Design and Development of Network Monitoring Strategies in P4-enabled Programmable Switches

Damu Ding

Supervisor: Dr.Domenico Siracusa (FBK)
Co-supervisors: Dr.Marco Savi and Dr.Federico Pederzolli

University of Bologna, Bologna, Italy

NOMS 2022 dissertation digest
28th April, 2022

Network monitoring functionalities in Software-Defined Networks

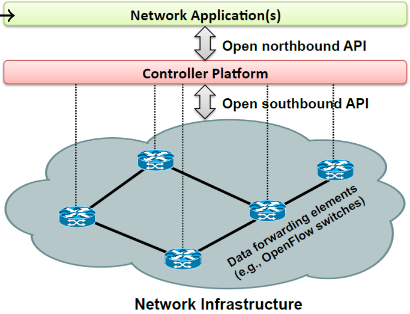
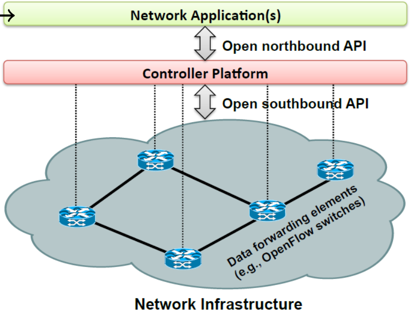


Figure source: Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76. and https://n0where.net/real-time-network-monitoring-cyberprobe

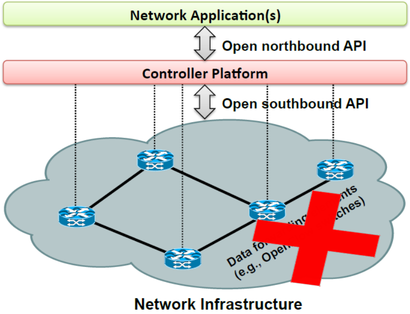
Network monitoring functionalities in Software-Defined Networks



1. **Significant communication overhead**
2. **The latency caused by interaction**
3. **Cannot perform monitoring at line-rate speed (Up to 100 Gbps)**

Figure source: Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76. and https://n0where.net/real-time-network-monitoring-cyberprobe

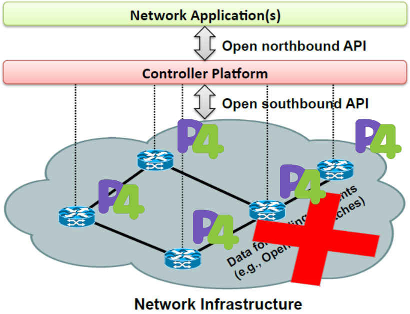
Network monitoring functionalities in Software-Defined Networks



Data plane programmable switches

Figure source: Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76. and https://n0where.net/real-time-network-monitoring-cyberprobe

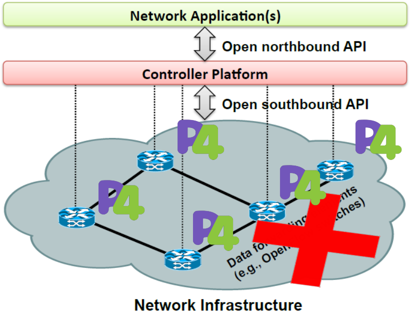
Network monitoring functionalities in Software-Defined Networks



Data plane programmable switches

Figure source: Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76. and https://n0where.net/real-time-network-monitoring-cyberprobe

Network monitoring functionalities in Software-Defined Networks



Data plane programmable switches

Figure source: Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76. and https://n0where.net/real-time-network-monitoring-cyberprobe

Network monitoring functionalities in Software-Defined Networks

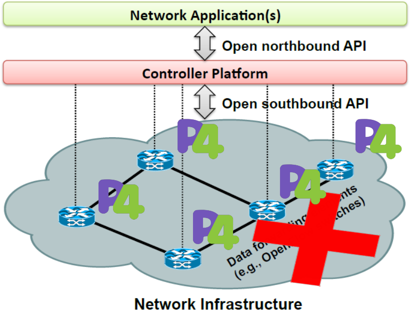
Heavy-hitter detection

Flow cardinality estimation

Network traffic entropy estimation

Traffic volume estimation

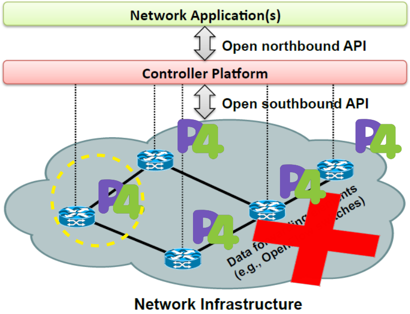
Volumetric DDoS detection



Data plane programmable switches

Figure source: Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76. and https://n0where.net/real-time-network-monitoring-cyberprobe

Network monitoring functionalities in Software-Defined Networks

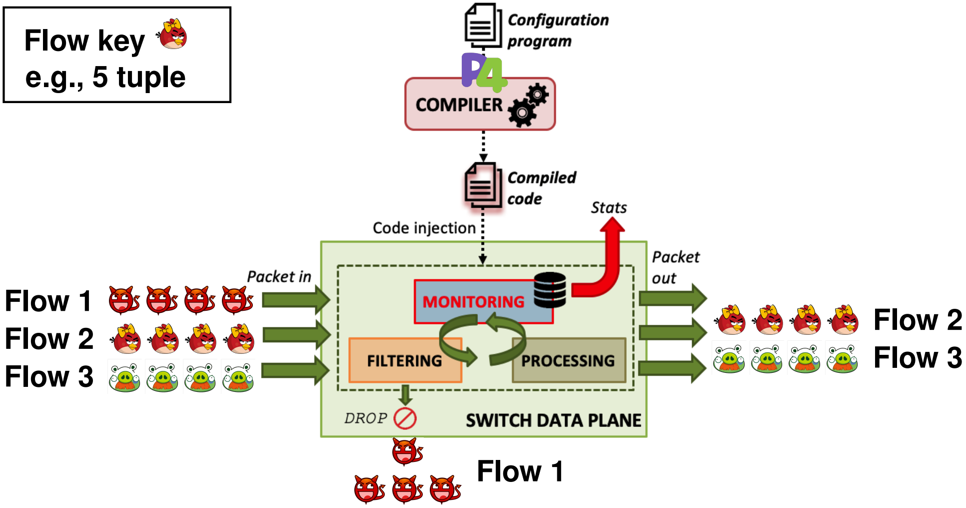


Heavy-hitter detection
Flow cardinality estimation
Network traffic entropy estimation
Traffic volume estimation
Volumetric DDoS detection

