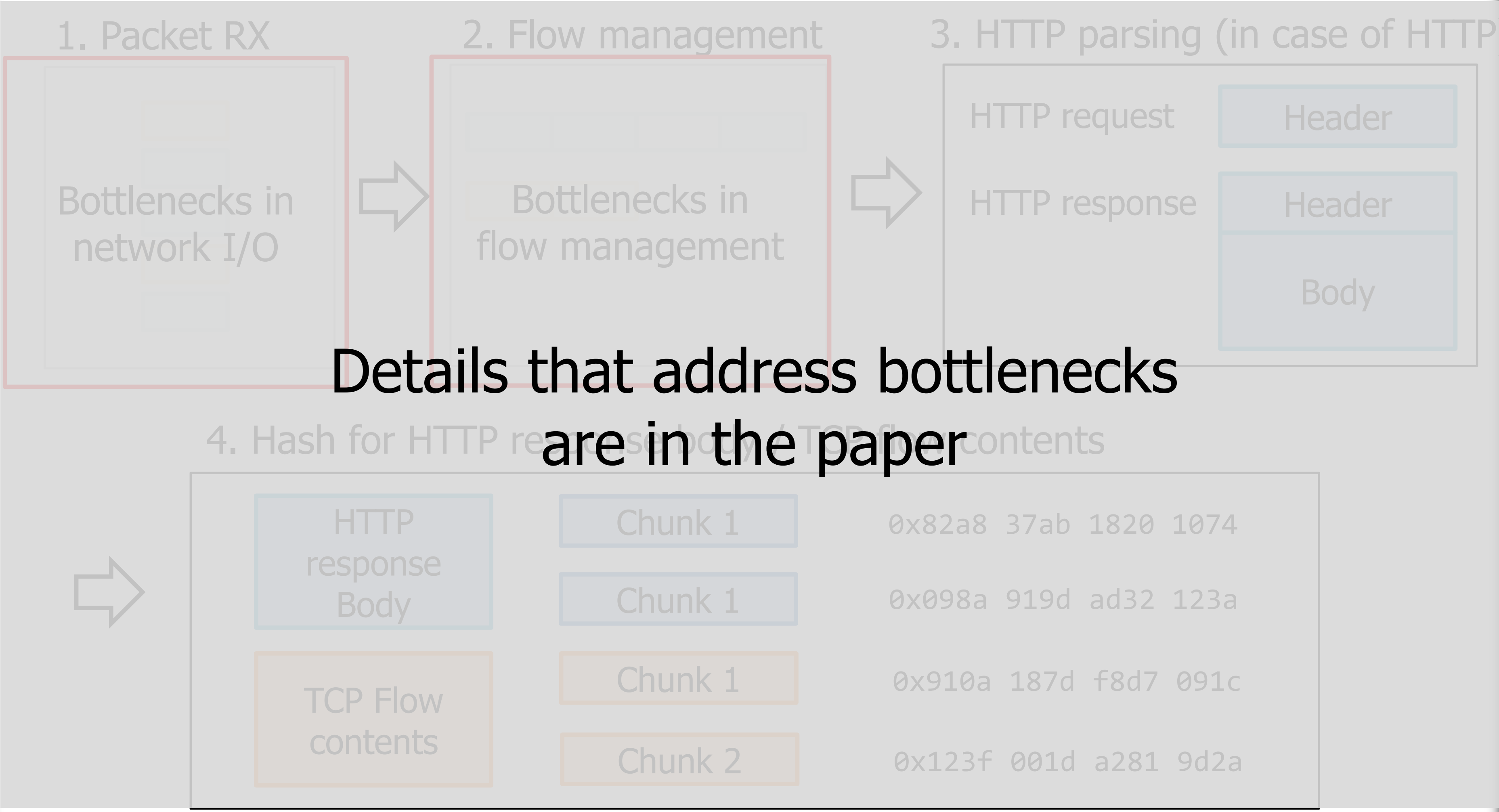
Contributions

- MonBot: a highly-scalable *flow-level traffic analysis* system on commodity hardware
  - High-performance: millions of packets/s, 100K concurrent flows
  - Flexibility: TCP, HTTP, content-level logs in real time
  - Inexpensive: multi-core, commodity 10G NIC ($4000)

- Content redundancy analysis on *real* 3G traffic
  - 8.3 billon TCP connections, 590 billion packets, 370 TBs
  - 59% content-level redundancy
  - The largest 9.4 % flows account for 68.4 % of redundancy

