Portscan Detection with Sampled NetFlow

Ignasi Paredes-Oliva Pere Barlet-Ros Josep Solé-Pareta

Universitat Politècnica de Catalunya (UPC)
Barcelona, Spain
{iparedes, pbarlet, pareta}@ac.upc.edu

1st Workshop on Traffic Monitoring and Analysis (TMA)
Aachen, Germany. May 11, 2009

UNIVERSITAT POLITÈCNICA DE CATALUNYA

Advanced Broadband Communications Center

Outline

Outline

The problem

- Processing all packets is unfeasible in high-speed links and sampling techniques must be applied (e.g., Sampled NetFlow)
- Are actual portscan detection algorithms robust enough to sampling to continue detecting attacks reliably?
- Former studies concluded that *Flow Sampling* was best choice for portscan detection

The problem

- Processing all packets is unfeasible in high-speed links and sampling techniques must be applied (e.g., Sampled NetFlow)
- Are actual portscan detection algorithms robust enough to sampling to continue detecting attacks reliably?
- Former studies concluded that *Flow Sampling* was best choice for portscan detection

Main motivation

- NetFlow only implements *Packet Sampling*
- Unfair conditions for *Packet Sampling* in previous works
  - Taking 10% of flows results in sampling $< 3\%$ of packets for *Packet Sampling* while *Flow Sampling* keeps $> 10\%$ of packets
- What is going to happen if we change that?

Related Work and Contributions

Previous work

- Study the impact of sampling on portscan detection
  - Analyzed portscan detection algorithms: TRW, TAPS, entropy-based method
  - Using the same portion of sampled flows
- Best option: *Flow Sampling*

Related Work and Contributions

Previous work

- Study the impact of sampling on portscan detection
  - Analyzed portscan detection algorithms: TRW, TAPS, entropy-based method
  - Using the same portion of sampled flows
- Best option: *Flow Sampling*

Contributions of this paper

- Validate the impact of sampling on two well-known portscan detection algorithms (TRW and TAPS)
- Both the portion of sampled packets and flows are used
- We found that *Flow Sampling* is not always the best option for portscan detection

Outline

Sampling Techniques

Random packet sampling (PS)

- It takes each packet with probability p

Random flow sampling (FS)

- It takes each flow with probability q
  - flow = <srcIP, dstIP, srcPort, dstPort, protocol>
- Hash-based implementation

Sample and Hold (SH)

- If flow_ID(packet) already seen, packet keeped
- Otherwise, the packet is sampled with probability r
  - $r \approx h \cdot s$ (s is the size of the packet and h is the probability of sampling a single byte)

Portscan Detection Algorithms

Threshold Random Walk (TRW)

- A scanner will fail more connections than a legitimate host
- Each time a flow ends a per-srcIP ratio is updated
- Two thresholds (upper and lower) are used to take the final decision

Portscan Detection Algorithms

Threshold Random Walk (TRW)

- A scanner will fail more connections than a legitimate host
- Each time a flow ends a per-srcIP ratio is updated
- Two thresholds (upper and lower) are used to take the final decision

Time-based Access Pattern Sequential hypothesis testing (TAPS)

- Scanners connect to many more destination IPs vs ports (or vice versa) than benign hosts

$$\frac{\#accessed\_IPs}{\#accessed\_ports} > k \quad \text{or} \quad \frac{\#accessed\_ports}{\#accessed\_IPs} > k$$

- Both fractions are checked every certain interval of time
- Two thresholds (upper and lower) are used to take the final decision

Outline

Methodology

Equal portion of flows

- Same percentage of flows is taken
- FS sampling rate (q) is directly that value, the other (p and h) need empirical testing

Methodology

Equal portion of flows

- Same percentage of flows is taken
- FS sampling rate (q) is directly that value, the other (p and h) need empirical testing

N	%flows	*PS*		*FS*		*SH*	
		p	**%packets**	q	**%packets**	h	**%packets**
10	**10%**	0.028	**2.86%**	0.1	**10.90%**	1.8×10^{-4}	**15.58%**

Table: Percentage of sampled packets given a portion of sampled flows

Methodology

Equal portion of flows

- Same percentage of flows is taken
- FS sampling rate (q) is directly that value, the other (p and h) need empirical testing

