End-to-End Detection of Middlebox Interference

Vahab Pournaghshband
Network and Security Laboratory (Nets Lab)
Computer Science Department
University of San Francisco
vahab.p@usfca.edu

Peter Reiher
Laboratory of Advanced System Research
Computer Science Department
UCLA
reiher@cs.ucla.edu

UNIVERSITY OF SAN FRANCISCO

Nets Lab

Motivation

- Middleboxes
  - RFC 3234: “any intermediary device performing functions other than the normal, standard functions of an IP router on the datagram”
- Middleboxes are prevalent
- They can interfere with traffic
- Detecting middleboxes is important
  - Performance, security, debugging network issues

Motivation

- ***Transparent middleboxes***
  - A class of middleboxes that make no changes to the content of traffic, giving the appearance that nothing has been done to the traffic stream
  - Examples: eavesdropping, compressing/decompressing, dropping or delaying packets, encrypting/decrypting, etc.
- Detecting transparent middleboxes is challenging
- *Current practice:* if I want to show a middlebox is detectable, I develop a tool to detect the middlebox
  - We have many point solutions

Our Contribution

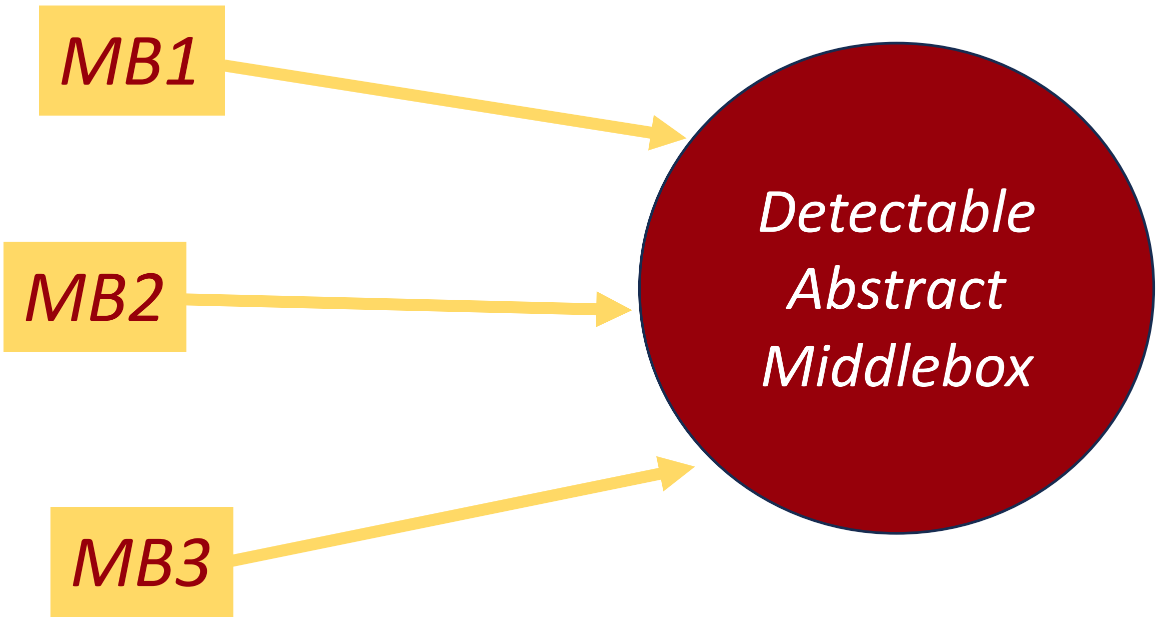
- Detecting a broader class of middleboxes
- Introducing a reduction framework to determine if a middlebox is detectable via only analytical means