MonBot: Software-based Flow Monitoring System

MonBot Workflow

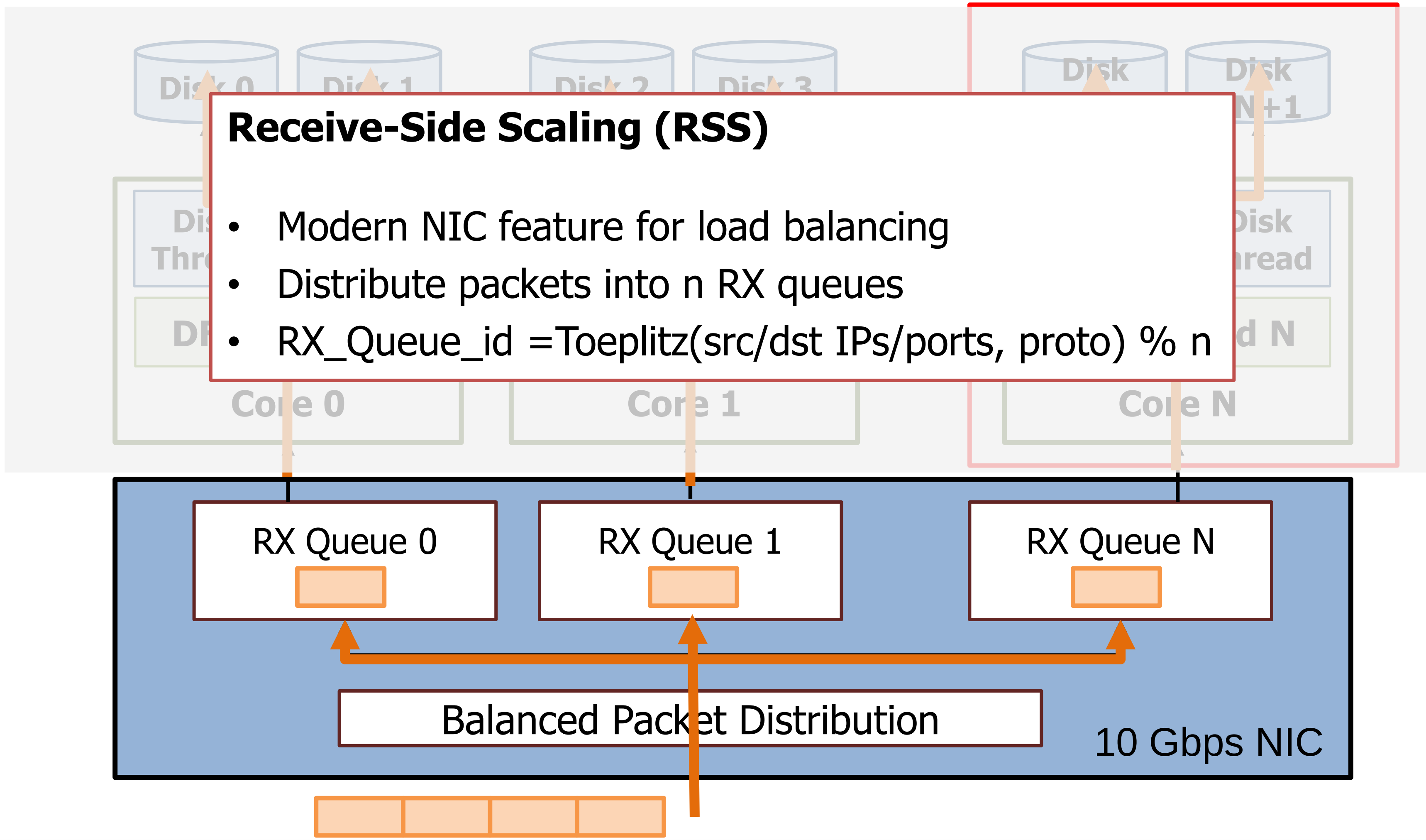


Details that address bottlenecks are in the paper

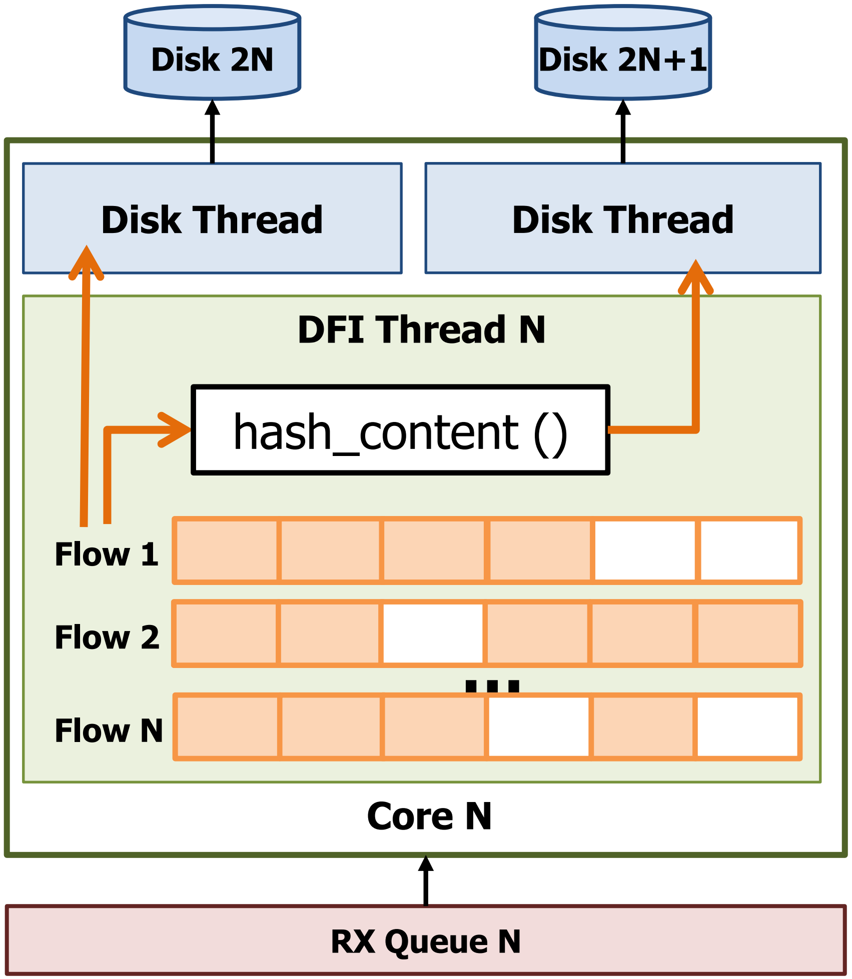
MonBot: Parallel Flow Monitoring

Receive-Side Scaling (RSS)