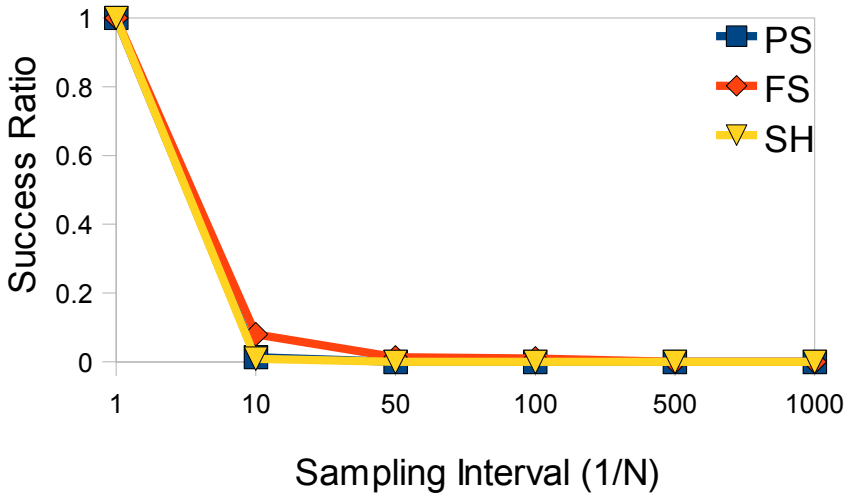
N	%flows	*PS*		*FS*		*SH*	
		p	**%packets**	q	**%packets**	h	**%packets**
10	**10%**	0.028	**2.86%**	0.1	**10.90%**	1.8×10^{-4}	**15.58%**

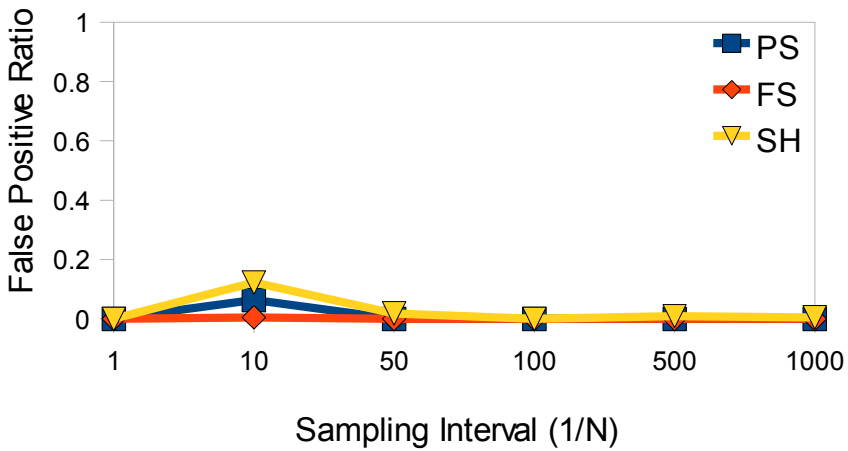
Table: Percentage of sampled packets given a portion of sampled flows

Equal portion of packets

- Same percentage of packets is taken
- PS sampling rate (p) is directly that value, the other (q and h) need empirical testing