Data plane programmable switches

Figure source: Kreutz, Diego, et al. "Software-defined networking: A comprehensive survey." Proceedings of the IEEE 103.1 (2015): 14-76. and https://n0where.net/real-time-network-monitoring-cyberprobe

P4-enabled programmable data plane for monitoring



Challenges

32x 100Gbps QSFP ports

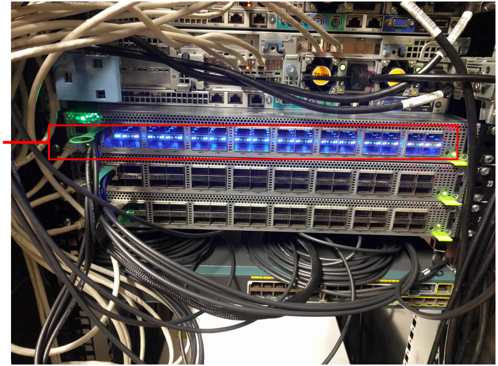


Figure: **Edgecore Wedge-100BF-32X switch equipped with Barefoot Tofino ASIC in FBK's lab**

Pros:

1. **Higher monitoring throughput**

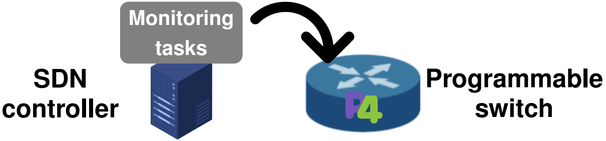
Cons:

1. **Limited hardware resources**
2. **Computational constraints**

Network monitoring tasks in literature cannot be directly offloaded to programmable switch data plane

Goal

Design and develop new strategies for specific monitoring tasks in P4-enabled programmable data planes considering the switch constraints

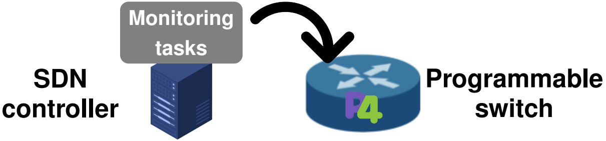


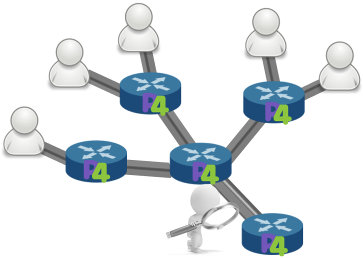
Motivation

Goal

Design and develop new strategies for specific monitoring tasks in P4-enabled programmable data planes considering the switch constraints

Focus on
ISP networks





- Minimize out-of-band actions
- High network performance

Outline

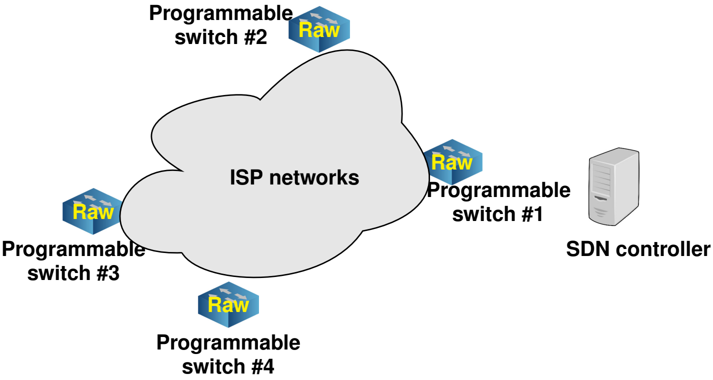
Part 1
Network-wide heavy-hitter detection

Part 2
Normalized network traffic entropy-based volumetric DDoS detection

Part 3
Per-flow cardinality-based volumetric DDoS detection

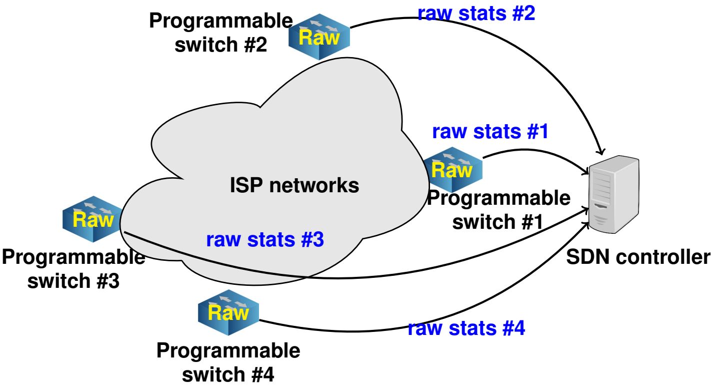
Network-wide heavy-hitter detection

Network-wide heavy-hitter detection



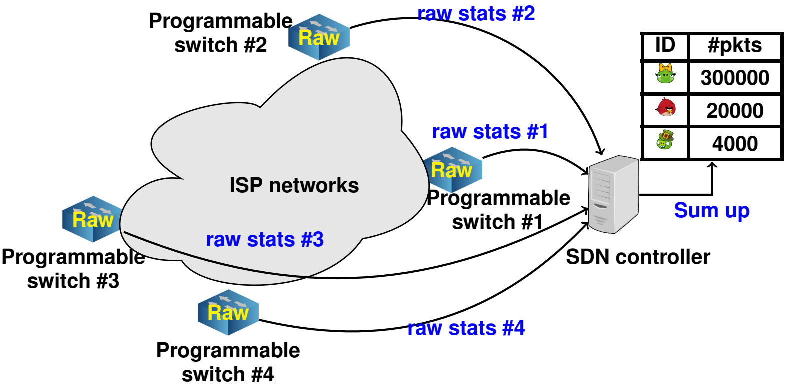
Harrison, Rob, et al. "Network-Wide Heavy Hitter Detection with Commodity Switches." Proceedings of the Symposium on SDN Research, 2018.

Network-wide heavy-hitter detection



Harrison, Rob, et al. "Network-Wide Heavy Hitter Detection with Commodity Switches." Proceedings of the Symposium on SDN Research, 2018.