- Modern NIC feature for load balancing
- Distribute packets into n RX queues
- RX_Queue_id =Toeplitz(src/dst IPs/ports, proto) % n



Processing Unit in a CPU Core



Disk threads
- Aggregate and store logs into disk
- Each thread is mapped to 1 disk

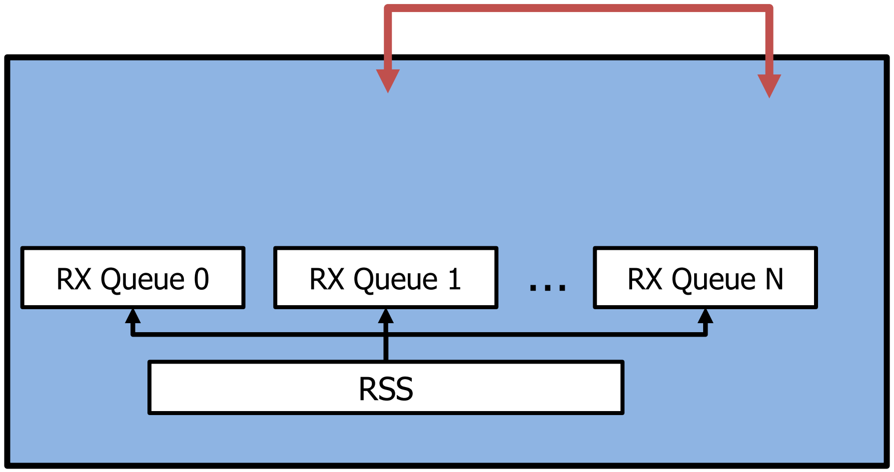
Deep *Flow* Inspection (DFI) thread
- Manage flow contents
- Create TCP/HTTP transaction logs
- Calculate SHA1 hash for flow contents

Problem with RSS

- Problem with RSS
  - Different RX queues for packets in the same connection

Software hashing to access the connection context

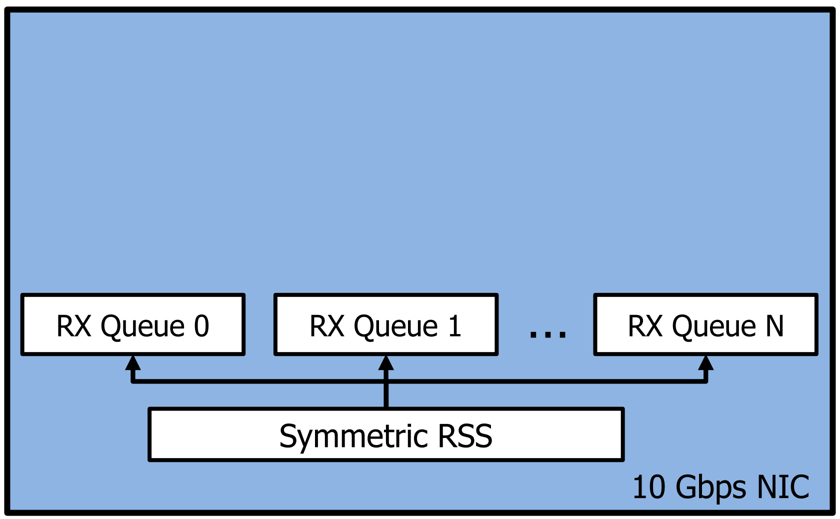
❌ Inefficiency from shared locks



Symmetric RSS

- Symmetric RSS

Same connection to the same core ✔ Line-rate processing from NIC h/w



RSS_Hash (src→dest:protocol,**RSK**)

==

RSS_Hash (dest→src:protocol,**RSK**)

