CC*DNI CAMPUS DESIGN NORTHERN NEW MEXICO COLLEGE

SCIENCE DMZ AND UNDERGRADUATE RESEARCH OPPORTUNITIES

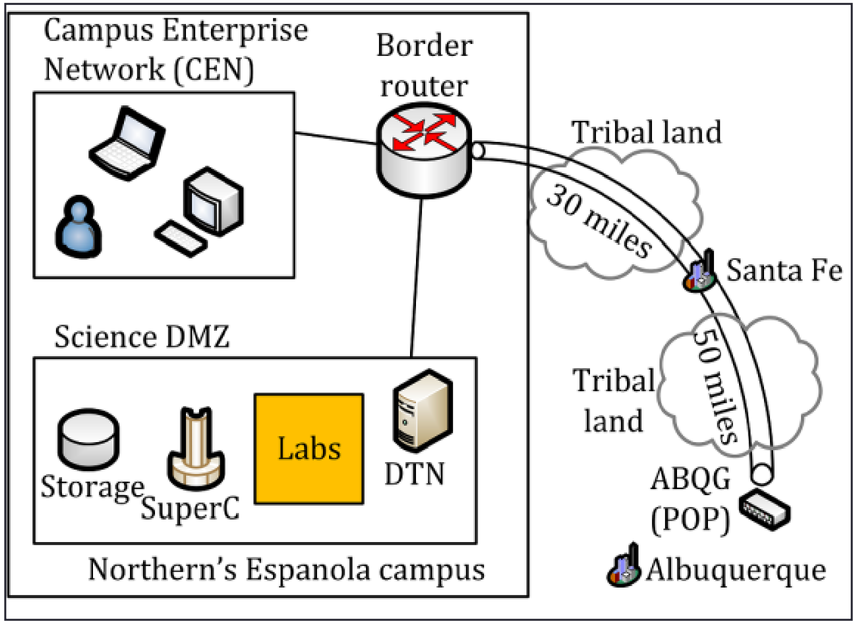
Jorge Crichigno
University of South Carolina

Northern New Mexico College

- Located in Espanola, NM
- “Under-resourced” institution; no network engineer; students helped deploy the new research network
- Oct. 2015 – Dec. 2017

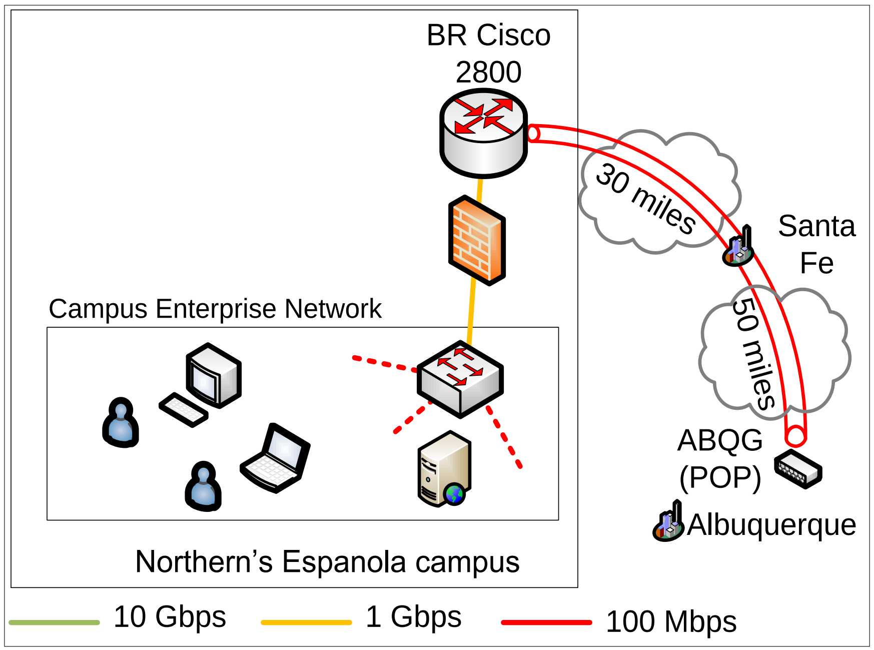
Northern's CC*DNI

- New border router to replace the older Cisco 2800 series
- Upgrades to increase intra-campus transfers from 100 Mbps to 10 Gbps
- Deployment of a Science DMZ and research network
- Increase external connectivity from 100 Mbps to at least 1 Gbps to Albuquerque (ABQG)



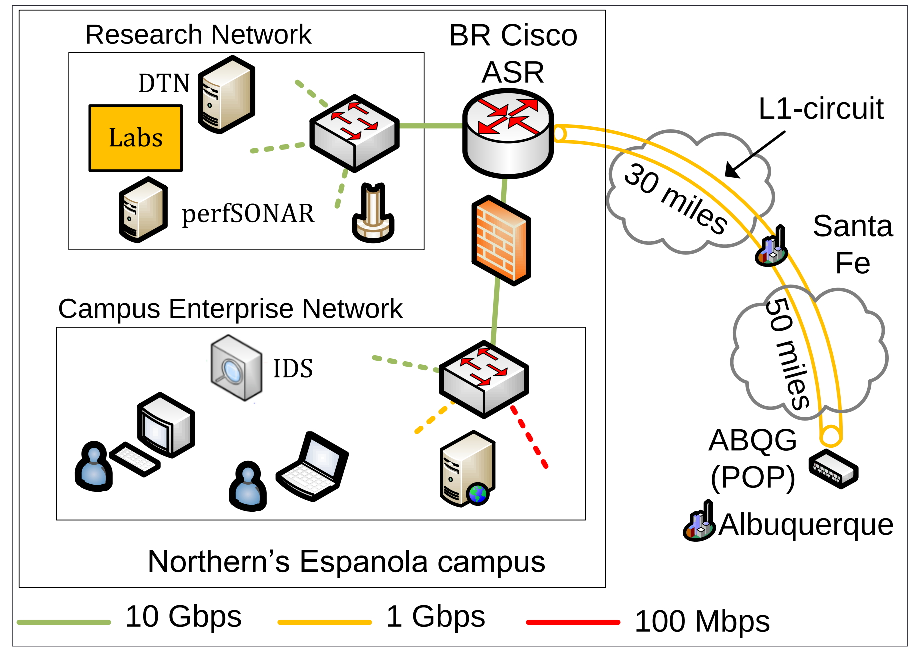
Northern's CC*DNI

Before



- ~50 msec RTT to ABQG, no traffic monitoring
- Border router Cisco 2800
- Multi-mode 1 Gbps fiber between some buildings only. Access switches at 100 Mbps