Abstract
Middlebox

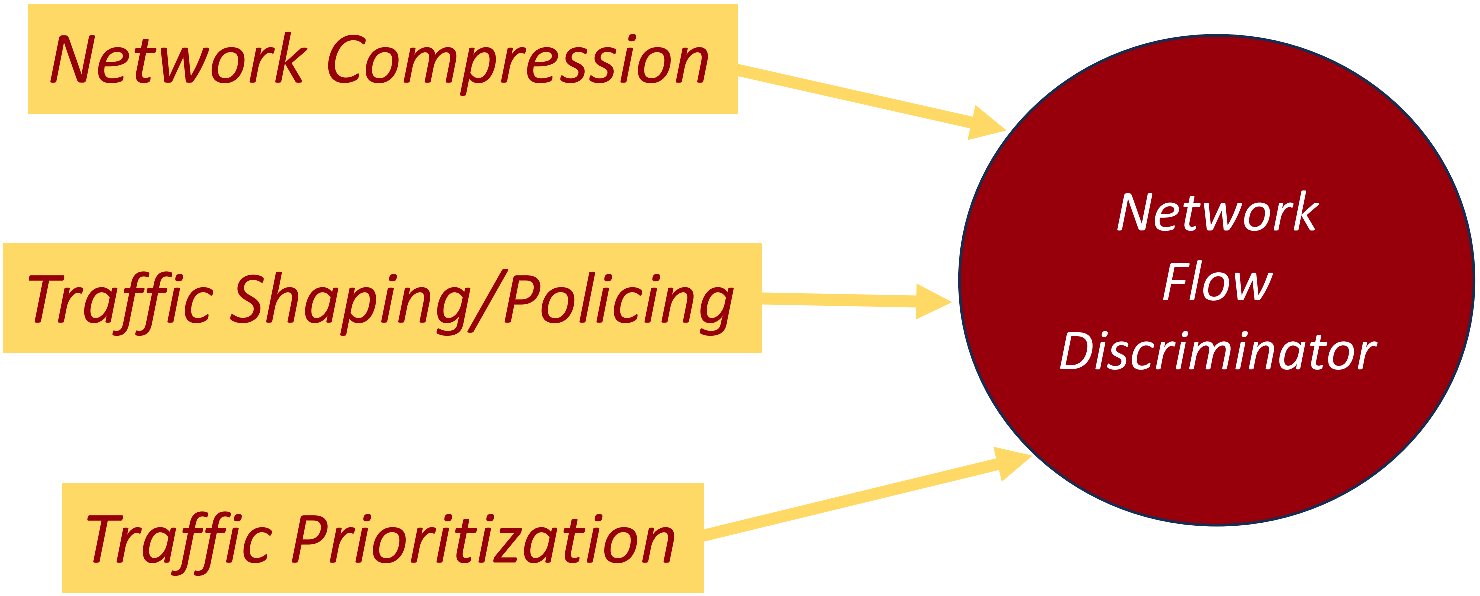
Detectable
Abstract
Middlebox

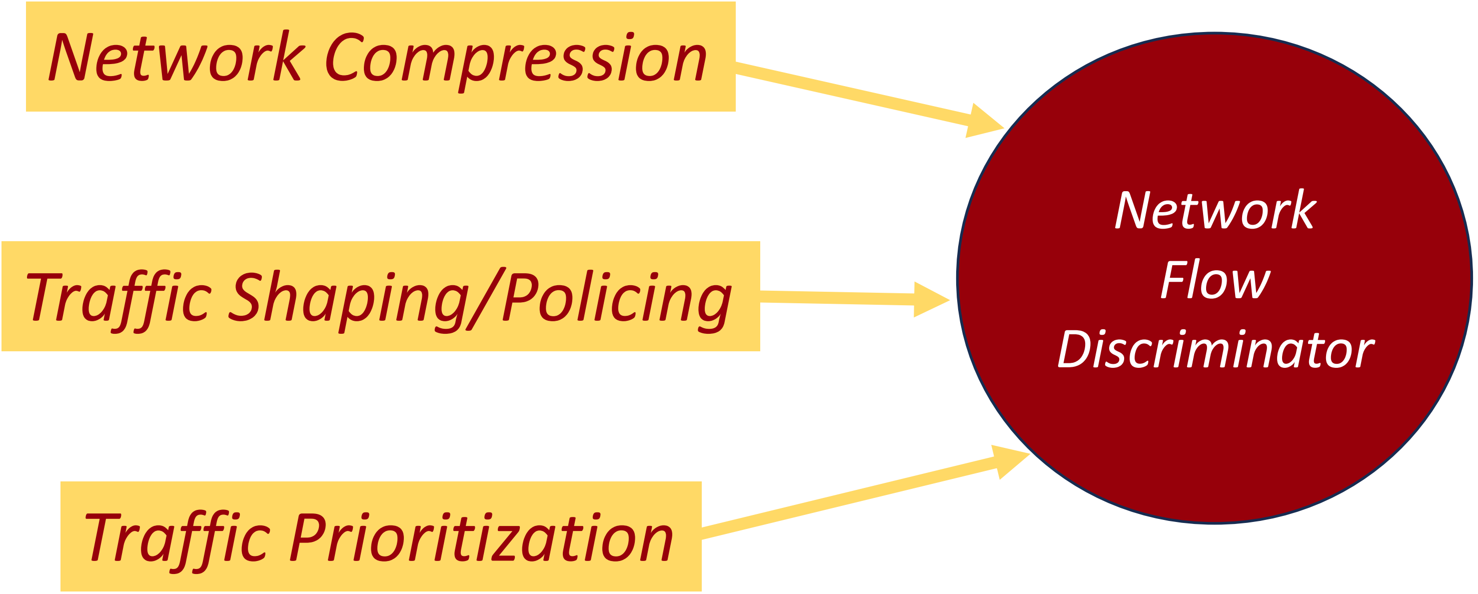

MB1
Detectable
Abstract
Middlebox


MB1
MB2
Detectable
Abstract
Middlebox


MB1
MB2
MB3
Detectable
Abstract
Middlebox


Network
Flow
Discriminator


Network Compression
Traffic Shaping/Policing
Traffic Prioritization
Network
Flow
Discriminator



We reduce the problem of “is *MB* detectable” to the problem of “is discriminator detectable” by showing that *MB* is a discriminator.