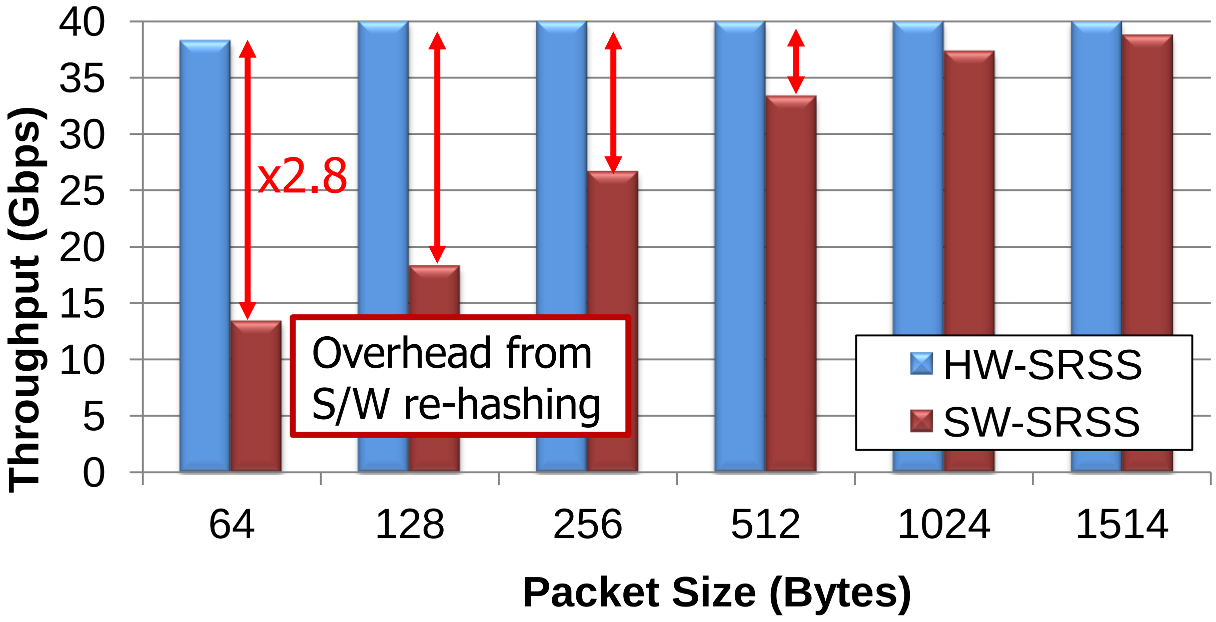
i) $RSK_{1..15} = RSK_{17..31} = RSK_{33..47}$
$= RSK_{49..63} = RSK_{65..79} = RSK_{81..95}$
$= RSK_{97..111} = RSK_{113..127}$

ii) $RSK_{16} = RSK_{48} = RSK_{80} = RSK_{96}$

iii) $RSK_{32} = RSK_{64}$

Symmetric RSS Performance

- Performance for flow-level load balancing
  - HW-SRSS: H/W load balancing (PSIO + Sym RSS)
  - SW-SRSS: S/W rehashing (PF_RING + RSS)



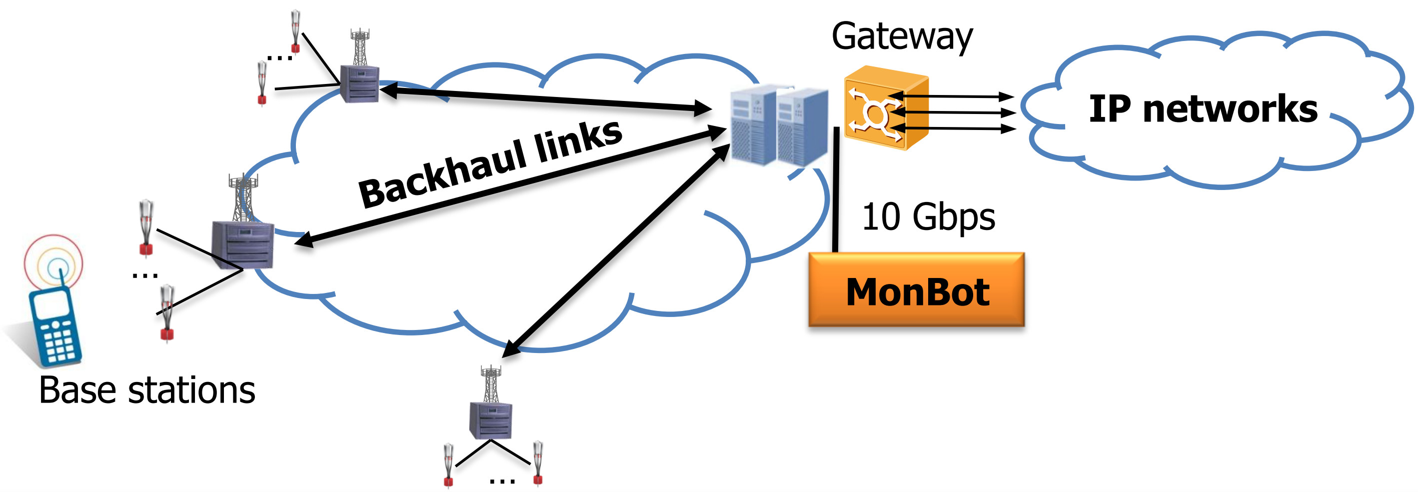
6 core Xeon X5690 * 2
144 GB RAM
Dual-port 10Gbps NIC
(Intel 82599)c

Sym-RSS allows high performance for small sizes of packets.