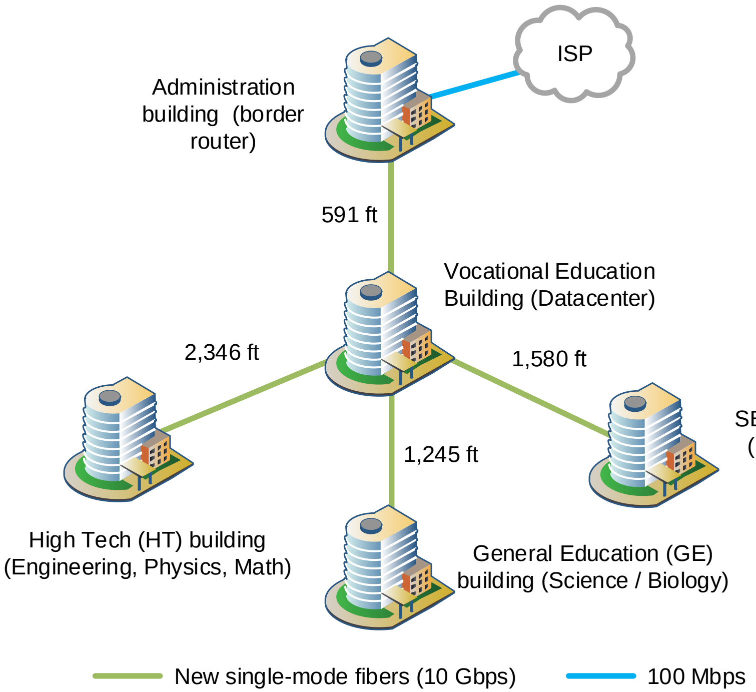
After



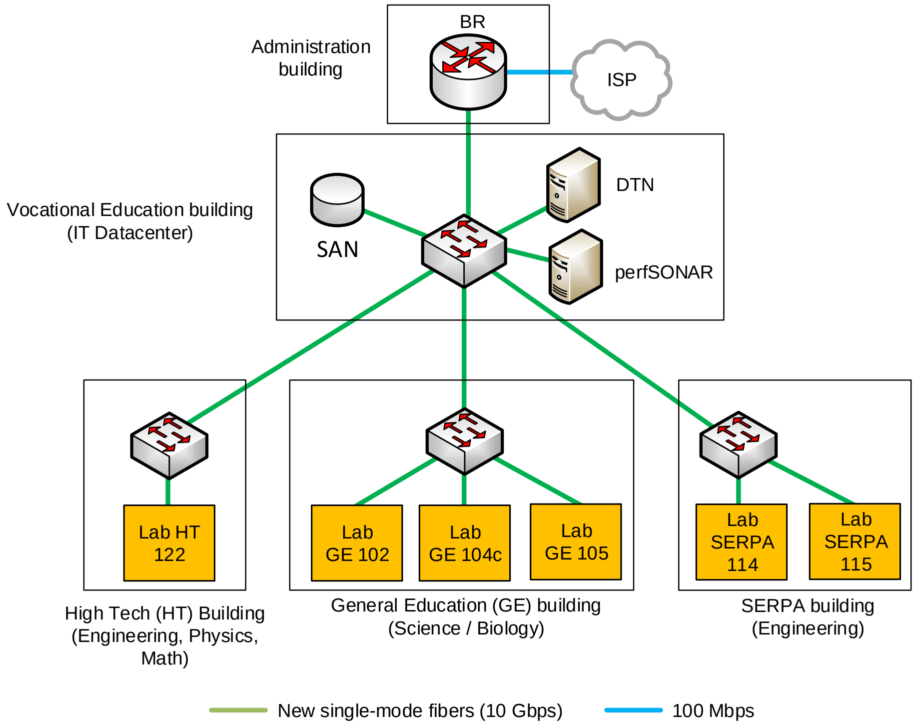
- P2P connection to ABQG; traffic monitoring
- Cisco ASR-family border router
- Single-mode 10 Gbps; research network
- Research activity; workforce development

Northern's CC*DNI

- Fiber deployment
- 1 Gbps connection to replace current 100 Mbps by 2018



Fiber deployment



Science DMZ

UNDERGRADUATE RESEARCH OPPORTUNITIES

Program

- Associate and Bachelor programs in Information Engineering Technology (IET)
- ABET Accreditor:

  "Engineering programs often focus on theory… while engineering technology programs usually focus on application and implementation"[1]

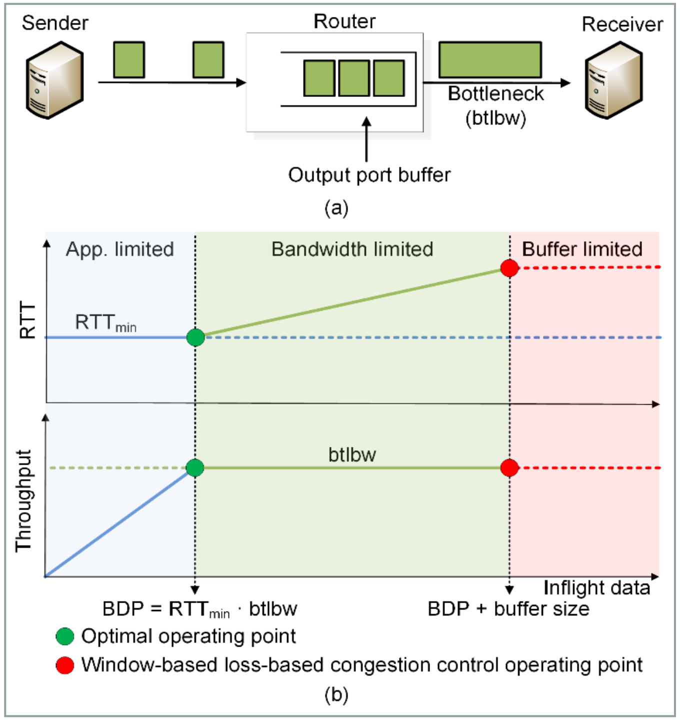
- Excellent opportunity for undergraduate applied research!

1. http://www.abet.org/accreditation/new-to-accreditation/engineering-vs-engineering-technology/

TCP BBR VS WINDOW-BASED LOSS-BASED CONGESTION CONTROL: EFFECT OF MSS AND PARALLEL STREAMS ON BIG FLOWS

BBR Brief Overview

- TCP BBR has been recently proposed as a congestion control algorithm (2016/17)[1]
- BBR represents a disruption from the window-based loss-based congestion control used during the last decades[2]
- BBR uses 'pacing' to try to match the bottleneck rate



(a) A viewpoint of a TCP connection. (b) Throughput and RTT, as a function of inflight data[1].

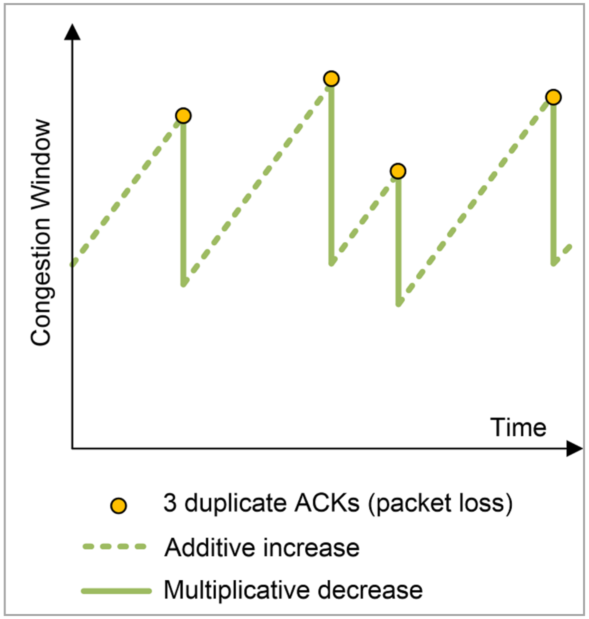
1. N. Cardwell, Y. Cheng, C. Gunn, S. Yeganeh, V. Jacobson, "Bbr: congestion-based congestion control," *Communications of the ACM*, vol 60, no. 2, pp. 58-66, Feb. 2017.
2. https://www.thequilt.net/wp-content/uploads/BBR-TCP-Opportunities.pdf

MSS and Parallel Streams

- Two of the main features impacting big flows
  - Maximum segment size (MSS)
  - The use of parallel streams

MSS

- Large MSS produces a faster recovery after a packet loss



$$\text{TCP throughput} = \frac{c \cdot MSS}{RTT \cdot \sqrt{p}}$$

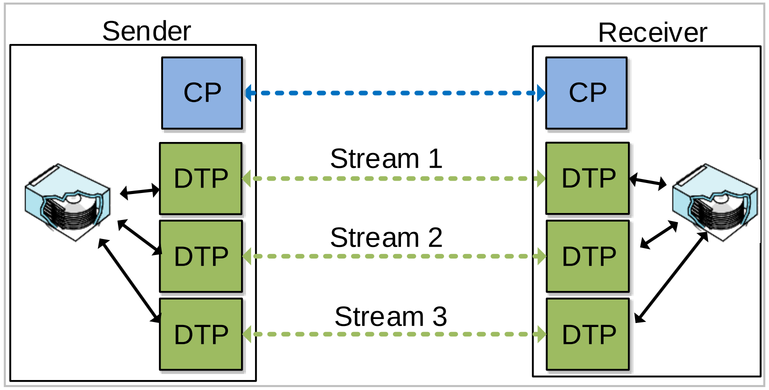
MSS: maximum segment size
RTT: round-trip time
p: loss rate
c: constant

Note: the above equation does not apply to BBR

M. Mathis, J. Semke, J. Mahdavi, T. Ott, "The macroscopic behavior of the tcp congestion avoidance algorithm," *ACM Computer Communication Review*, vol. 27, no 3, pp. 67-82, Jul. 1997.