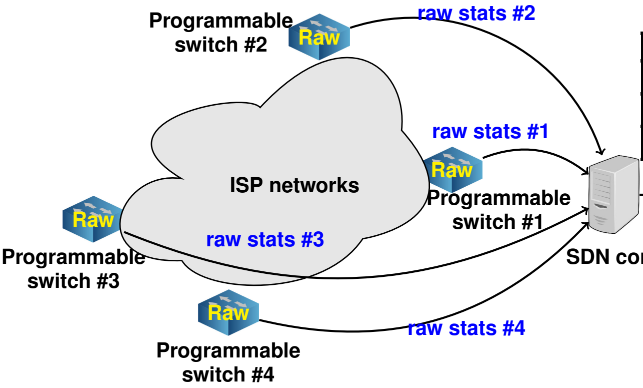
Network-wide heavy-hitter detection



Network-wide heavy-hitter detection: identifies the flows that contain more than a fraction of overall number of packets in the network

Harrison, Rob, et al. "Network-Wide Heavy Hitter Detection with Commodity Switches." Proceedings of the Symposium on SDN Research, 2018.

Network-wide heavy-hitter detection SOTA



ID	#pkts
	300000
	20000
	4000

Network-wide heavy-hitter detection: identifies the flows that contain more than a fraction of overall number of packets in the network

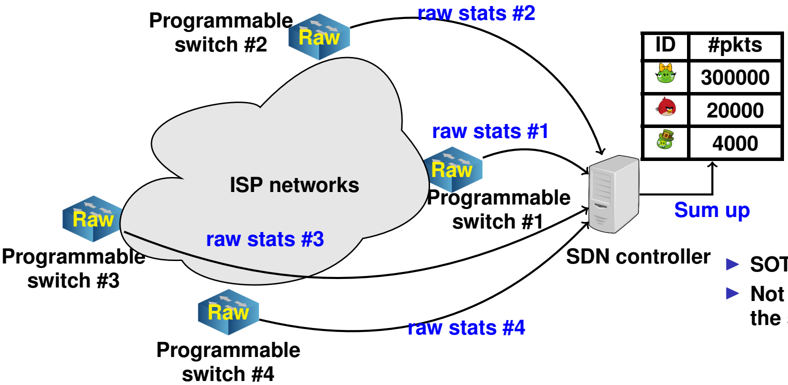
Limitations:

- SOTA stores raw data in the switch
- Not easy to define a threshold for the sum of flow packet counts

---

Harrison, Rob, et al. "Network-Wide Heavy Hitter Detection with Commodity Switches." Proceedings of the Symposium on SDN Research, 2018.

Network-wide heavy-hitter detection SOTA



Network-wide heavy-hitter detection: identifies the flows that contain more than a fraction of overall number of packets in the network

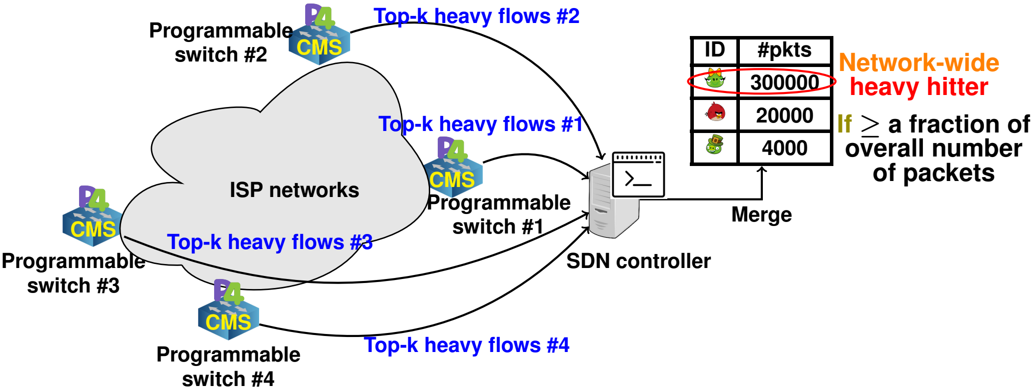
Limitations:

- SOTA stores raw data in the switch
- Not easy to define a threshold for the sum of flow packet counts

- **RQ1**: How to efficiently collect flow statistics in the switch?
- **RQ2**: How to accurately merge flow statistics in the controller?

Harrison, Rob, et al. "Network-Wide Heavy Hitter Detection with Commodity Switches." Proceedings of the Symposium on SDN Research, 2018.

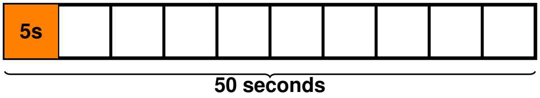
Network-wide heavy-hitter detection (NWHHD+)

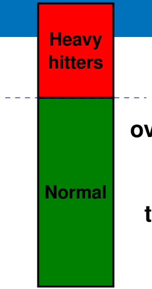


Cormode, Graham, and M. Muthukrishnan. "Count-Min Sketch." (2009): 511-516.
Damu Ding, Marco Savi, Gianni Antichi, and Domenico Siracusa. *An incrementally-deployable P4-enabled architecture for network-wide heavy-hitter detection.* IEEE Transactions on Network and Service Management (TNSM) 17.1 (2020): 75-88.

Evaluation settings

Normal traffic (CAIDA 2018)

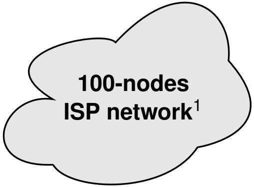




$\geq 0.05\% \cdot$ **overall number of packets in the time interval**

$$Recall = \frac{TP}{TP+TN} \qquad Precision = \frac{TP}{TP+FP}$$

$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$



- Each node has the same probability to be an ingress or egress point
- Each packet is forwarded from the ingress point to the egress point following the shortest path

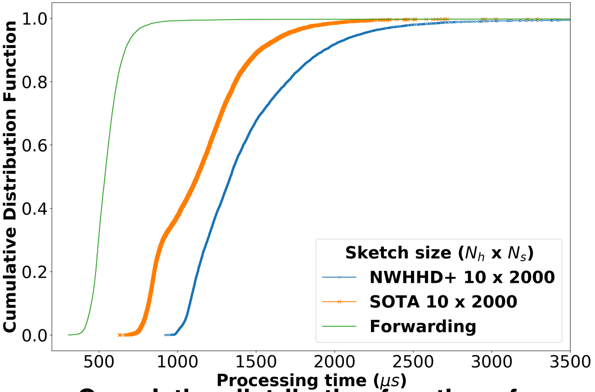
[1] https://sites.uclouvain.be/defo

$$TP = Count^{detected/true}_{Heavyhitter}, FP = Count^{detected/false}_{Heavyhitter}, TN = Count^{undetected/true}_{Heavyhitter}$$

Simulation and emulation results

Evaluation metrics	SOTA [2]	NWHHD+
F1 score	0.821	0.907
Communication overhead*	71877	60354
Occupied memory*	760042	60255