Redundancy over Commercial Cellular Traffic

Measurement Environment

- Analysis on the commercial 3G traffic for a week
  - In the largest cellular ISP in South Korea
  - Full flow-level analysis at a 10G link
- Traffic volume
  - 8.3 billion TCP connections
  - 370 TB in bytes or 590 billion packets



Overall Traffic Characteristics

- Finding1: HTTP is the dominant protocol
  - 75% of the downlink traffic is HTTP
  - Majority of the mobile apps use HTTP
- Finding2: Traffic volume driven by human mobility
  - Local peaks at morning rush hour and lunch time
- Finding3: Most flows are small in the cellular networks
  - Only 9.4% of flows are larger than 32 KB
  - But large flows contribute to 93.7% of bytes

Web Caching vs. Prefix-based Web Caching

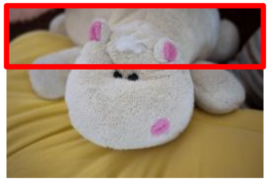
- Problem of Web caching

http://ndsl.kr/hippo.jpeg

Miss

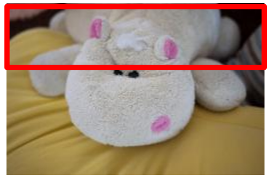
http://ndsl.kr/cute.jpeg

- Prefix-based Web caching
  - Commercially used to solve alias problem
  - Prefix key: (hash of the first n bytes, content length)
  - False positive: two different objects, same prefix key



http://ndsl.kr/hippo.jpeg
(0x0785 332a, 24 KB)

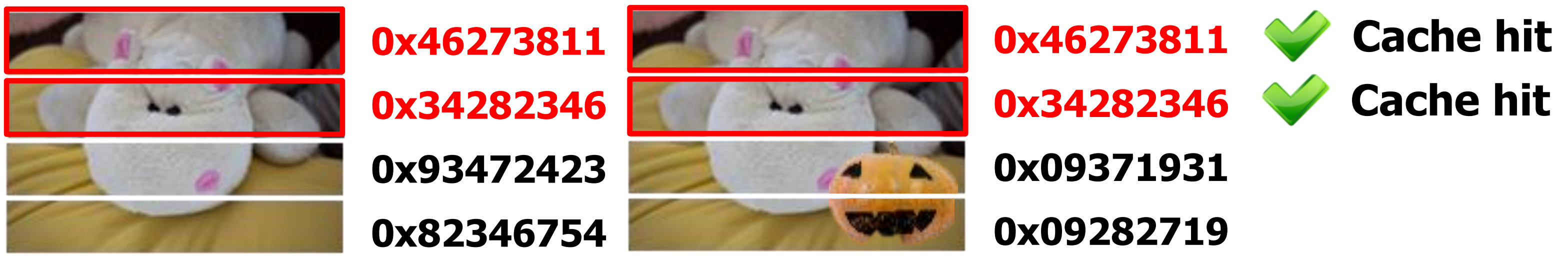
Hit



http://ndsl.kr/cute.jpeg
(0x0785 332a, 24 KB)

TCP Flow-based Redundancy Elimination

- TCP-RE
  - Content-based caching
  - Use the hash of content chunk as the key
  - Fine-grained suppression



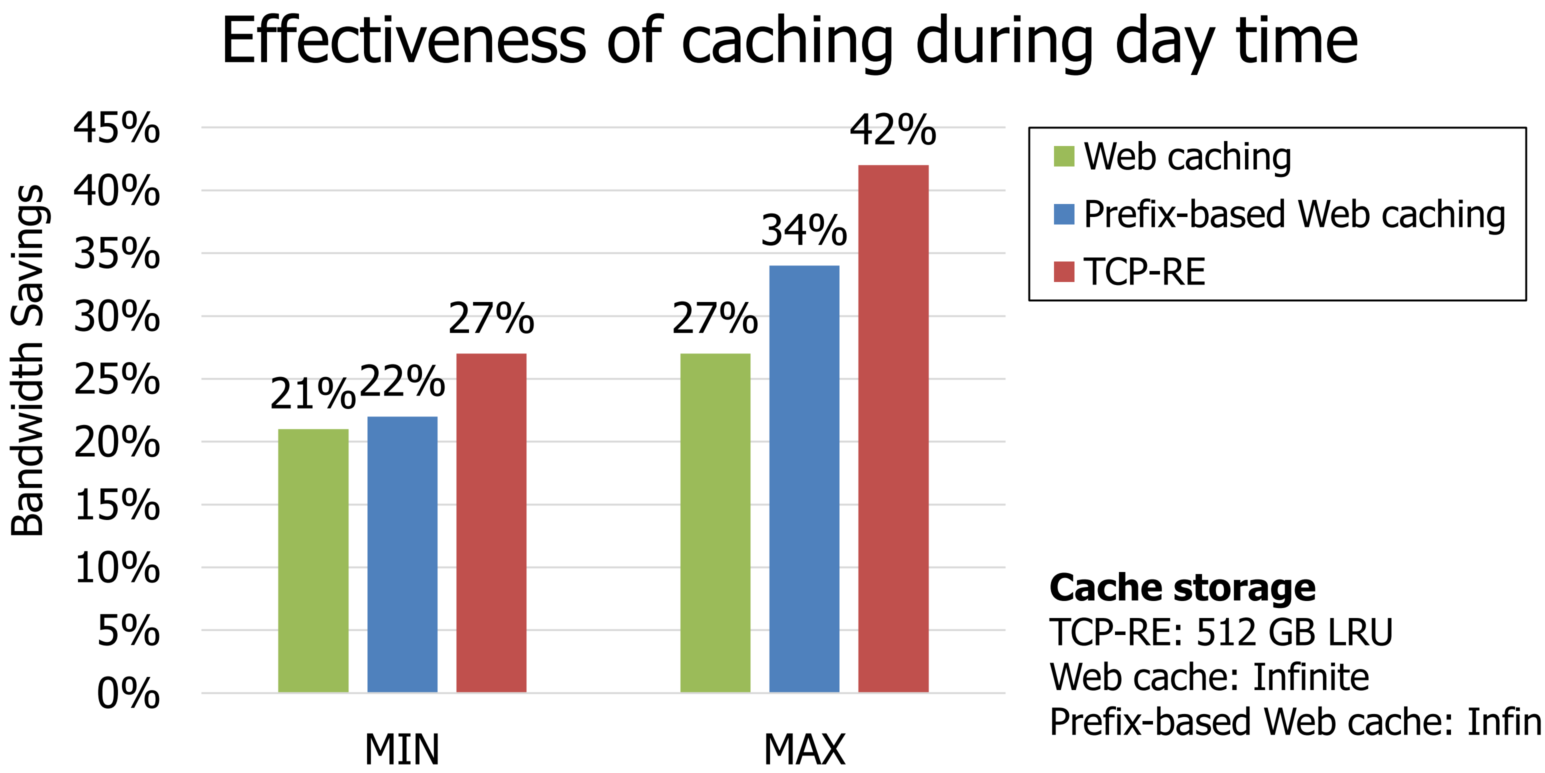
- ✔ Smaller index size than packet level
- ✔ Policy control for different set of flows
- ✘ Need flow management