Impact of sampling on TRW using eq. % of flows

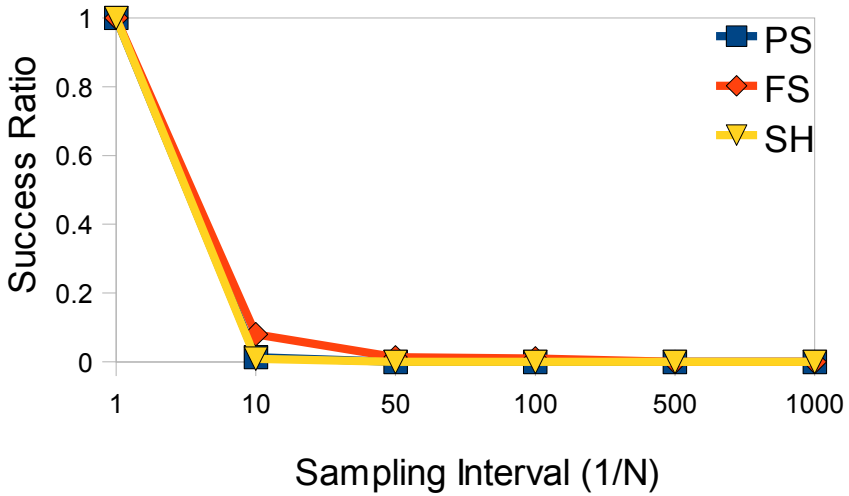


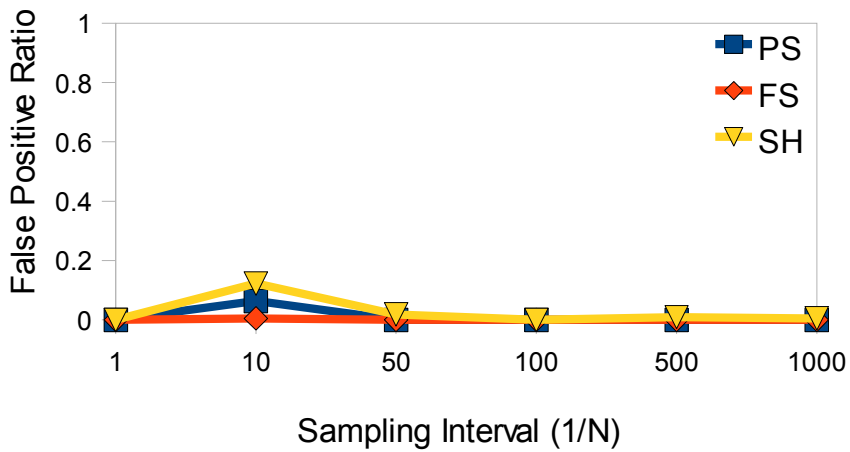


[1] $success\_ratio = \frac{true\_scanners}{total\_scanners}$

[2] $false\_positive\_ratio = \frac{false\_scanners}{total\_scanners}$

Impact of sampling on TRW using eq. % of flows





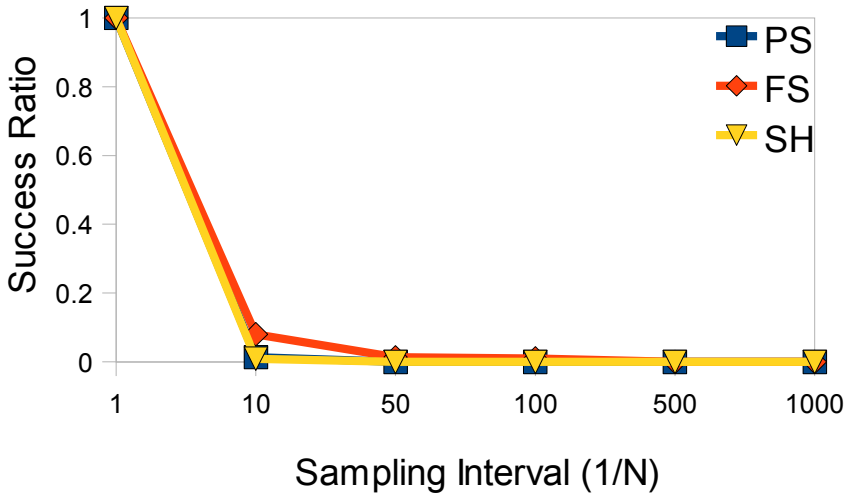
Success ratio[1] drops dramatically

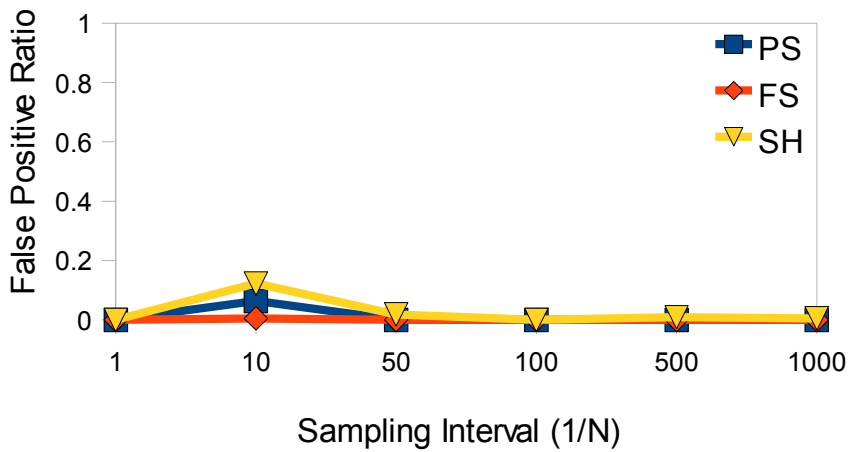
- FS, PS and SH show almost the same behaviour