*Measurement units	ID	#pkts
		2000



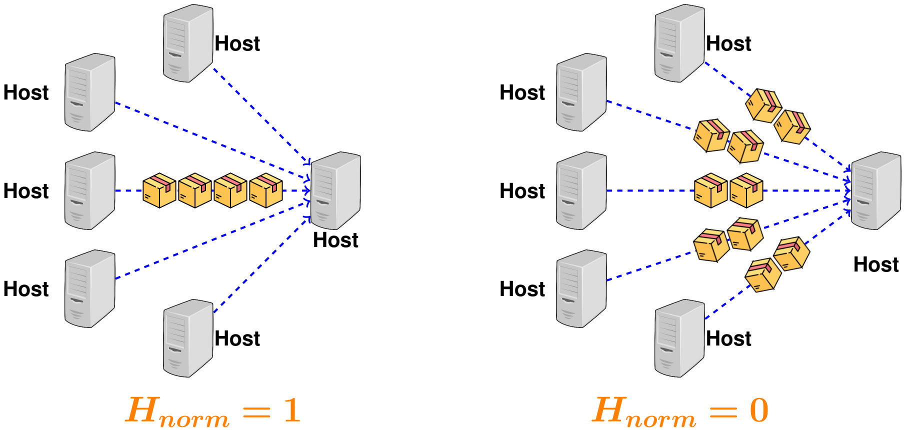
Cumulative distribution function of packet processing time in mininet (10000 packets)

[2] Harrison, Rob, et al. "Network-Wide Heavy Hitter Detection with Commodity Switches." Proceedings of the Symposium on SDN Research, 2018.

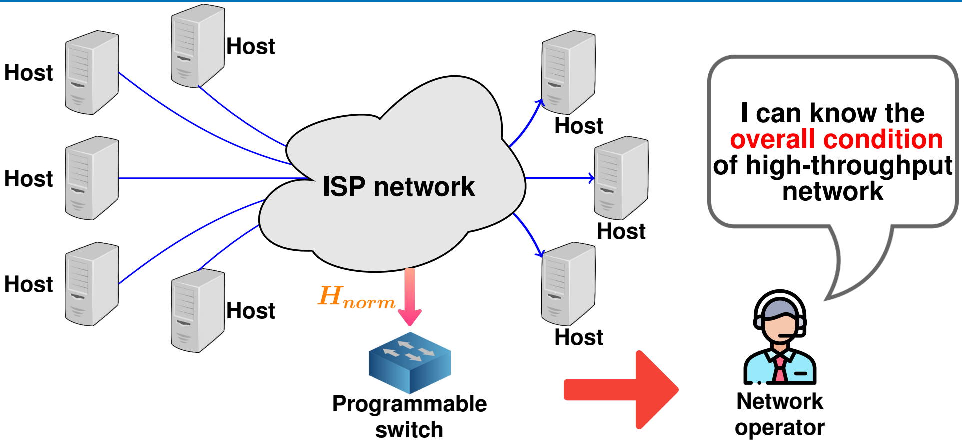
Normalized network traffic entropy-based DDoS detection

Normalized network traffic entropy

Normalized network traffic entropy H_{norm} indicates network traffic distribution



Normalized network traffic entropy in programmable switches

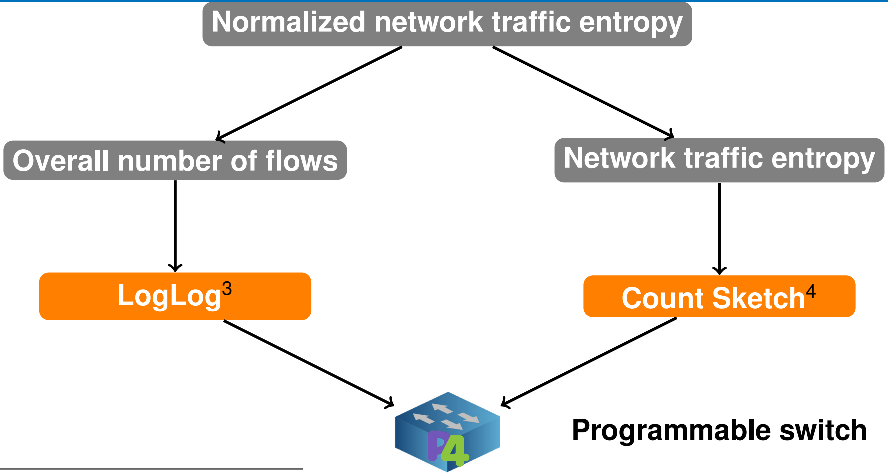


Normalized Shannon entropy

$$H_{norm} = \frac{-\sum_{i=1}^{n_{tot}} \frac{f_i}{S_{tot}} \log_2 \frac{f_i}{S_{tot}}}{\log_2 n_{tot}}$$

- f_i: Packet count of flow i
- S_{tot}: Overall number of packets ✓
- n_{tot}: Overall number of flows