Roadmap

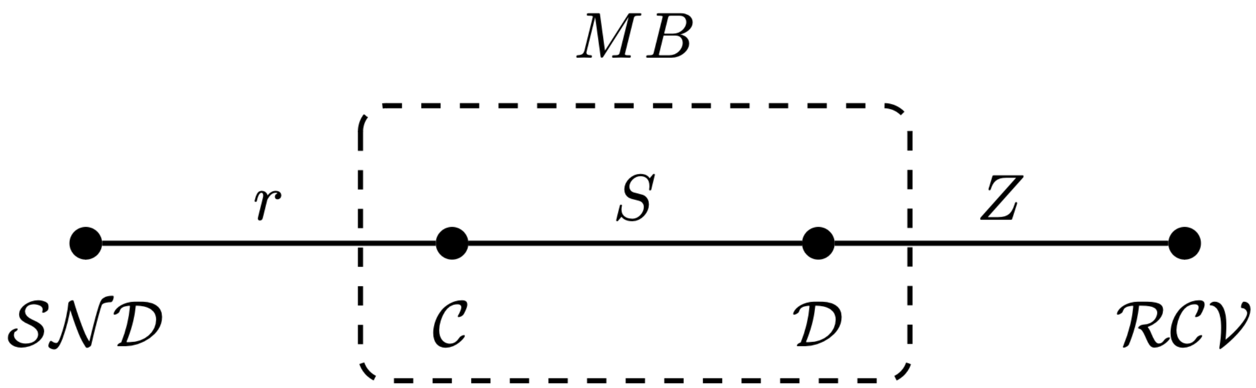
- Network Discriminators
- Detecting Network Discriminators
- Detection Framework
- Detecting Traffic Prioritization
- Results
- Future Work
- Summary

Notations

- P_A: A packet with predefined property A
- P_B: A packet with predefined property B
- MB: The middlebox in question
- $t_a(P)$: Arrival time of the last bit of packet P to the middlebox
- $t_d(P)$: Departure time (time to transmit the first bit of packet P into the middlebox
- $d_{MB}(P) := t_d(P) - t_a(P)$

Notations

- t_T: Transmission time
- c_p: The path capacity
- Z: The link capacity of the outgoing link from MB
- r: sender's sending rate of the probe packets as perceived by MB



Definition: Delay-Discriminatory Sets

An ordered pair of sets of packets (P_A, P_B) are said to be *delay-discriminatory* if:

$|P_A| = |P_B| \neq 0$

$\exists P_{A_i} \in P_A, \exists P_{B_j} \in P_B$:

$\delta_i := d_{MB}(P_{A_i}) - d_{MB}(P_{B_j}) \geq \delta_{min} > 0$

and

$t_T(P_{A_i}) \geq t_T(P_{B_j})$

and

$(P_A - \{P_{A_i}\}, P_B - \{P_{B_j}\})$ is also delay-discriminatory or is $(\emptyset, \emptyset)$

Definition: Delay-Discriminatory Sets

An ordered pair of sets of packets (P_A, P_B) are said to be *delay-discriminatory* if:

$| P_A | = | P_B | \neq 0$

$\exists P_{A_i} \in P_A, \exists P_{B_j} \in P_B$:

$\delta_i := d_{MB}(P_{A_i}) - d_{MB}(P_{B_j}) \geq \delta_{min} > 0$

and

$t_T(P_{A_i}) \geq t_T(P_{B_j})$

and

$(P_A - \{P_{A_i}\}, P_B - \{P_{B_j}\})$ is also delay-discriminatory or is $(\emptyset, \emptyset)$

Definition: Delay-Discriminatory Sets

An ordered pair of sets of packets (P_A, P_B) are said to be *delay-discriminatory* if:

$| P_A | = | P_B | \neq 0$

$\exists P_{A_i} \in P_A, \exists P_{B_j} \in P_B$:

$\delta_i := d_{MB}(P_{A_i}) - d_{MB}(P_{B_j}) \geq \delta_{min} > 0$

$\delta_{min} := \min_{\forall i} \delta_i$

and

$t_T(P_{A_i}) \geq t_T(P_{B_j})$

and

$(P_A - \{P_{A_i}\}, P_B - \{P_{B_j}\})$ is also delay-discriminatory or is $(\emptyset, \emptyset)$