Parallel Streams

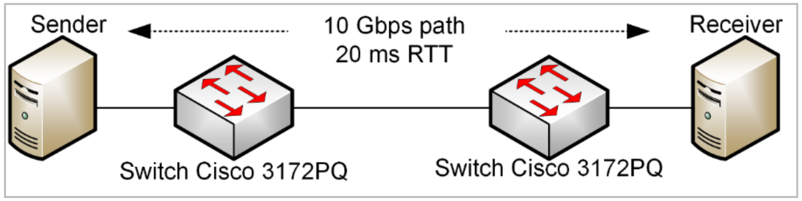
- Opening parallel connections essentially creates a large virtual MSS on the aggregate connection

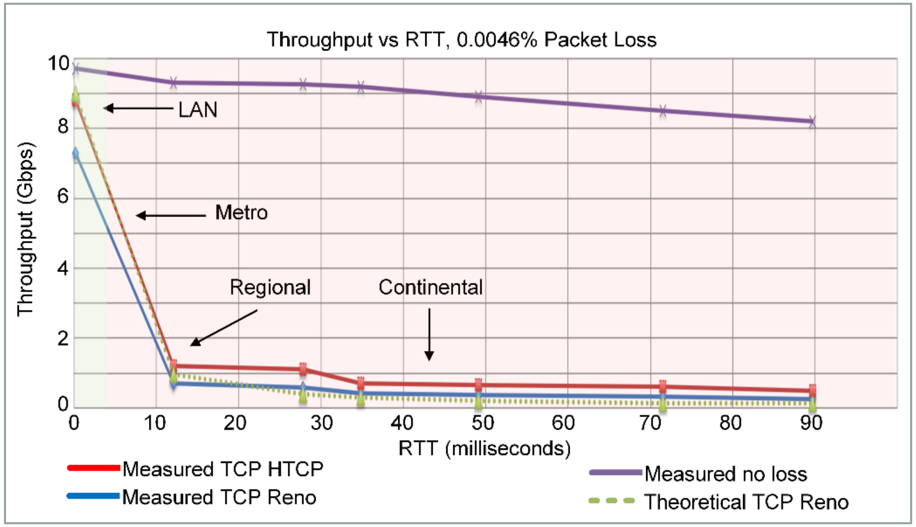


CP: Control process
DTP: Data transfer process

Scenario

- Sender/receiver connected by a 10 Gbps path, 20 ms RTT, running CentOS 7
- Memory-to-memory tests using iPerf3
- NeTem used to adjust loss rate
- At 20 ms RTT, throughput already collapses when subject to a small loss rate

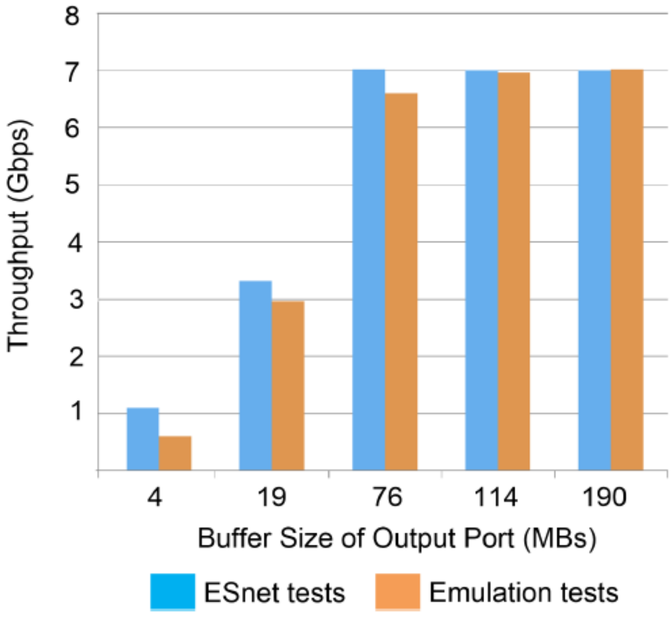




E. Dart, L. Rotman, B. Tierney, M. Hester, J. Zurawski, "The science dmz: a network design pattern for data-intensive science," *International Conference on High Performance Computing, Networking, Storage and Analysis*, Nov. 2013.

Emulation vs Real Networks

- Throughput of two TCP flows
- RTT: 70 milliseconds; 10 Gbps for all links; bandwidth-delay product: 83.4 MBs



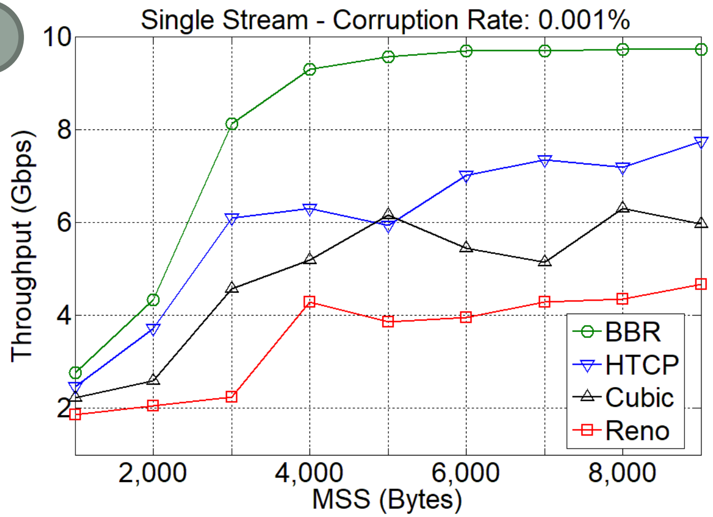
M. Smitasin, B. Tierney, "Evaluating network buffer size requirements," in *2015 Technology Exchange Workshop*, Oct. 2015. [Online]. Available: https://meetings.internet2.edu/media/medialibrary/2015/10/05/20151005-smitasin-buffersize.pdf

Scenario

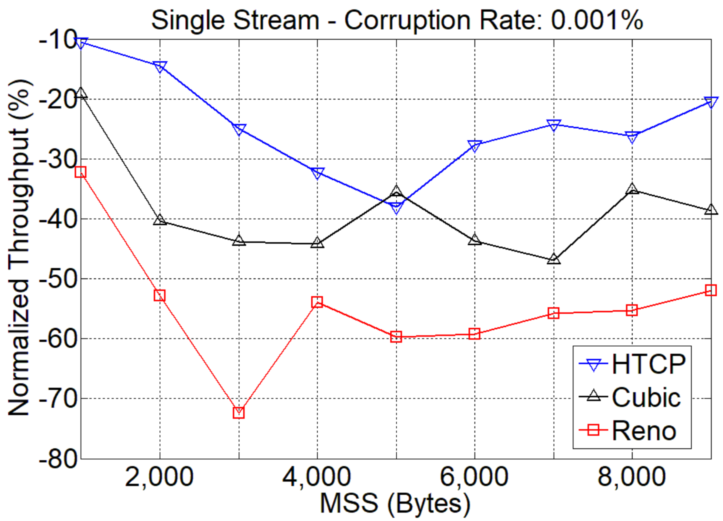
- Each experiment lasted 70 seconds (first 10 seconds were not taken into account)
- For each test condition, ten experiments were conducted and the average throughput was computed