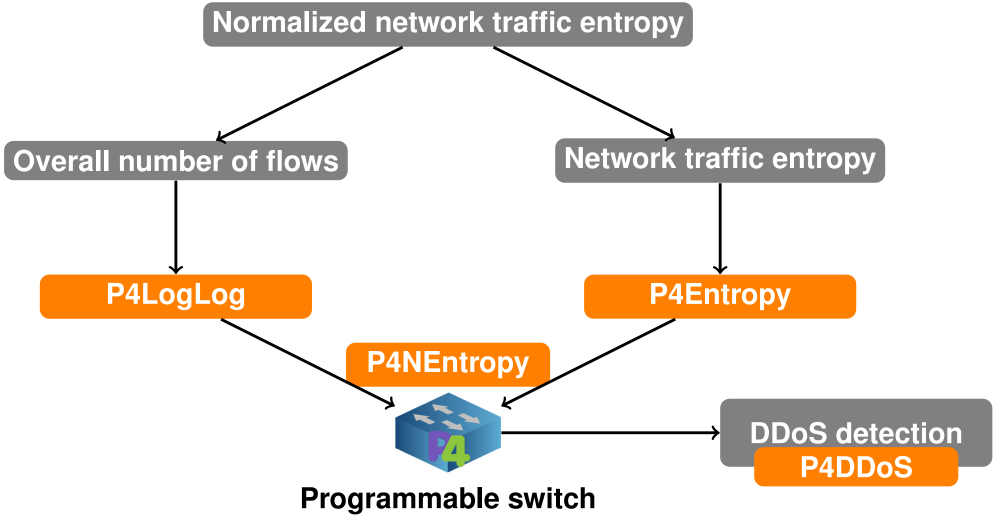
Normalized network traffic entropy



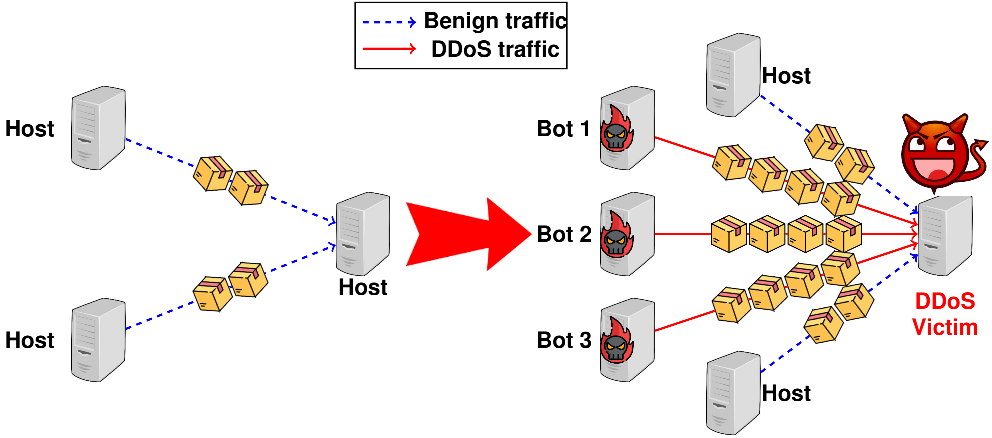
[3] Durand, Marianne et al. "Loglog counting of large cardinalities." European Symposium on Algorithms. Springer, Berlin, Heidelberg, 2003.

[4] M. Charikar et al, "Finding frequent items in data streams," in Springer International Colloquium on Automata,Languages, and Programming (ICALP), 2002.

Normalized network traffic entropy-based DDoS detection

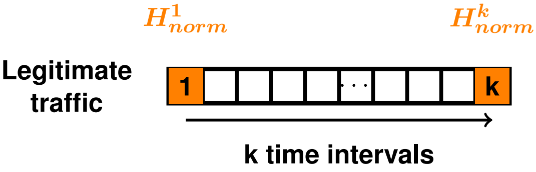


Property of volumetric DDoS attacks



H_{norm} **of destination IPs** ↓

Adaptive threshold for DDoS detection



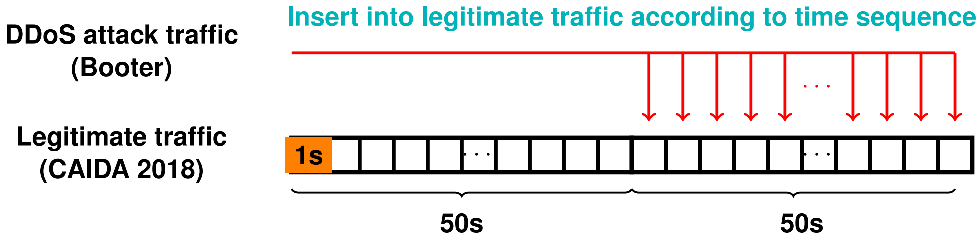
Exponentially weighted moving average (EWMA) of normalized entropy

$$EWMA_{norm}^1 = H_{norm}^1$$
$$EWMA_{norm}^2 = \alpha H_{norm}^1 + (1-\alpha) H_{norm}^2$$
$$\cdots$$
$$EWMA_{norm}^k = \alpha H_{norm}^{k-1} + (1-\alpha) H_{norm}^k$$

DDoS threshold

$$\lambda_{norm}^k = EWMA_{norm}^k - \epsilon$$

Evaluation settings

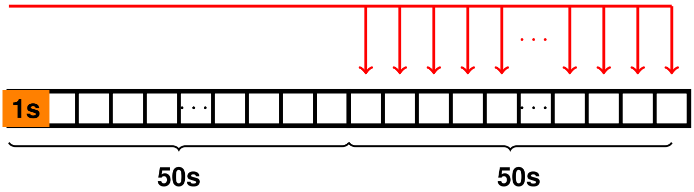


Evaluation settings

DDoS attack traffic
(Booter)

Legitimate traffic
(CAIDA 2018)

Insert into legitimate traffic according to time sequence

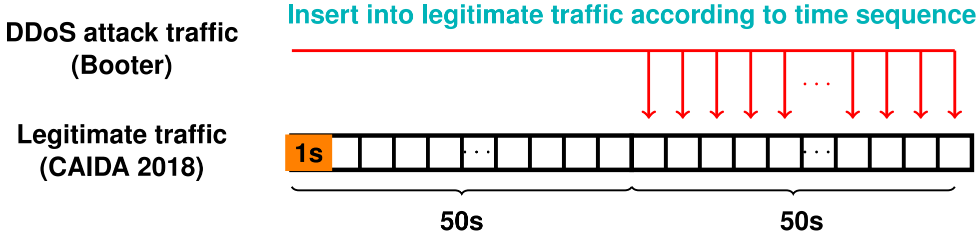


DDoS trace name	Packets per second	Attack source IPs
Booter 6	$\sim$ 90000	7379
Booter 7	$\sim$ 41000	6075
Booter 1	$\sim$ 96000	4486
Booter 4	$\sim$ 80000	2970

DNS-amplification
DDoS attacks

Booter is a class of on-demand services
that provide illegal support to launch DDoS attacks
targeting websites and networks.

Evaluation settings



	DDoS	NO DDoS
Alarm	True Positive (TP)	False Positive (FP)
NO alarm	False Negative (FN)	True Negative (TN)

$$D_{tp} = \frac{\#Time\ intervals[TP]}{\#Time\ intervals[TP+FN]} \qquad D_{fp} = \frac{\#Time\ intervals[FP]}{\#Time\ intervals[TN+FP]}$$

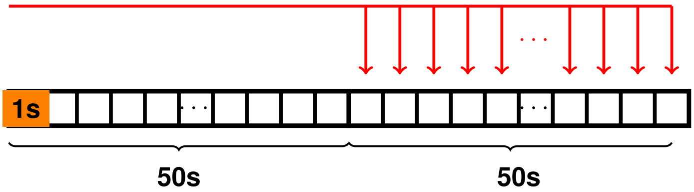
$$D_{acc} = \frac{\#Time\ intervals[TP+TN]}{\#Time\ intervals[TP+TN+FP+FN]}$$

Evaluation settings

DDoS attack traffic (Booter)

Legitimate traffic (CAIDA 2018)