[1] $success\_ratio = \frac{true\_scanners}{total\_scanners}$

[2] $false\_positive\_ratio = \frac{false\_scanners}{total\_scanners}$

Impact of sampling on TRW using eq. % of flows





Success ratio[1] drops dramatically

- FS, PS and SH show almost the same behaviour

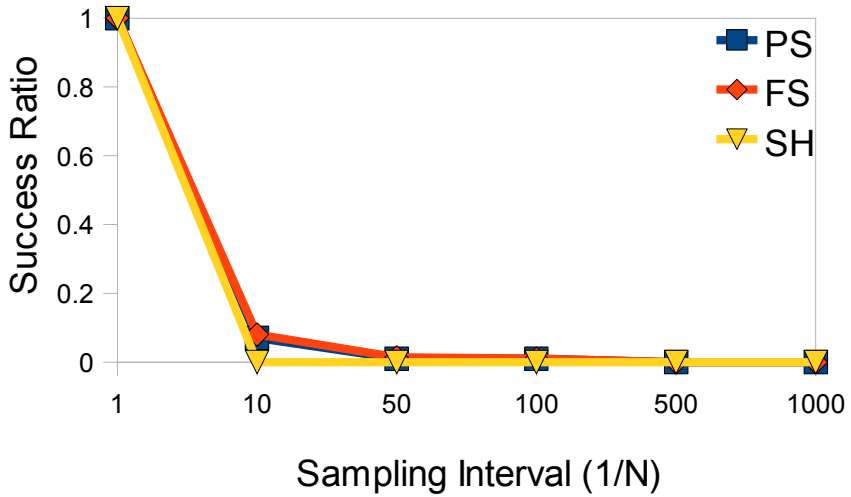
FS has no false positives[2]