Results

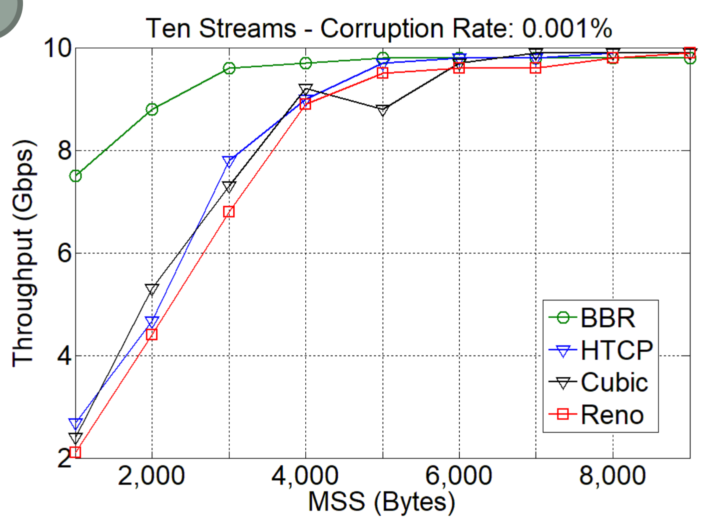
1



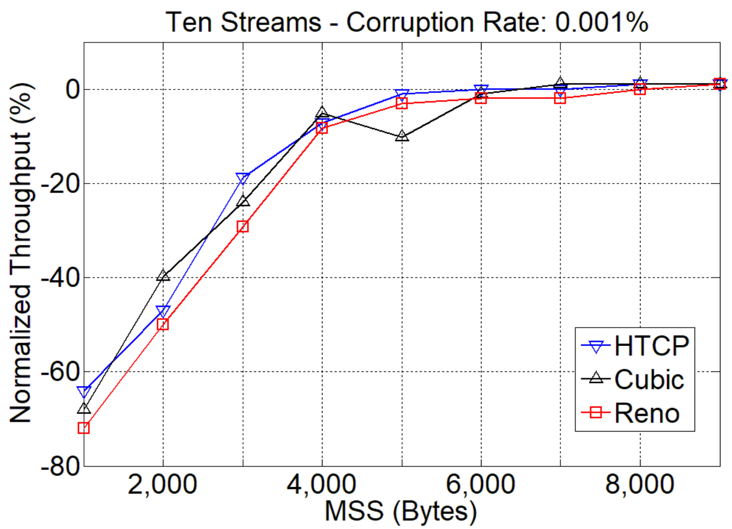
2



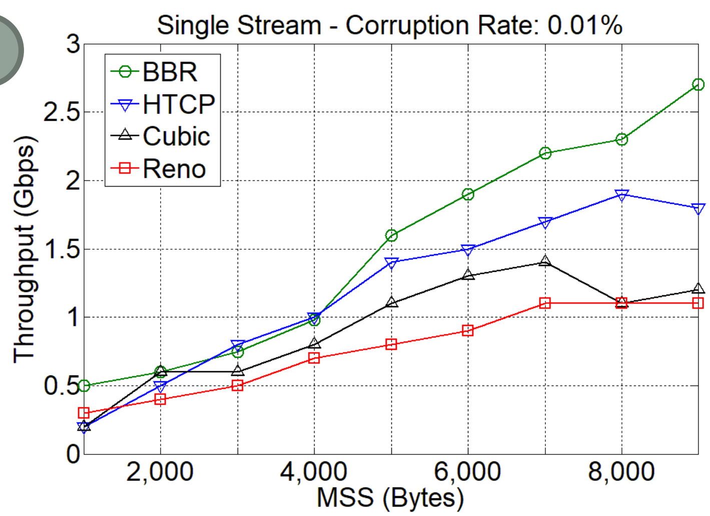
3

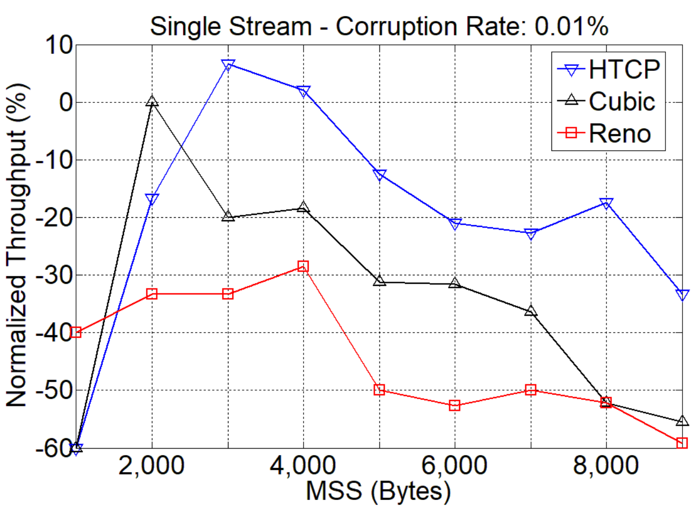


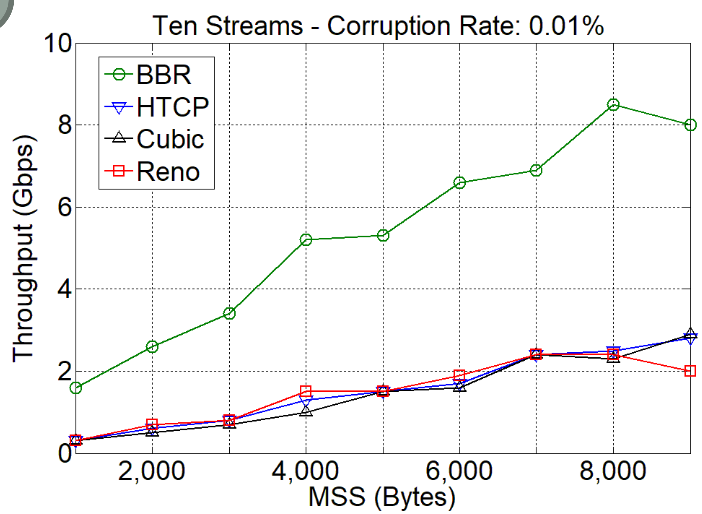
4

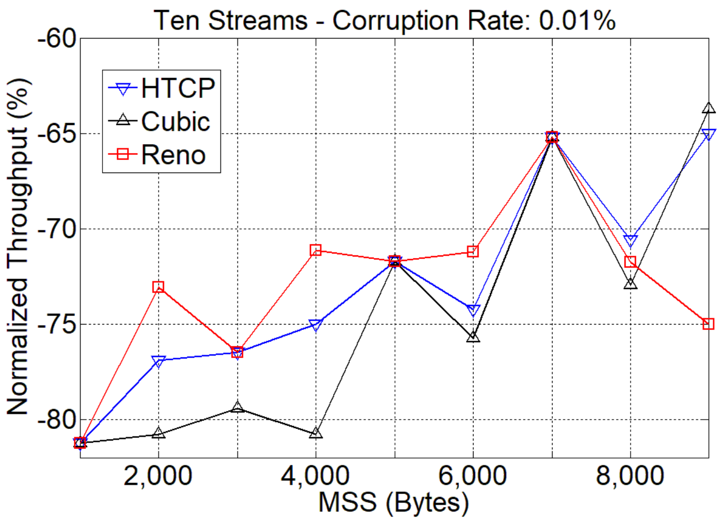


Results

1


2


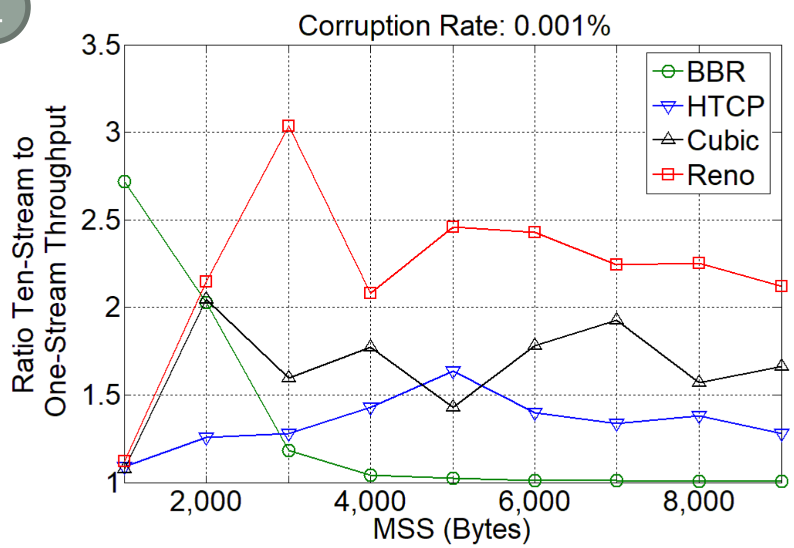
3


4


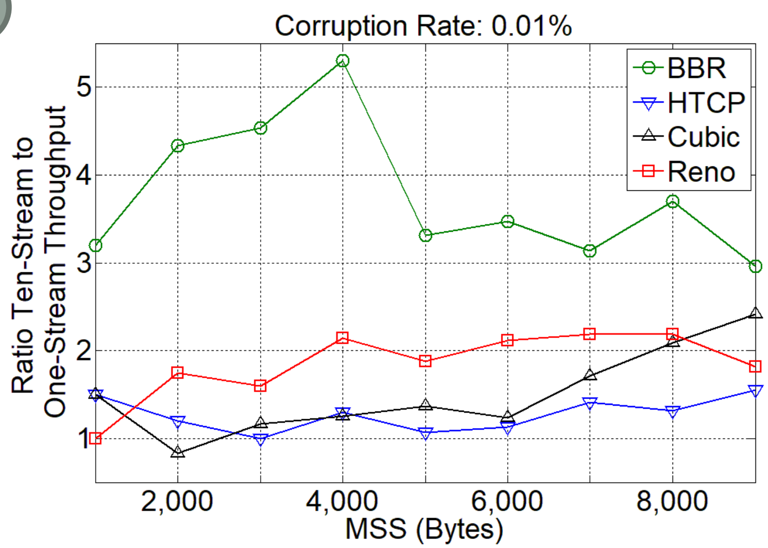
Results

- When not limited by network bandwidth, parallel streams improved BBR's throughput by more than a factor of 3
- The improvement factor for loss-based CC is lower
- When parallel streams are used, the performance of HTCP, Cubic, and Reno are similar