Insert into legitimate traffic according to time sequence

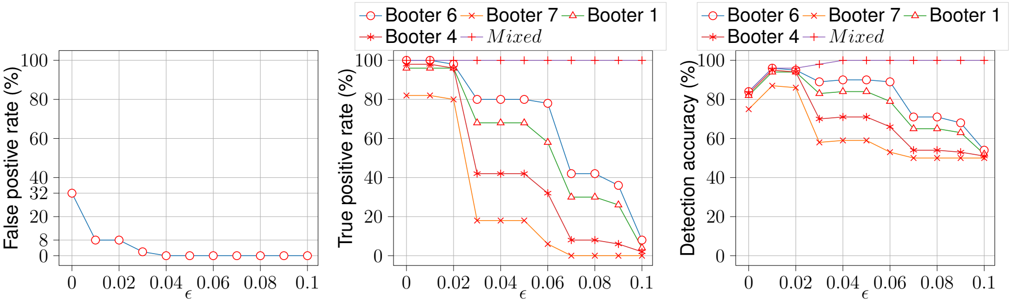


$$EWMA^{50}_{norm}(\alpha = 0.13)$$

$$\lambda^{51}_{norm} = EWMA^{50}_{norm} - \epsilon$$

Maximize DDoS detection accuracy

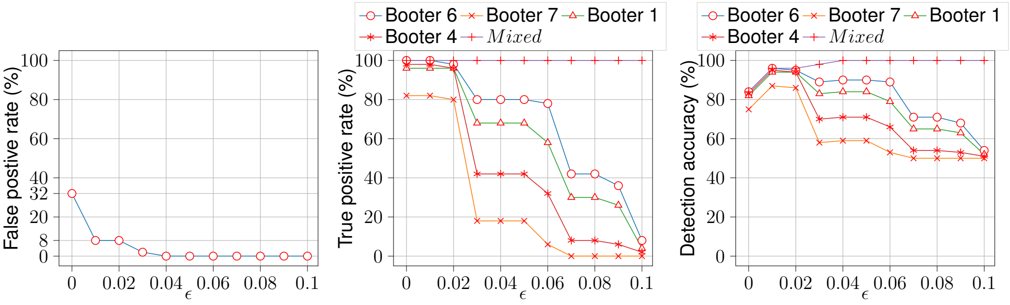
Configuring DDoS detection threshold



(a) Sensitivity to D_{fp} (b) Sensitivity to D_{tp} (c) Sensitivity to D_{acc}

- **Minimize** false positive rate D_{fp} ($\epsilon \in [0.01, 0.1]$)
- **Maximize** true positive rate D_{tp} ($\epsilon \in [0, 0.02]$)
- **Maximize** detection accuracy D_{acc} ($\epsilon \in [0.01, 0.02]$)

Configuring DDoS detection threshold

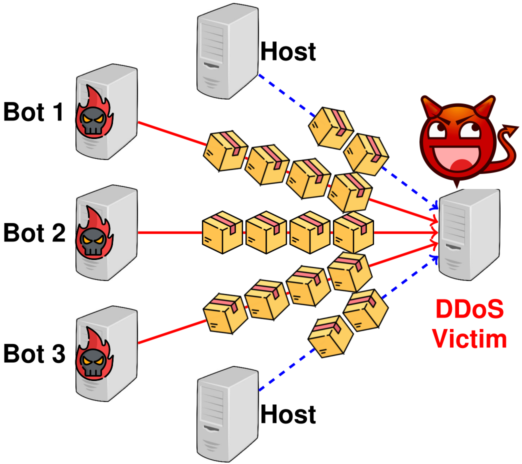


(a) Sensitivity to D_{fp} (b) Sensitivity to D_{tp} (c) Sensitivity to D_{acc}

- **Minimize** false positive rate D_{fp} ($\epsilon \in [0.01, 0.1]$)
- **Maximize** true positive rate D_{tp} ($\epsilon \in [0, 0.02]$)
- **Maximize** detection accuracy D_{acc} ($\epsilon \in [0.01, 0.02]$)

$$\epsilon = \mathbf{0.01}$$

State of the art (SOTA_DDoS)



Network traffic entropy
of source IPs H_{src} increases $\Uparrow$
OR
Network traffic entropy
of destination IPs H_{dst} decreases $\Downarrow$

Lapolli, Angelo Cardoso, Jonatas Adilson Marques, and Luciano Paschoal Gaspary. "Offloading real-time ddos attack detection to programmable data planes." 2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). IEEE, 2019.

Comparing to SOTA

Algorithm	False-positive rate D_{fp}	True-positive rate D_{tp} / Detection accuracy D_{acc}				
		Booter 6	Booter 7	Booter 1	Booter 4	Mixed
P4DDoS	8%	100% / 96%	82% / 87%	96% / 94%	98% / 95%	100% / 96%
SOTA_DDoS [5]	10%	100% / 95%	74% / 82%	100% / 95%	94% / 92%	100% / 95%

Booter name	PPS	Attack sources
Booter 6	$\sim$ 90000	7379
Booter 7	$\sim$ 41000	6075
Booter 1	$\sim$ 96000	4486
Booter 4	$\sim$ 80000	2970

[5] Lapolli, Angelo Cardoso, Jonatas Adilson Marques, and Luciano Paschoal Gaspary. "Offloading real-time ddos attack detection to programmable data planes." 2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). IEEE, 2019.

Comparing to SOTA

Algorithm	False-positive rate D_{fp}	True-positive rate D_{tp} / Detection accuracy D_{acc}				
		Booter 6	Booter 7	Booter 1	Booter 4	Mixed
P4DDoS	8%	100% / 96%	82% / 87%	96% / 94%	98% / 95%	100% / 96%
SOTA_DDoS [5]	10%	100% / 95%	74% / 82%	100% / 95%	94% / 92%	100% / 95%

And ...