Finding 1: HTTP is dominant

- HTTP is the dominant application (75%)

- Question: will Web caching be enough?
- What is the performance of various caching schemes?

Effectiveness of Various Caching Schemes

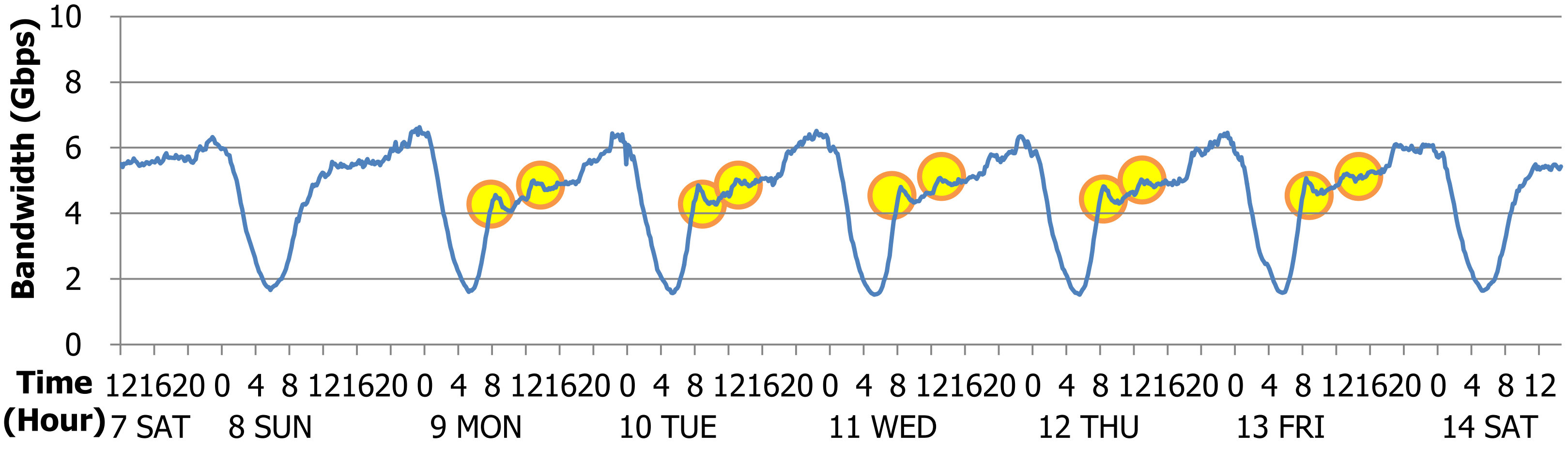


Cache storage
TCP-RE: 512 GB LRU
Web cache: Infinite
Prefix-based Web cache: Infinite

TCP-RE outperforms even with the smaller cache size than Web caching.