1



2



TRAFFIC CHARACTERIZATION USING NETFLOW

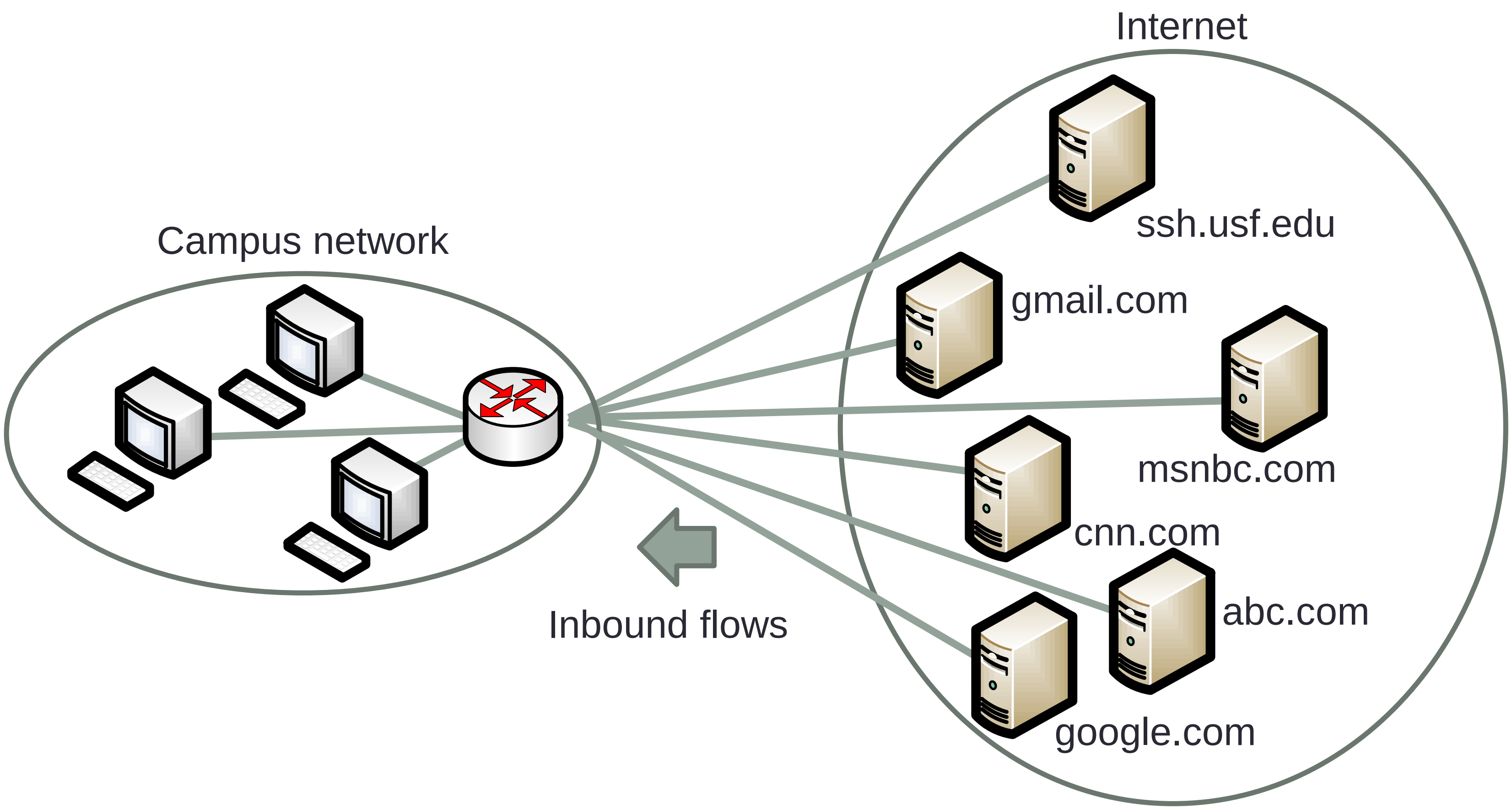
Motivation

- Border router acquired with the NSF CC*DNI grant has Netflow capability
- Flow statistics are available
- Flow-based IDS is more scalable than payload-based IDS[1]
- Goal: characterize normal flow behavior

1. R. Hofstede, P. Celeda, B. Trammell, I. Drago, R. Sadre, A. Sperotto, A. Pras, “Flow monitoring explained: from packet capture to data analysis with netFlow and ipfix,” *IEEE Communications Surveys and Tutorials*, vol. 16, no. 4, 2014.

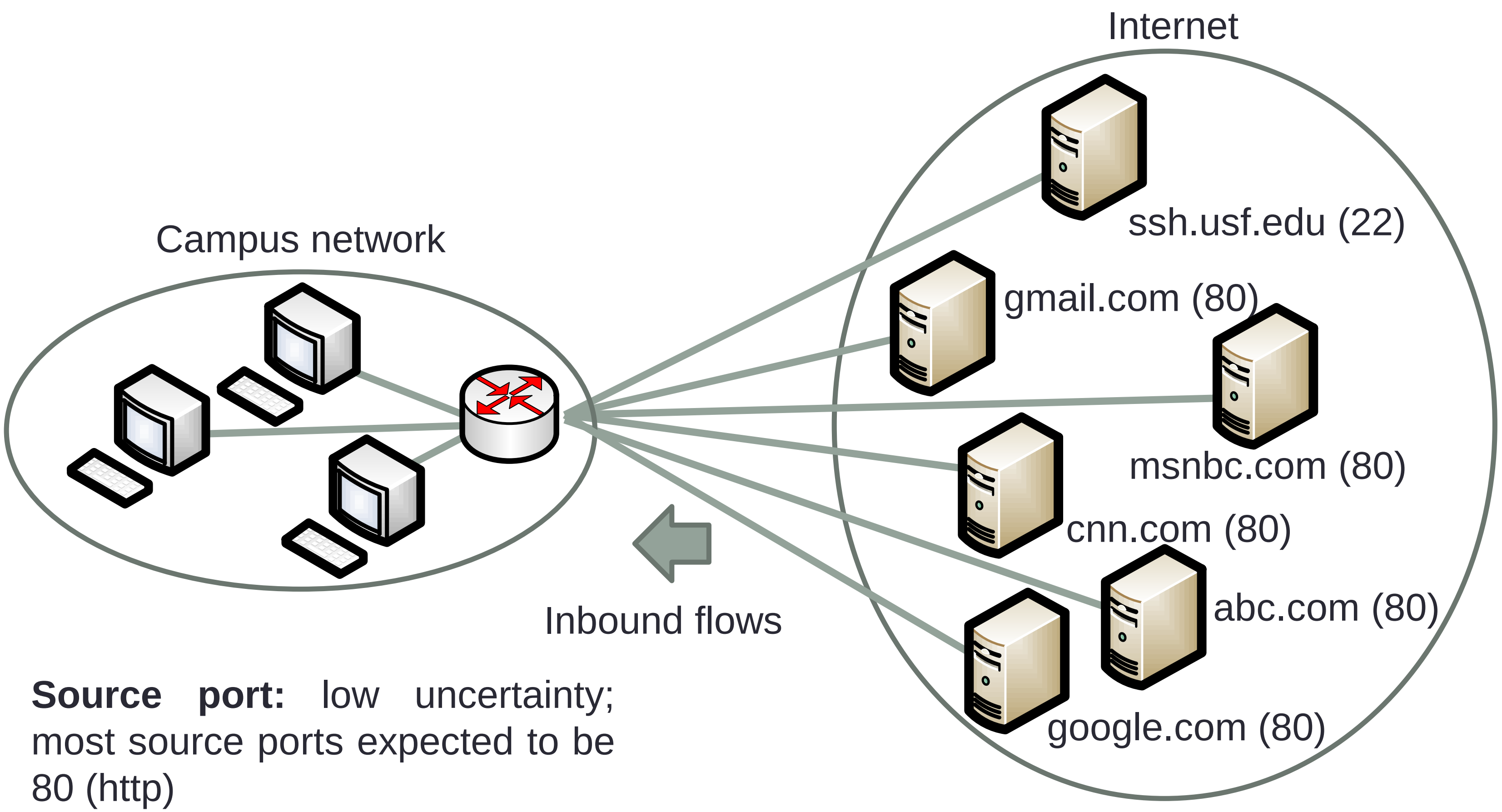
Motivation

- One approach for flow characterization is to measure the *randomness* or *uncertainty* of elements of a flow