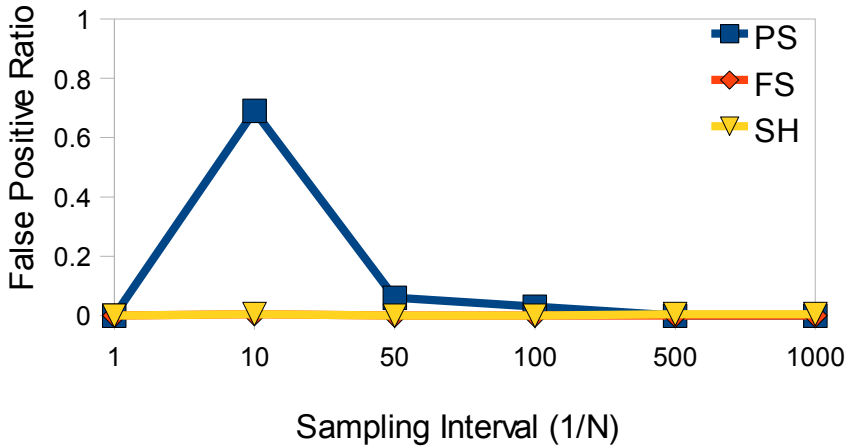
- FS is better than PS and SH

[1] $success\_ratio = \frac{true\_scanners}{total\_scanners}$

[2] $false\_positive\_ratio = \frac{false\_scanners}{total\_scanners}$

Impact of sampling on TRW using eq. % of packets

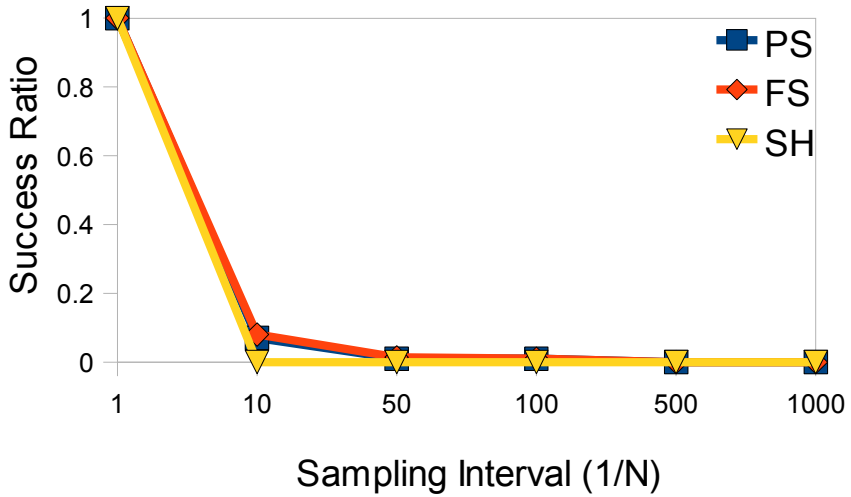


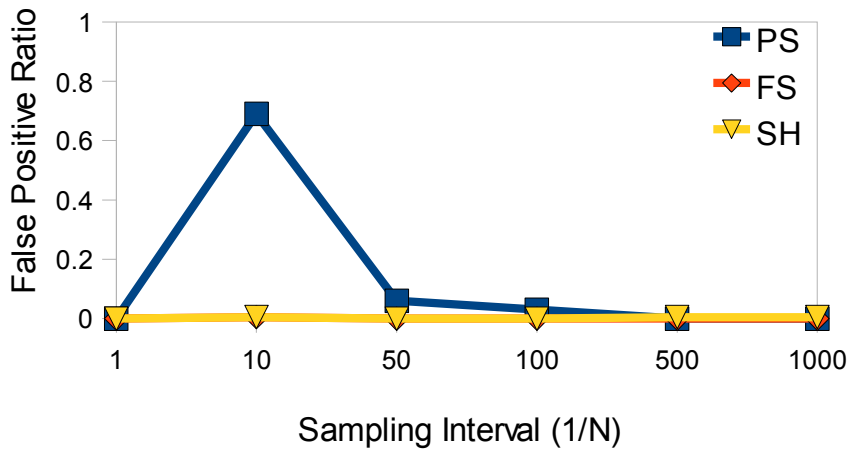


Success ratio degrades quickly

- FS and PS show almost the same behaviour in terms of success ratio