Finding 2: Peak Traffic with Human Mobility

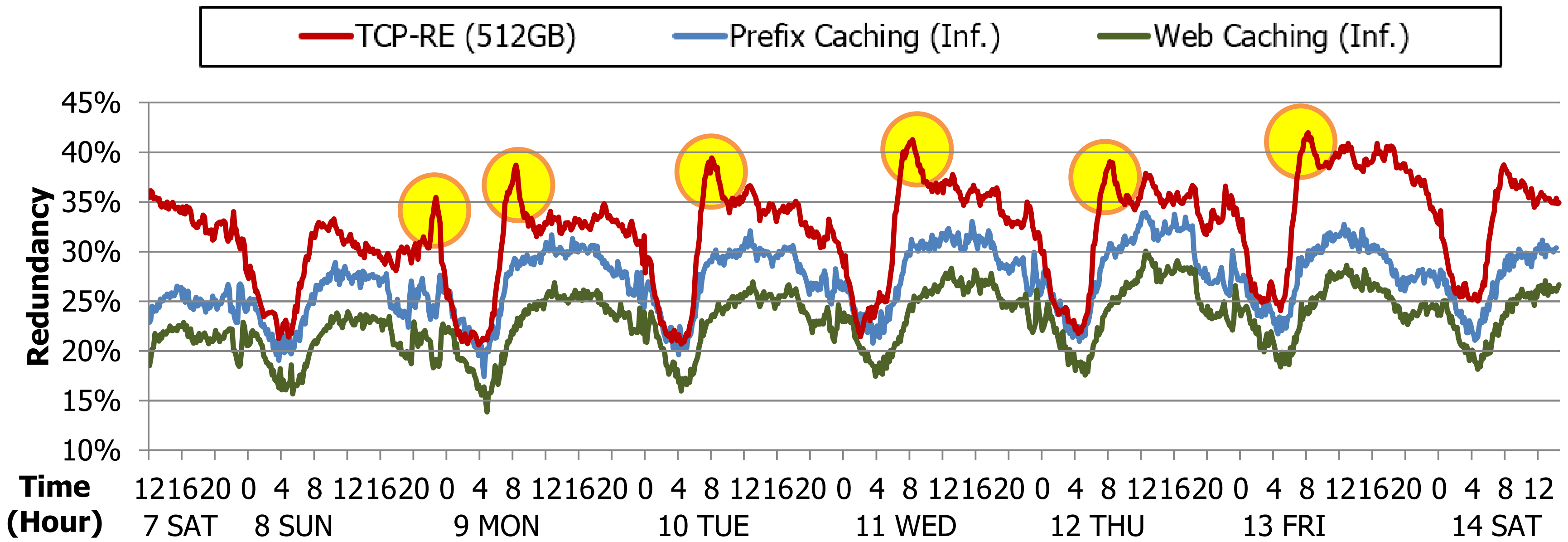
- Peak traffic with human mobility
  - Morning rush hour / lunch time



- Most effective caching strategy for local peaks?

Peak Redundancy from TCP-RE

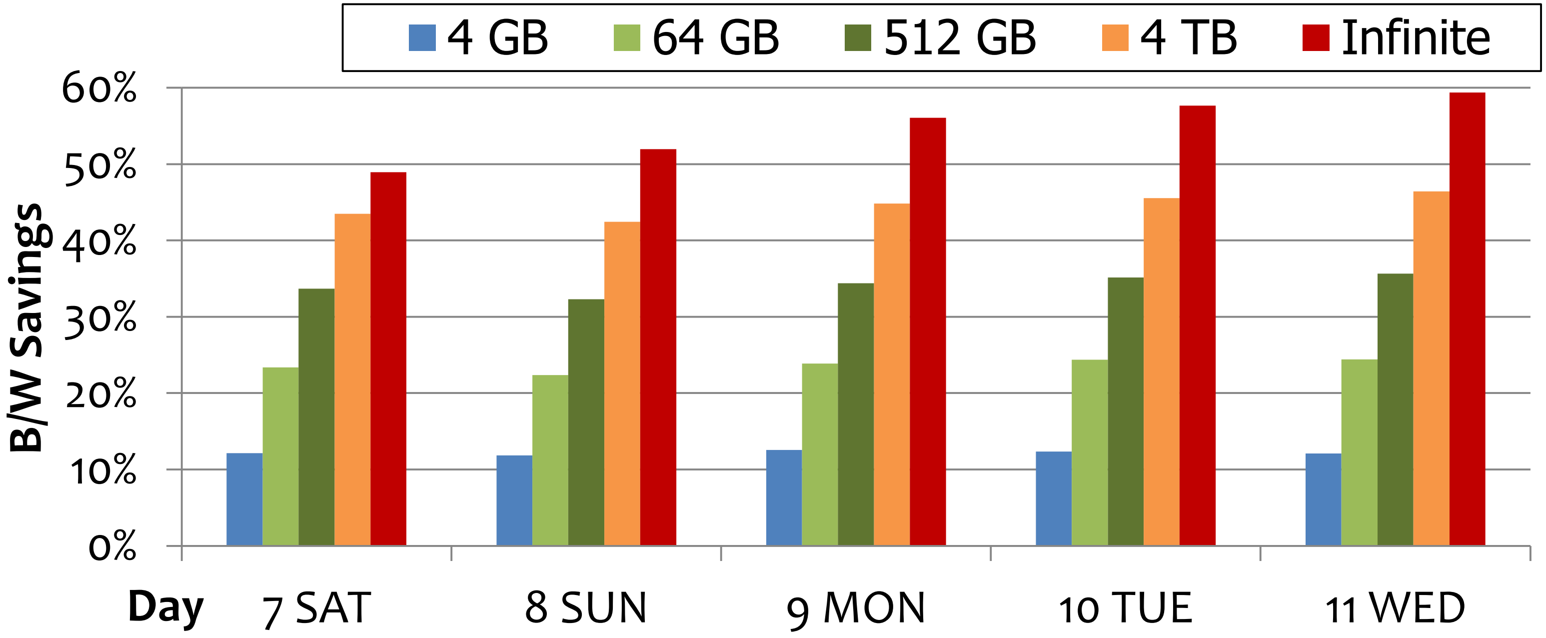
Cache storage
TCP-RE: 512 GB LRU
Web cache: Infinite
Prefix-based Web cache: Infinite



TCP-RE can detect the peak redundancy.