Motivation

- One approach for flow characterization is to measure the *randomness* or *uncertainty* of elements of a flow



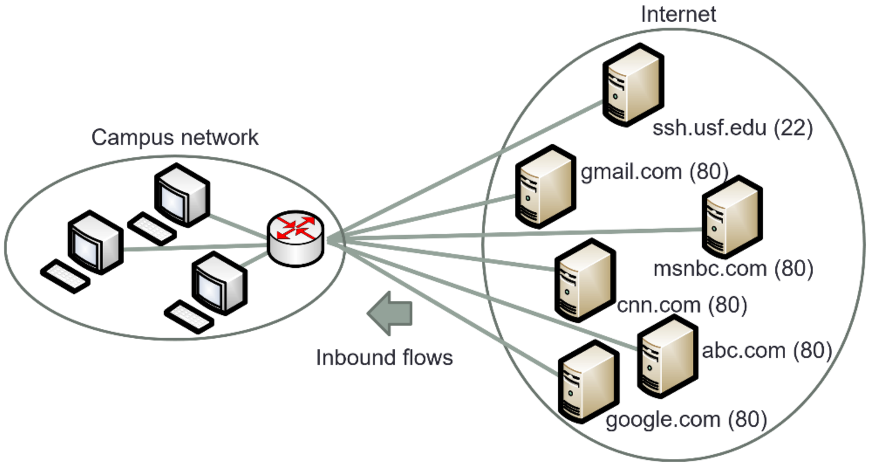
Source port: low uncertainty; most source ports expected to be 80 (http)

Motivation

- Entropy provides a measure of randomness or uncertainty
- For a variable X, entropy of X = $\sum_{x \in X} p_x \log_2 \left(\frac{1}{p_x} \right)$
- For the previous port example, let *X* be the variable indicating the source port

X : random variable indicating the source port

$$X = \begin{cases} 80 \text{ with probability } p_1 = \frac{5}{6} \\ 22 \text{ with probability } p_2 = \frac{1}{6} \end{cases}$$

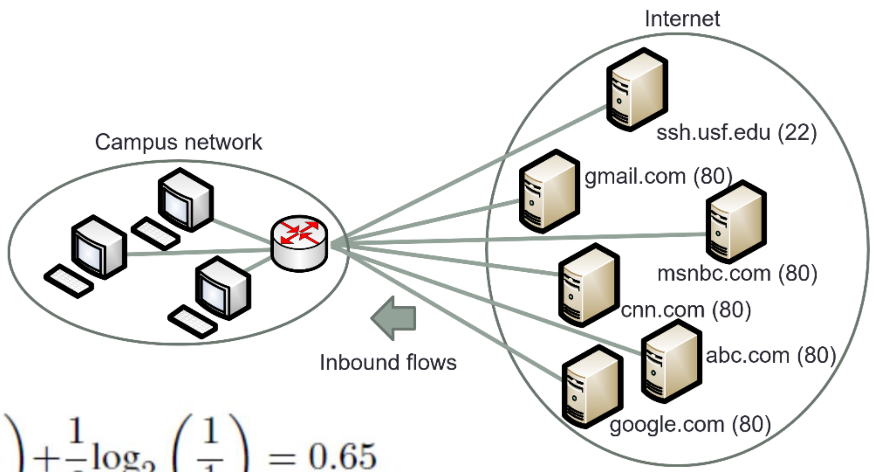


Motivation

- Entropy provides a measure of randomness or uncertainty
- For a variable X, entropy of X = $\sum_{x \in X} p_x \log_2 \left(\frac{1}{p_x} \right)$
- For the previous port example, let *X* be the variable indicating the source port

X : random variable indicating the source port

$$X = \begin{cases} 80 \text{ with probability } p_1 = \frac{5}{6} \\ 22 \text{ with probability } p_2 = \frac{1}{6} \end{cases}$$



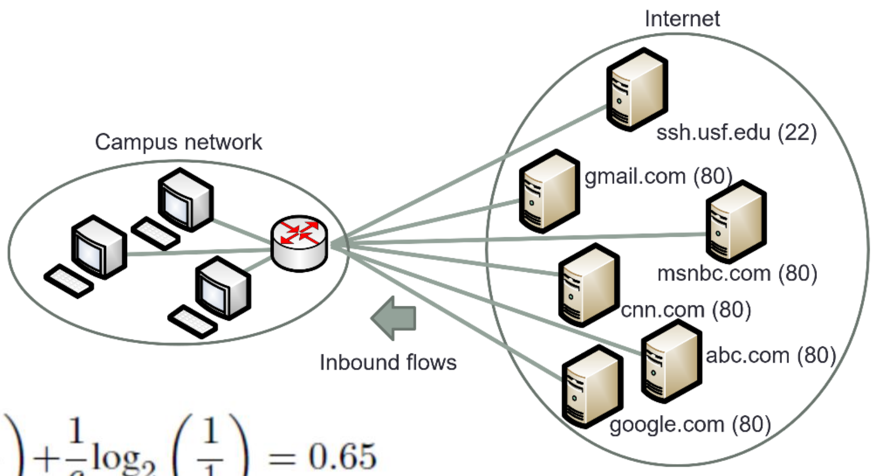
$$\text{Entropy Source Port} = \sum_{i=1}^{2} p_i \log_2 \left(\frac{1}{p_i} \right) = \frac{5}{6} \log_2 \left(\frac{1}{\frac{5}{6}} \right) + \frac{1}{6} \log_2 \left(\frac{1}{\frac{1}{6}} \right) = 0.65$$

Motivation

- Entropy provides a measure of randomness or uncertainty
- For a variable X, entropy of X = $\sum_{x \in X} p_x \log_2\left(\frac{1}{p_x}\right)$
- For the previous port example, let *X* be the variable indicating the source port

X : random variable indicating the source port

$$X = \begin{cases} 80 \text{ with probability } p_1 = \frac{5}{6} \\ 22 \text{ with probability } p_2 = \frac{1}{6} \end{cases}$$

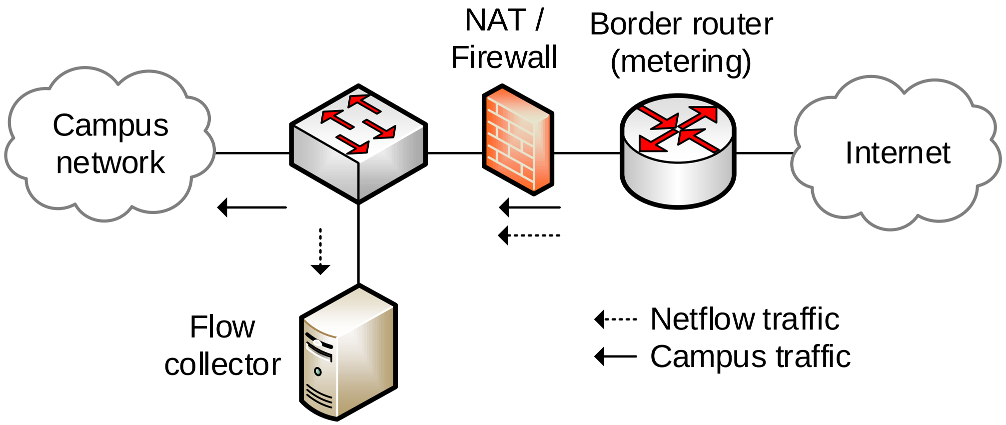


$$\text{Entropy Source Port} = \sum_{i=1}^{2} p_i \log_2\left(\frac{1}{p_i}\right) = \frac{5}{6}\log_2\left(\frac{1}{\frac{5}{6}}\right) + \frac{1}{6}\log_2\left(\frac{1}{\frac{1}{6}}\right) = 0.65$$