Impact of sampling on TRW using eq. % of packets





Success ratio degrades quickly

- FS and PS show almost the same behaviour in terms of success ratio

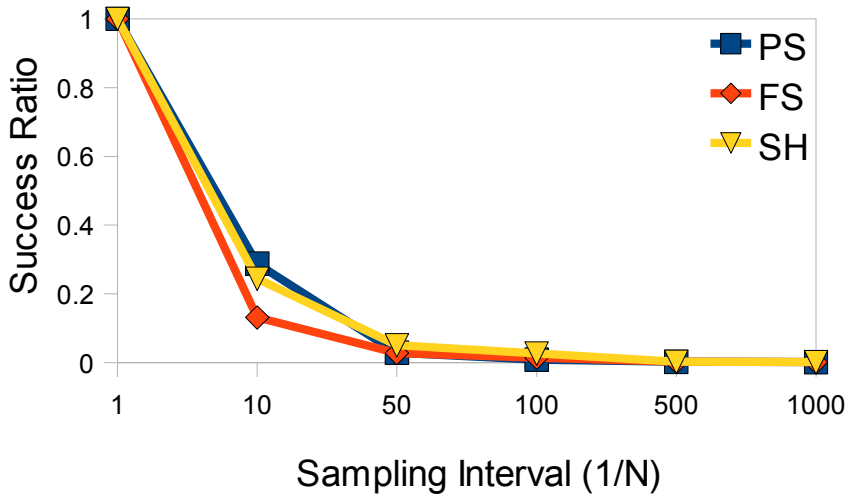
False Positves Peak

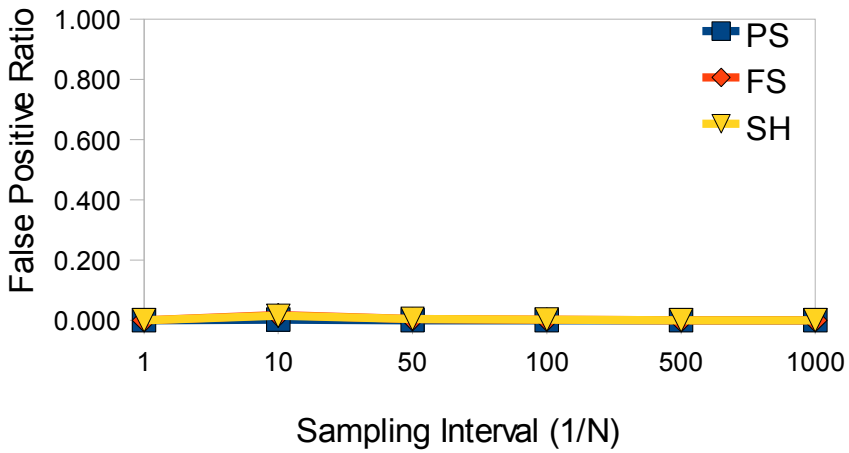
- PS reaches 70%!

Negligible false positives

- FS and SH obtain almost no false positives

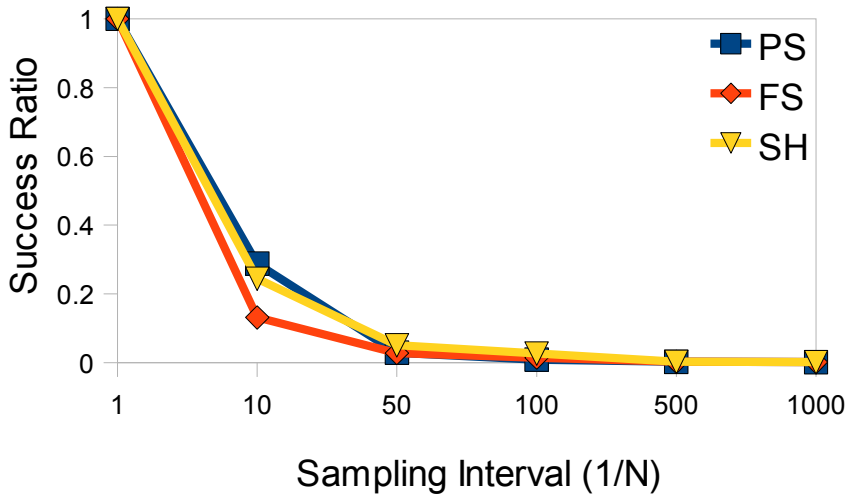
Impact of sampling on TAPS using eq. % of flows