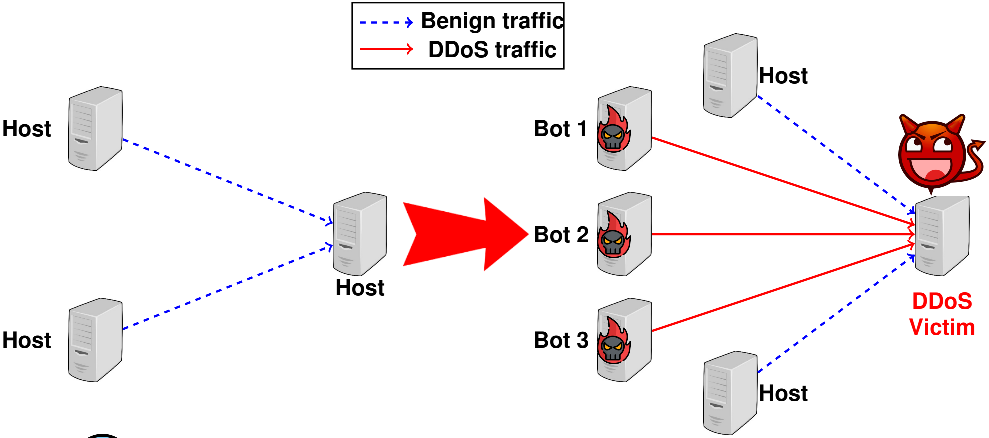
- No need to use power-hungry TCAM memory
  - Only relies on P4-supported operations
- Much simpler, i.e., lower implementation complexity
  - Only relies on normalized entropy of destination IPs
- Robust to the flow fluctuations in different time intervals
  - Normalized entropy instead of only entropy

Booter name	PPS	Attack sources
Booter 6	~ 90000	7379
Booter 7	~ 41000	6075
Booter 1	~ 96000	4486
Booter 4	~ 80000	2970

[5] Lapolli, Angelo Cardoso, Jonatas Adilson Marques, and Luciano Paschoal Gaspary. "Offloading real-time ddos attack detection to programmable data planes." 2019 IFIP/IEEE Symposium on Integrated Network and Service Management (IM). IEEE, 2019.

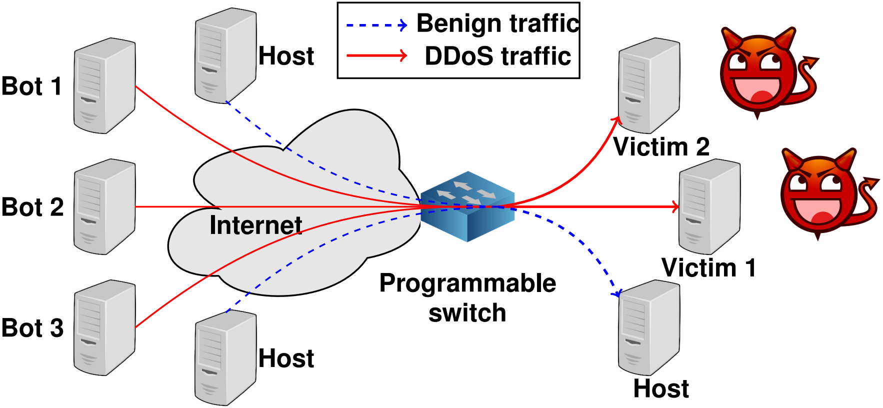
Per-flow cardinality-based DDoS detection

Property of volumetric DDoS attacks



When a DDoS attack is taking place, the number of distinct flows (i.e. flow cardinality) significantly increases

Threat model and deployment scenario



A memory-efficient data structure to count the number of flows targeting different destinations in the programmable switch is necessary

Spread Sketch

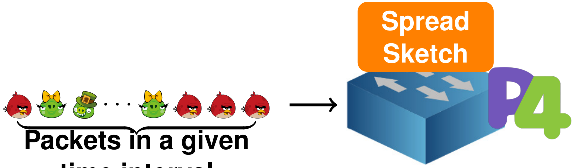
SOTA

SDN controller

dst IP	#src IPs
192.168.0.1	4000
192.168.0.2	300
192.168.0.3	200

DDoS victim

⇑ End of time interval

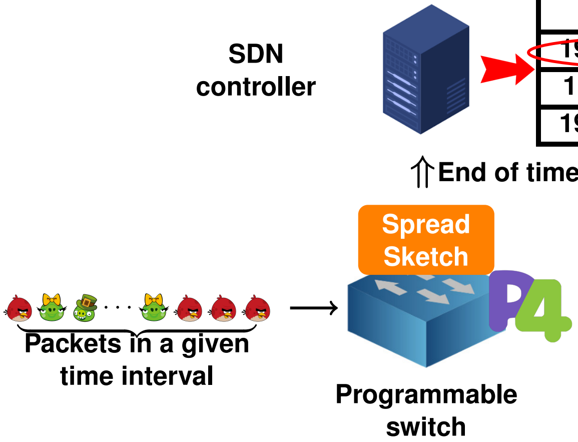


Packets in a given time interval

Programmable switch

Tang, Lu, Qun Huang, and Patrick PC Lee. "Spreadsketch: Toward invertible and network-wide detection of superspreaders." IEEE INFOCOM 2020-IEEE Conference on Computer Communications. IEEE, 2020.

Spread Sketch



dst IP	#src IPs
192.168.0.1	4000
192.168.0.2	300
192.168.0.3	200

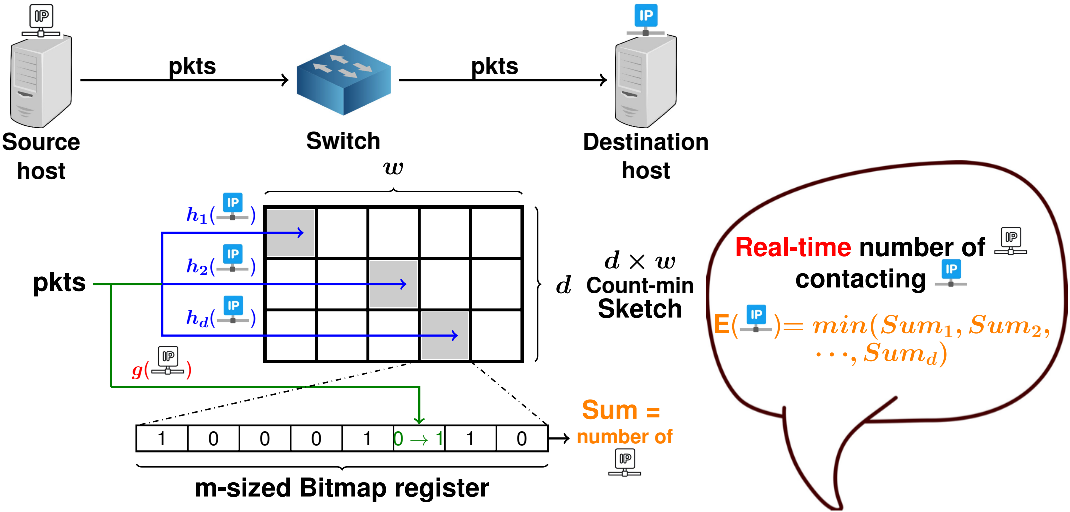
DDoS victim

Limitations:

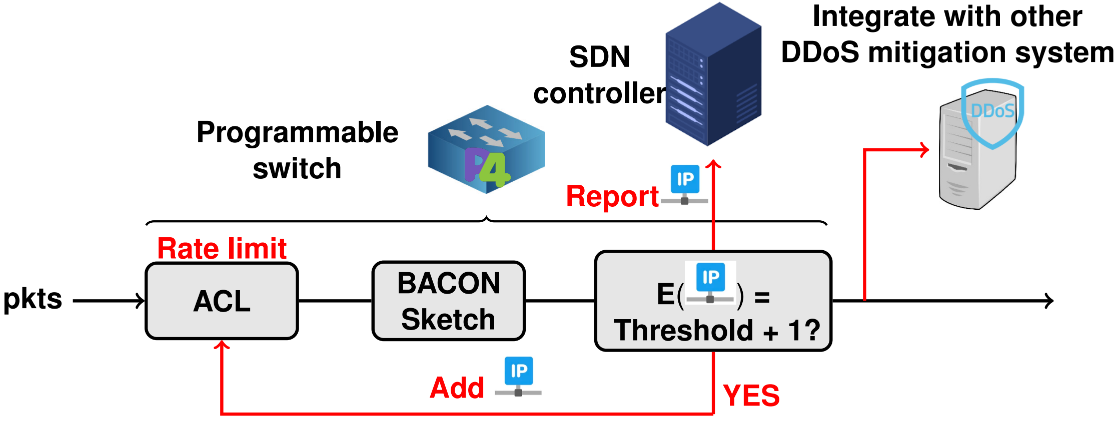
- Cannot estimate #flows in real time
- Cannot detect DDoS victims in the switch directly
- Incurs communication overhead and latency

Tang, Lu, Qun Huang, and Patrick PC Lee. "Spreadsketch: Toward invertible and network-wide detection of superspreaders." IEEE INFOCOM 2020-IEEE Conference on Computer Communications. IEEE, 2020.

BACON Sketch



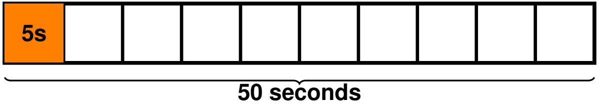
In-network DDoS victim identification (INDDoS)

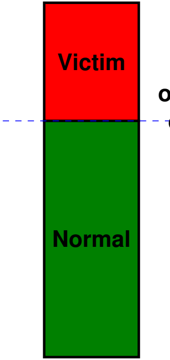


Damu Ding, Marco Savi, Federico Pederzolli, Mauro Campanella, and Domenico Siracusa. *In-Network Volumetric DDoS Victim Identification Using Programmable Commodity Switches* IEEE Transactions on Network and Service Management (TNSM).

Evaluation settings

Normal traffic (CAIDA 2018)





$\geq 0.5\%\cdot$ overall number of source IPs in the time interval

$$Recall = \frac{TP}{TP+TN} \qquad Precision = \frac{TP}{TP+FP}$$

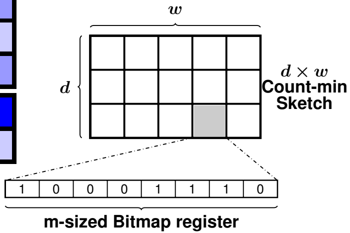
$$F1 = \frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$$

$TP = Count_{DDoSvictim}^{detected/true}$, $FP = Count_{DDoSvictim}^{detected/false}$, $TN = Count_{DDoSvictim}^{undetected/true}$

Sensitivity analysis of DDoS victim identification

BACON Sketch size ($d \times w \times m$)	Recall	Precision	F1 score
$3 \times 1024 \times 1024$	0.96	0.99	0.97
$1 \times 2048 \times 1024$	0.98	0.54	0.70
$1 \times 1024 \times 2048$	0.94	0.38	0.54
$5 \times 1024 \times 512$	0.12	1.0	0.22
$5 \times 512 \times 1024$	0.96	0.89	0.92

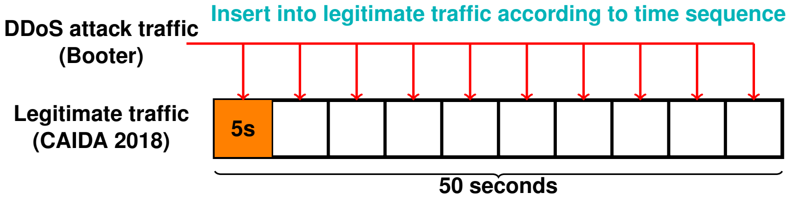
Spread Sketch[6] size ($d \times w \times m$)	Recall	Precision	F1 score
$3 \times 1024 \times 1024$	0.92	0.94	0.93

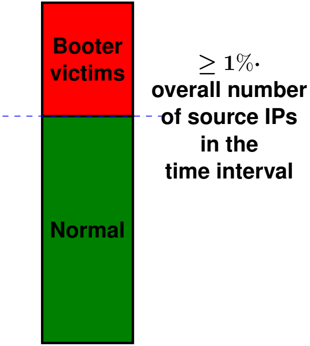


NB. Spread Sketch cannot be fully executed in programmable data planes

[6] Tang, Lu, Qun Huang, and Patrick PC Lee. "Spreadsketch: Toward invertible and network-wide detection of superspreaders." IEEE INFOCOM 2020-IEEE Conference on Computer Communications. IEEE, 2020.

DDoS victim identification accuracy under Booter attacks





DDoS attack flow trace	Recall	Precision	F1 score
Booter 6	1.0 (1/1)	1.0 (1/1)	1.0 (1/1)
Booter 7	1.0 (1/1)	1.0 (1/1)	1.0 (1/1)
Booter 1	1.0 (1/1)	1.0 (1/1)	1.0 (1/1)
Booter 4	1.0 (1/1)	1.0 (1/1)	1.0 (1/1)
Mixed	1.0 (4/4)	1.0 (4/4)	1.0 (4/4)

Conclusion

- Offload monitoring tasks from SDN controller to data plane programmable switches leveraging various memory-efficient data structures
  - Count-min Sketch
  - LogLog counting
  - Count Sketch
  - and much more ...
- Focus on smart monitoring strategies in programmable data planes
  - Network-wide heavy-hitter detection
  - Normalized entropy-based volumetric DDoS detection
  - Per-flow cardinality-based volumetric DDoS detection
  - and much more ...
- Proved network monitoring performance using programmable switches
  - High monitoring accuracy
  - Low packet processing time for monitoring
  - Valid for high-throughput networks

Thank you!