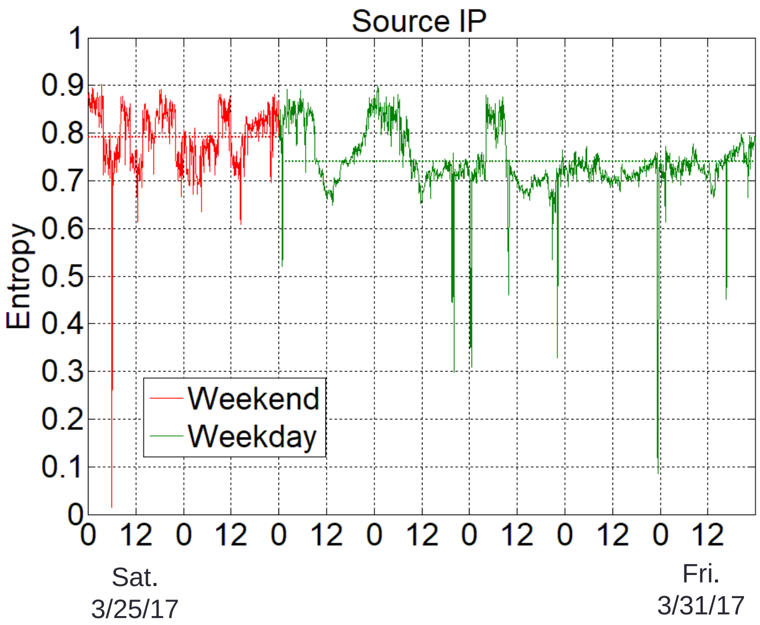
- 0 entropy -> no uncertainty… (e.g., all src ports are 80)
- 1 entropy -> random -> high uncertainty

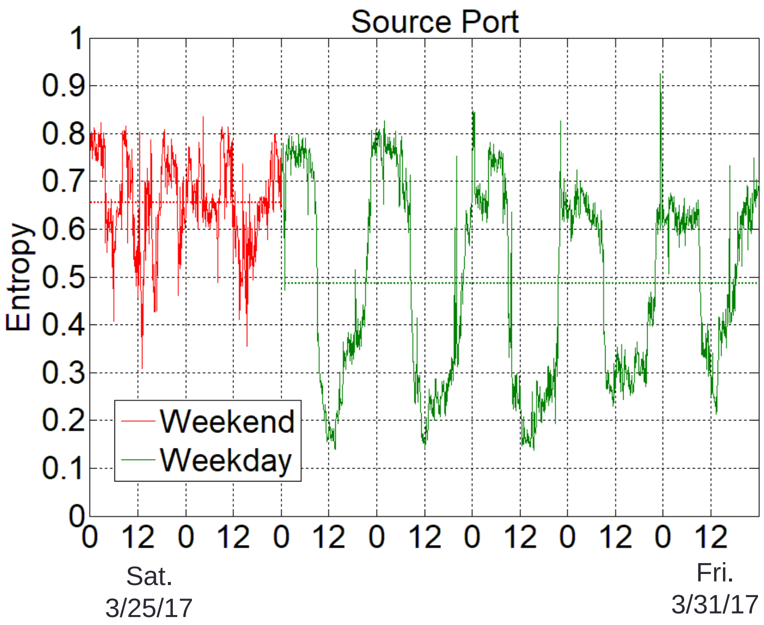
Scenario

- Small campus network ~12/15 buildings
- Inbound traffic is used as a reference (source IP address is in the Internet, destination IP address is on campus)
- The collector organizes flows in five-minute time slots



Results





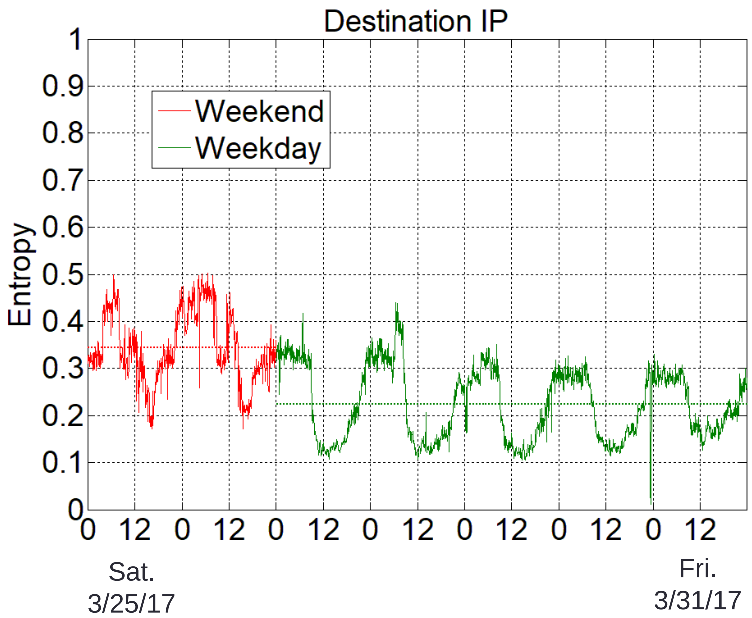
Source IP

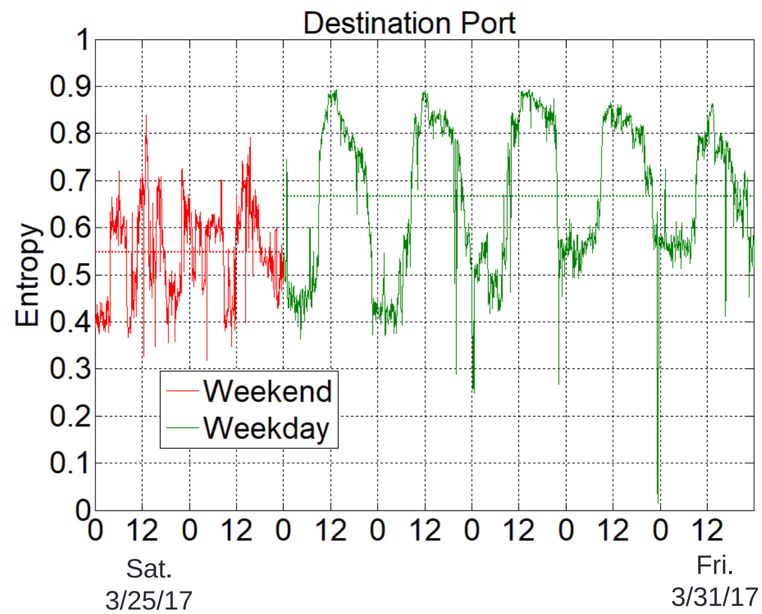
- In general, high entropy, 'many' source IP addresses
- Source IPs dispersed in the Internet
- Abnormal low entropy points
- Entropy near zero (no uncertainty of the source IP address, or 'very low' level (few source IP addresses dominate the distribution)

Source port

- Higher entropy during the night, weekends
- Low entropy during the day, noon
- Large volume of http flows when students are on campus (less uncertainty/entropy on source port)
- Abnormal high entropy points