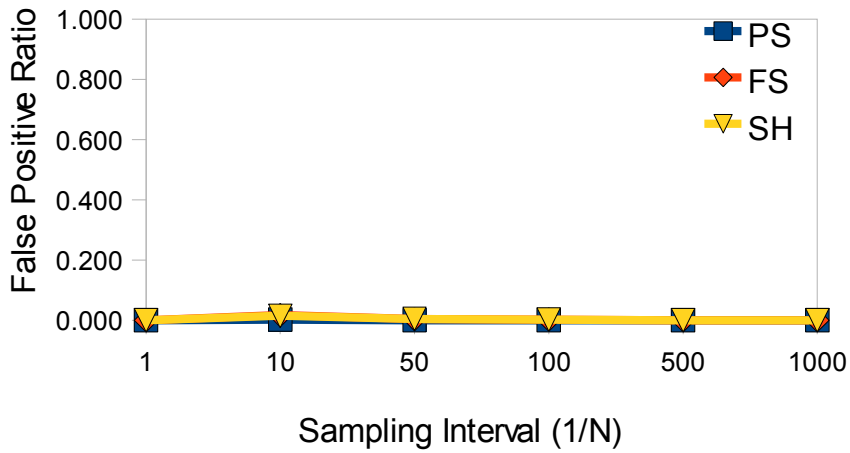




- Similar performance degradation

Impact of sampling on TAPS using eq. % of flows



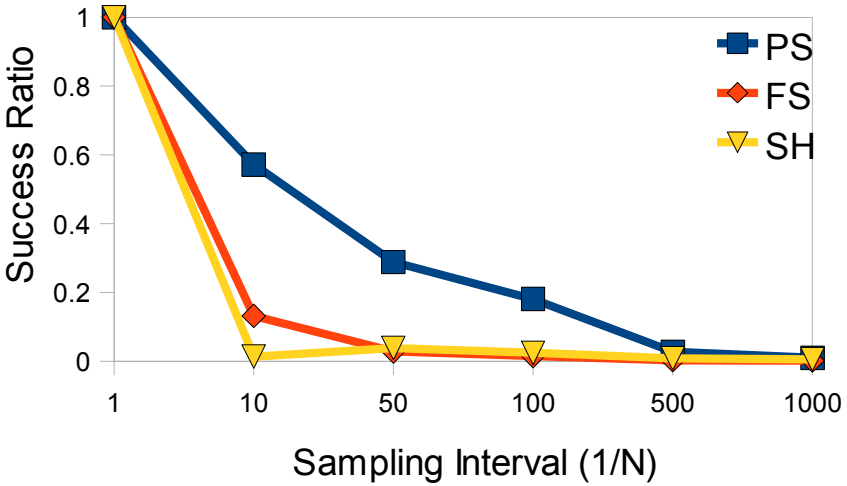


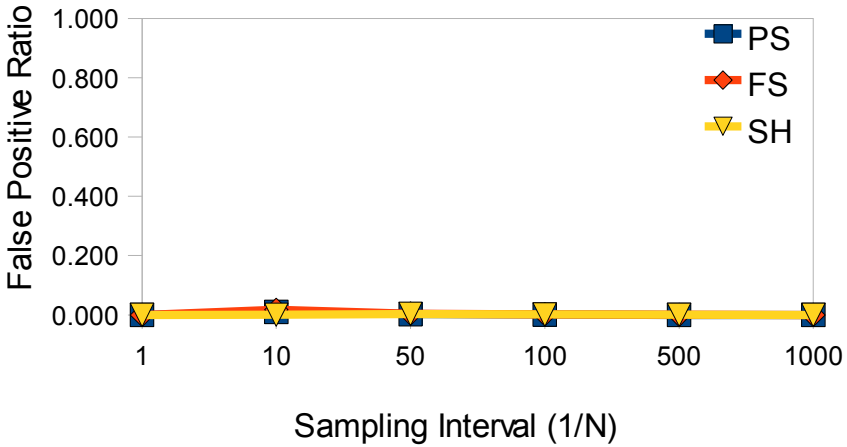
- Similar performance degradation

Equal false positives

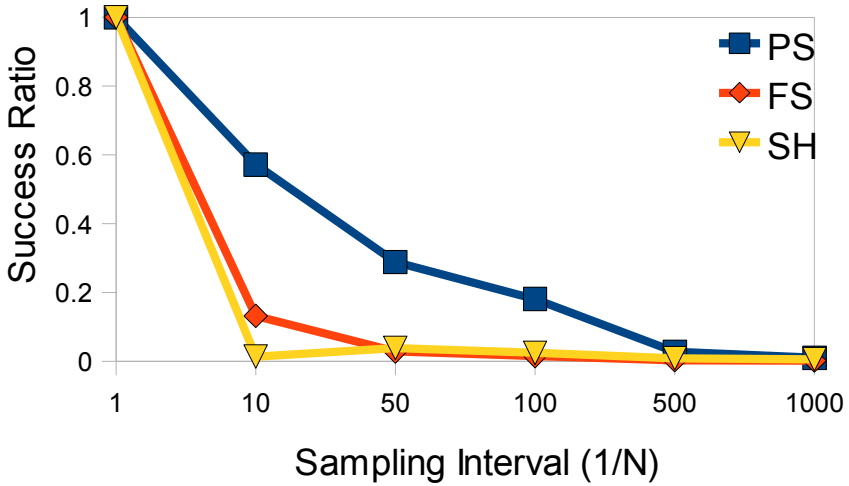
- Almost negligible

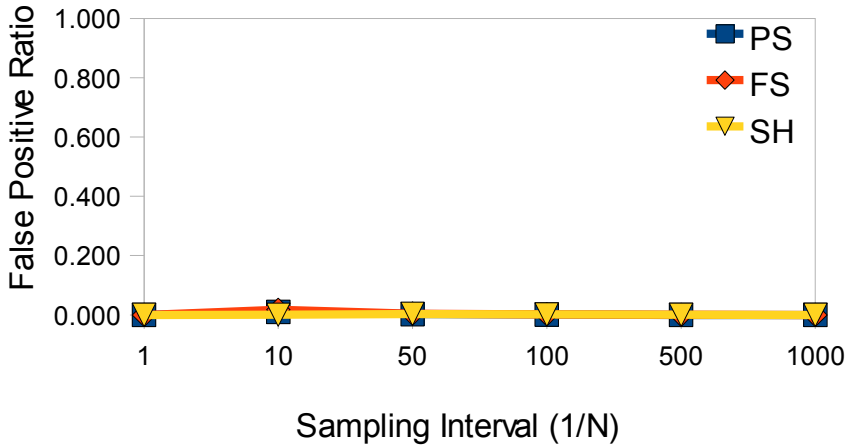
Impact of sampling on TAPS using eq. % of packets