Redundancy Elimination for Various Cache Size

- TCP-RE with different size LRU cache

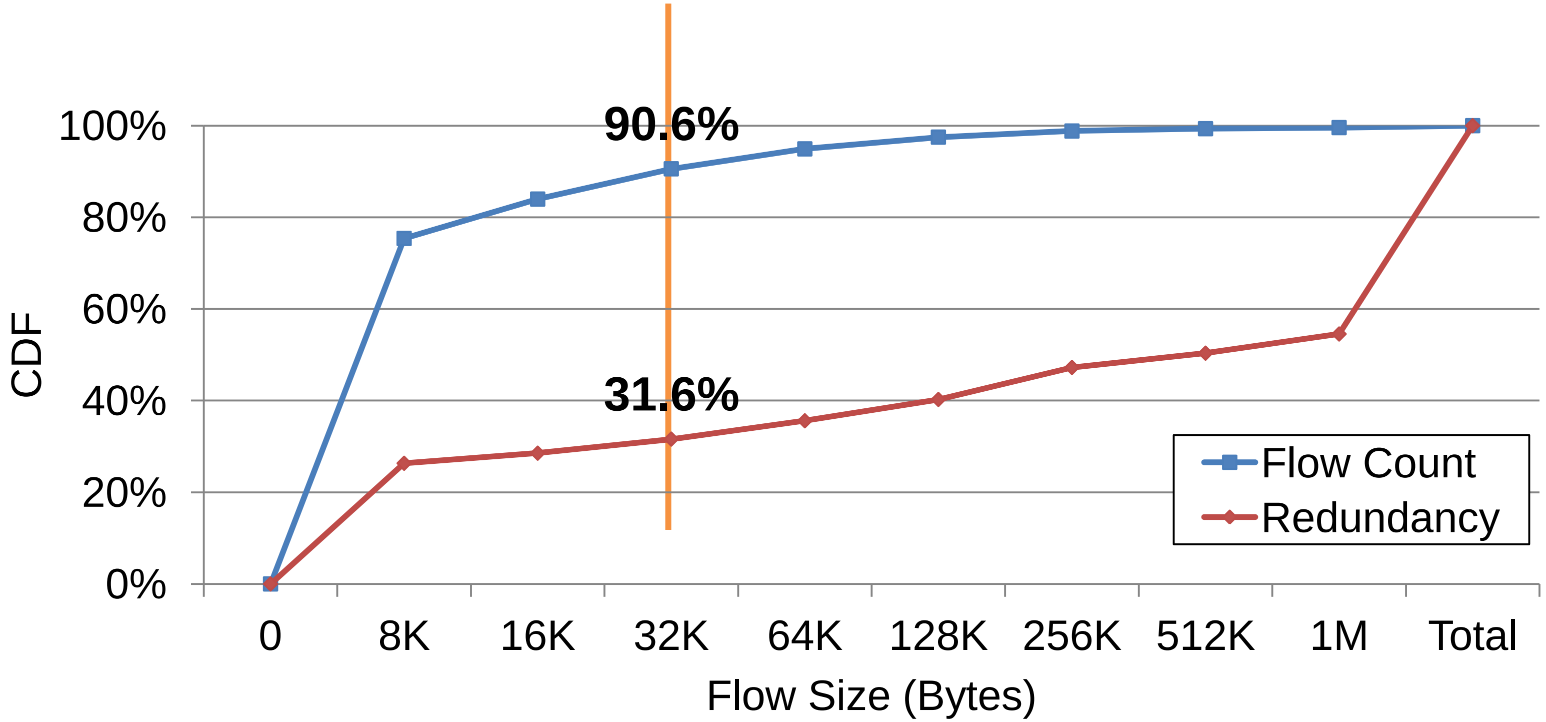


59 % of total traffic is redundant!

Finding 3: Small Large Flows Dominate BW

- Small large flows take up the majority of bytes
  - 9.4% of flows are larger than 32 KB
  - But it contributes 93.7 % of bytes
- Is the majority of redundancy coming from large flows?

Flow Size vs. Bandwidth Savings



Largest ***9.4 % flows*** *account for* ***68.4 % of redundancy***

Large flows: bigger opportunity ↑ smaller flow management cost ↓

More Details In Our Mobisys'13 Paper

- Detailed TCP flow / HTTP statistics
- Bandwidth savings for different cache sizes
- Bandwidth savings for different cache replacement policy
- Redundancy by HTTP content types
- False positive ratio of prefix-based Web cache
- The origin of redundancy
- Temporality of redundancy

"Comparison of Caching Strategies in Modern Cellular Backhaul Networks", Woo et.al., ACM Mobisys 2013

Conclusion

- MonBot, a software-based flow-level monitoring system
  - Flow-level functionality: hashing flow contents
  - High performance: 10 Gbps w/o packet drop
  - Symmetric RSS: flow-level load balancing
  - Inexpensive: using commodity server
- Implications on caching for mobile networks
  - TCP-RE can detect 59% redundancy
  - TCP-RE effectively reduce the local peak
  - Large flows provide significant bandwidth savings with a minimal flow management cost

Thank You