Results





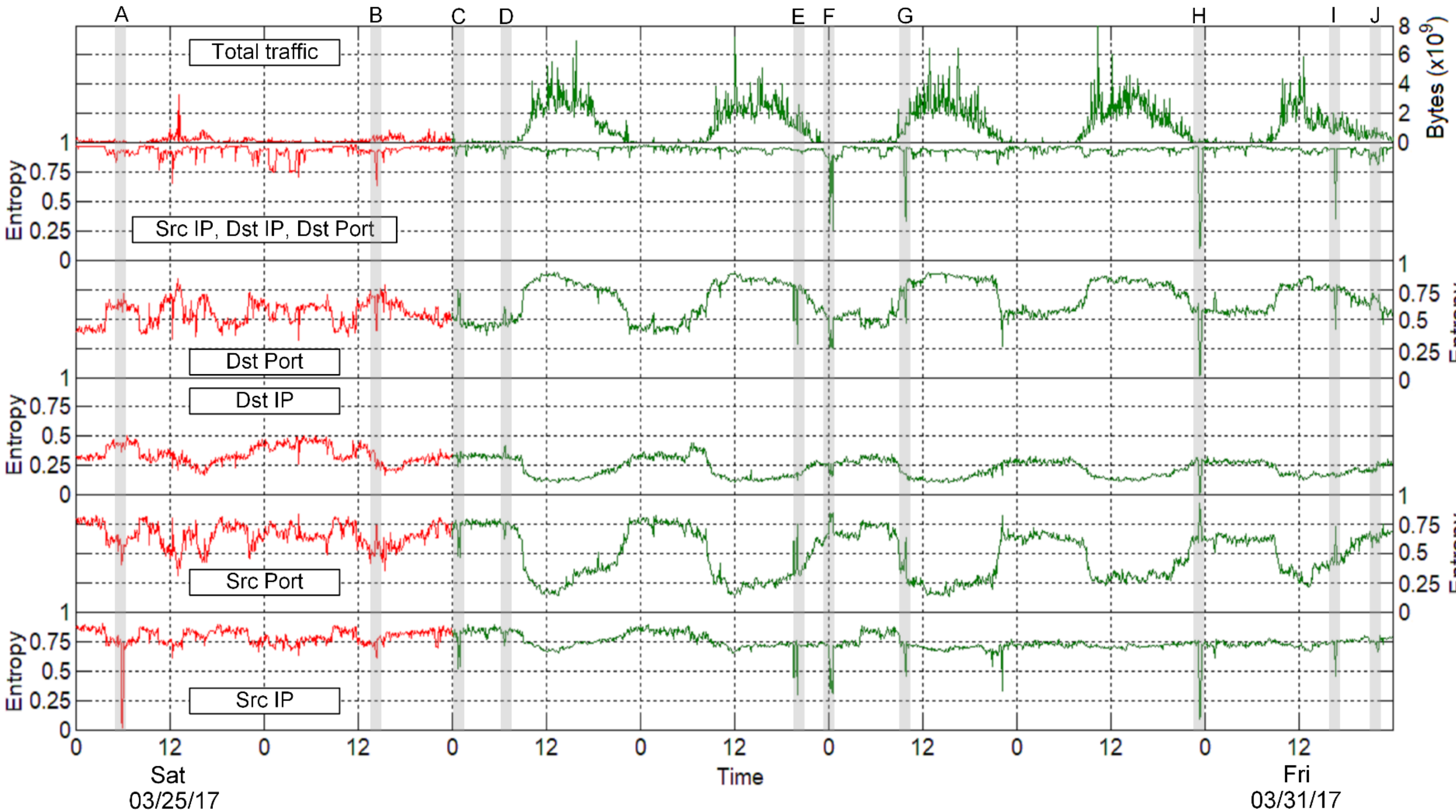
Destination IP

- In general, low entropy, 'few' IP addresses on campus
- Higher entropy on weekends and at night
- Lower entropy when students are on campus
- A handful of public IP addresses used for regular Internet connectivity (network address translation)

Destination port

- Lower entropy at night
- High entropy (close to uniform distribution) at noon
- Dynamic ports used by browsers when students connect to the Internet
- Abnormal low entropy points

Results



- Anomalies are detected by correlating different features
- E.g., event I: low destination port's entropy, high source port's entropy, low source IP's entropy

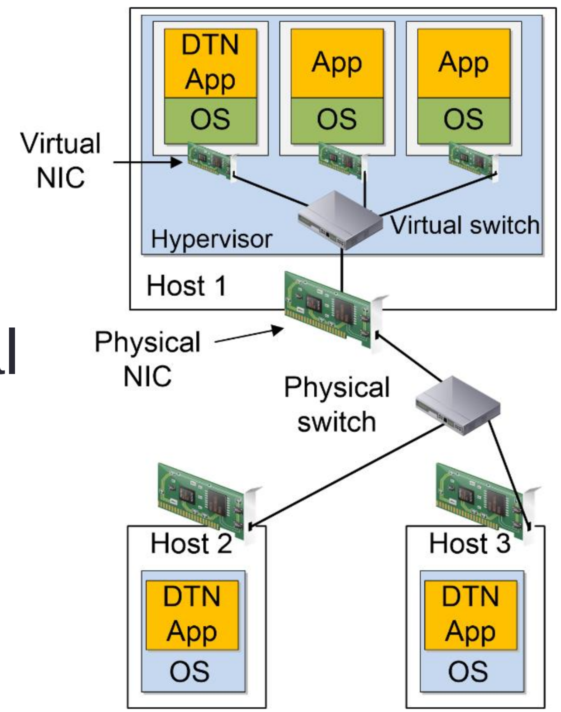
PERFORMANCE EVALUATION OF VIRTUAL DTNS

Motivation

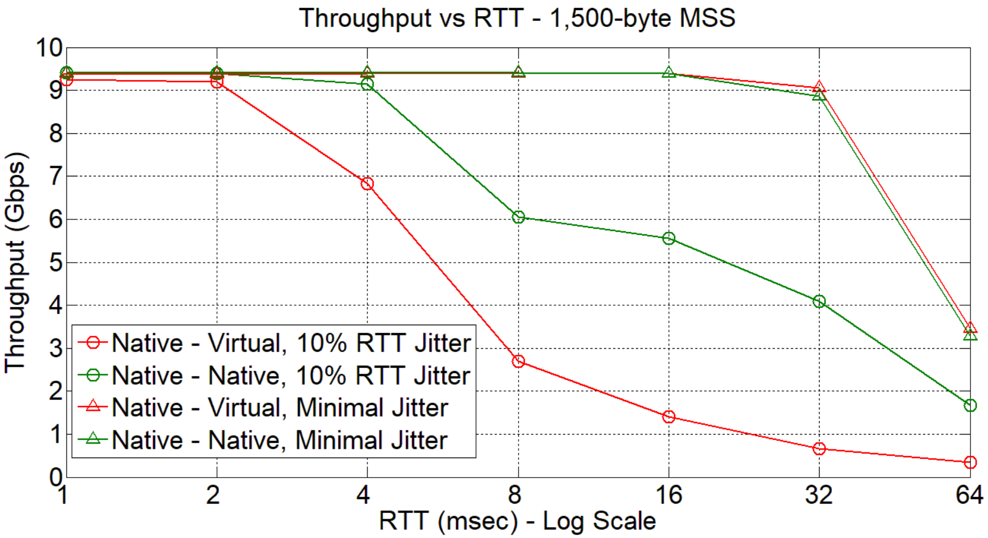
- vDTNs are attractive for some small institutions
- VMware is well known technology
- VMs are easy to deploy on demand

Scenario

- Two scenarios considered
  - From host 2 DTN to the virtual DTN located in host 1 (virtual environment using VMware's ESXi hypervisor)
  - From host 2 DTN to a host 3 DTN (native environment)
- The path capacity was 10 Gbps
- vDTN used VMXNET3 vNIC
- Memory-to-memory tests w/ iPerf3
- WAN emulation using NeTem
- Limited buffer capability by the physical switch (~8 MB)



Scenario

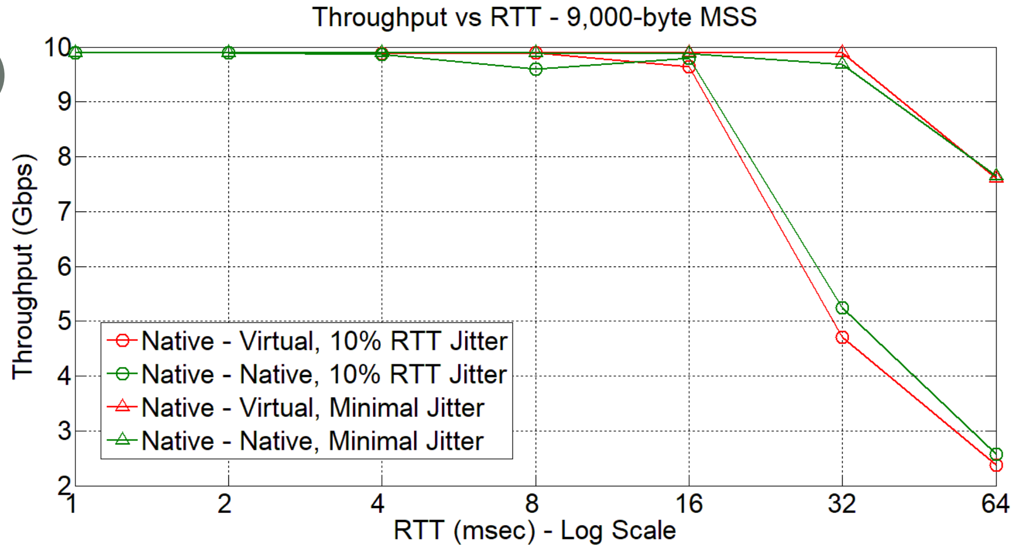


1

- For 8 MB buffer and 10 Gbps bandwidth, the critical RTT using the BDP relation is ~6.7 ms
- Small MSS means more segment processing
- Jitter leads to retransmissions, additional processing

2

- Comparable performance when MSS is large



Conclusion

- The NSF CC*DNI project had an impact well above expected
- Intra-campus connectivity improved from 100 Mbps to 10 Gbps
- Connectivity to Internet to improve from 100 Mbps to 1 Gbps
- Impact on science research, biology in particular
- For IT students, plenty of research and hands-on training opportunities