Impact of sampling on TAPS using eq. % of packets

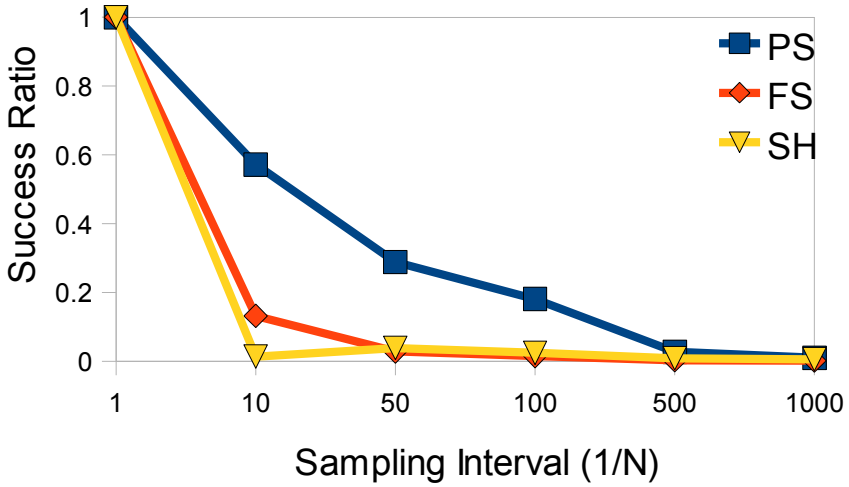


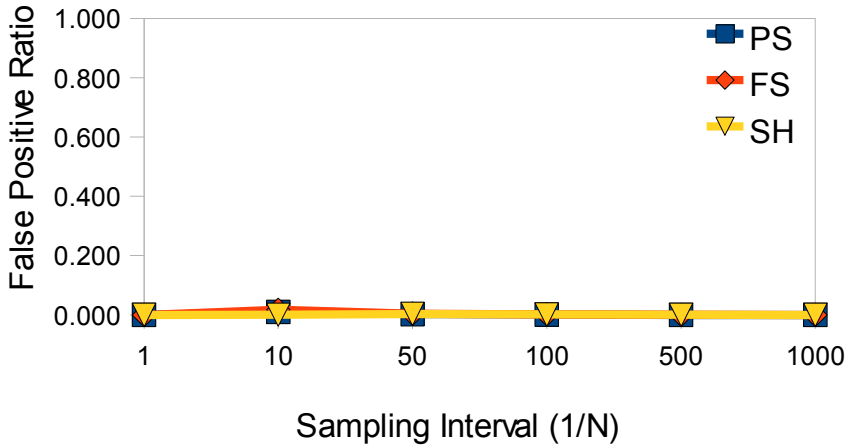


PS is clearly the best

- FS and SH success ratios degrade dramatically

Impact of sampling on TAPS using eq. % of packets





PS is clearly the best

- FS and SH success ratios degrade dramatically

PS and FS have similar false positive ratios

- FS is a slightly worse than PS and SH

Outline

Conclusions and Future Work

- Former studies used the portion of flows and concluded that *Flow Sampling* was the best choice for anomaly detection
- NetFlow only implements *Packet Sampling* and previous works treat it in an unfair way
- In addition to the same portion of flows, equal fraction of packets is used in this work
  - Under sampling, TAPS is superior than TRW
  - Performance degrades significantly for both algorithms
  - TRW: *Flow sampling* is better than *Packet Sampling*
  - TAPS: *Packet Sampling* outperforms *Flow sampling*
- Limitations and Future Work
  - Use other NetFlow traces to further validate this preliminary results
  - Study other portscan detection methods
  - Adapt portscan detection methods to make them sampling-resilient

Portscan Detection with Sampled NetFlow

Ignasi Paredes-Oliva Pere Barlet-Ros Josep Solé-Pareta

Universitat Politècnica de Catalunya (UPC)
Barcelona, Spain
{iparedes, pbarlet, pareta}@ac.upc.edu

1st Workshop on Traffic Monitoring and Analysis (TMA)
Aachen, Germany. May 11, 2009

Acknowledgments

- This work was done under the framework of the *COST Action IC0703 Data Traffic Monitoring and Analysis (TMA)*
- The authors thank UPCnet for the data traces provided for this study

Outline

Performance metrics used

Metrics

$$success\_ratio = \frac{true\_scanners}{total\_scanners} \quad \text{and} \quad false\_positive\_ratio = \frac{false\_scanners}{total\_scanners}$$

- *total_scanners*: ground truth of scanners
  - scanners detected by TRW/TAPS without sampling (not necessarily real scanners)
- *true_scanners* scanners detected under sampling that belong to the ground truth
- *false_scanners* detected scanners that fall out of that set

Our metrics differ from the classical definitions

- We do not check whether the detected scanners by TRW and TAPS (without sampling) are real scanners or not

Detailed information about the NetFlow Data

Date	Start time	Duration	Packets	Bytes	Flows	Total scanners	
						TRW	TAPS
06-11-2007	16:30	30min.	105.38×10^6	61.86×10^9	5.26×10^6	1457	4315

Portion of sampled flows given a percentage of taken packets

N	**%packets**	***PS***		***FS***		***SH***	
		p	**%flows**	p	**%flows**	h	**%flows**
10	**10%**	0.1	25.89%	0.092	10.24%	1.06×10^{-4}	6.84%
50	**2%**	0.02	7.95%	0.026	2.78%	2.8×10^{-5}	2.03%
100	**1%**	0.01	4.70%	0.015	1.85%	1.5×10^{-5}	1.05%
500	**0.2%**	0.002	1.44%	0.0036	0.95%	4×10^{-6}	0.53%
1000	**0.1%**	0.001	0.88%	0.0018	0.77%	2.7×10^{-6}	0.49%

Portion of sampled packets given a percentage of taken flows

N	%**flows**	***PS***		***FS***		***SH***	
		p	%**packets**	p	%**packets**	h	%**packets**
10	**10%**	0.028	2.86%	0.1	10.90%	1.8×10^{-4}	15.58%
50	**2%**	0.003	0.33%	0.02	1.60%	2.8×10^{-5}	1.98%
100	**1%**	1.2×10^{-3}	0.12%	0.01	0.59%	1.5×10^{-5}	1.05%
500	**0.2%**	1.3×10^{-4}	0.013%	0.002	0.11%	9.511×10^{-7}	0.02%
1000	**0.1%**	6.2×10^{-5}	0.0062%	0.001	0.05%	9.456×10^{-7}	0.018%