Definition: Network Flow Discriminator

A middlebox MB is a network flow discriminator if, given $d_{MB}()$, there exists a pair of non-empty sets of packets (P_A, P_B) that is delay-discriminatory.

i.e., $\delta_{min} := \min_{\forall i} \delta_i > 0$

Definition: Network Flow Discriminator

A middlebox MB is a network flow discriminator if, given $d_{MB}()$, there exists a pair of non-empty sets of packets (P_A, P_B) that is delay-discriminatory.

i.e., $\delta_{min} := \min_{\forall i} \delta_i > 0$

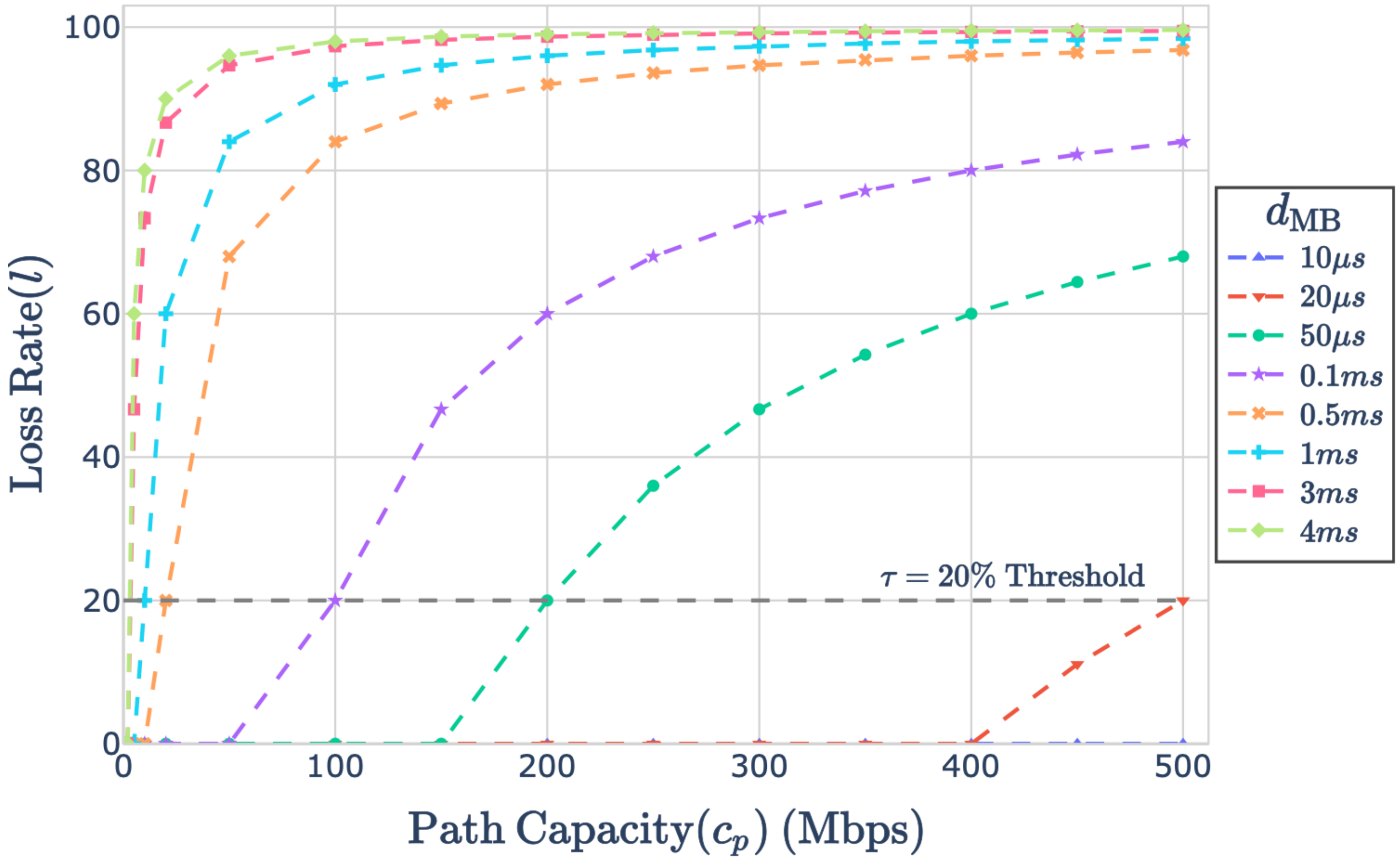
Packet drops: $\delta_i = +\infty$

Discriminator is Detectable

- We show that there exists *a* solution to detect discriminators
- Send packet trains of P_A and P_B

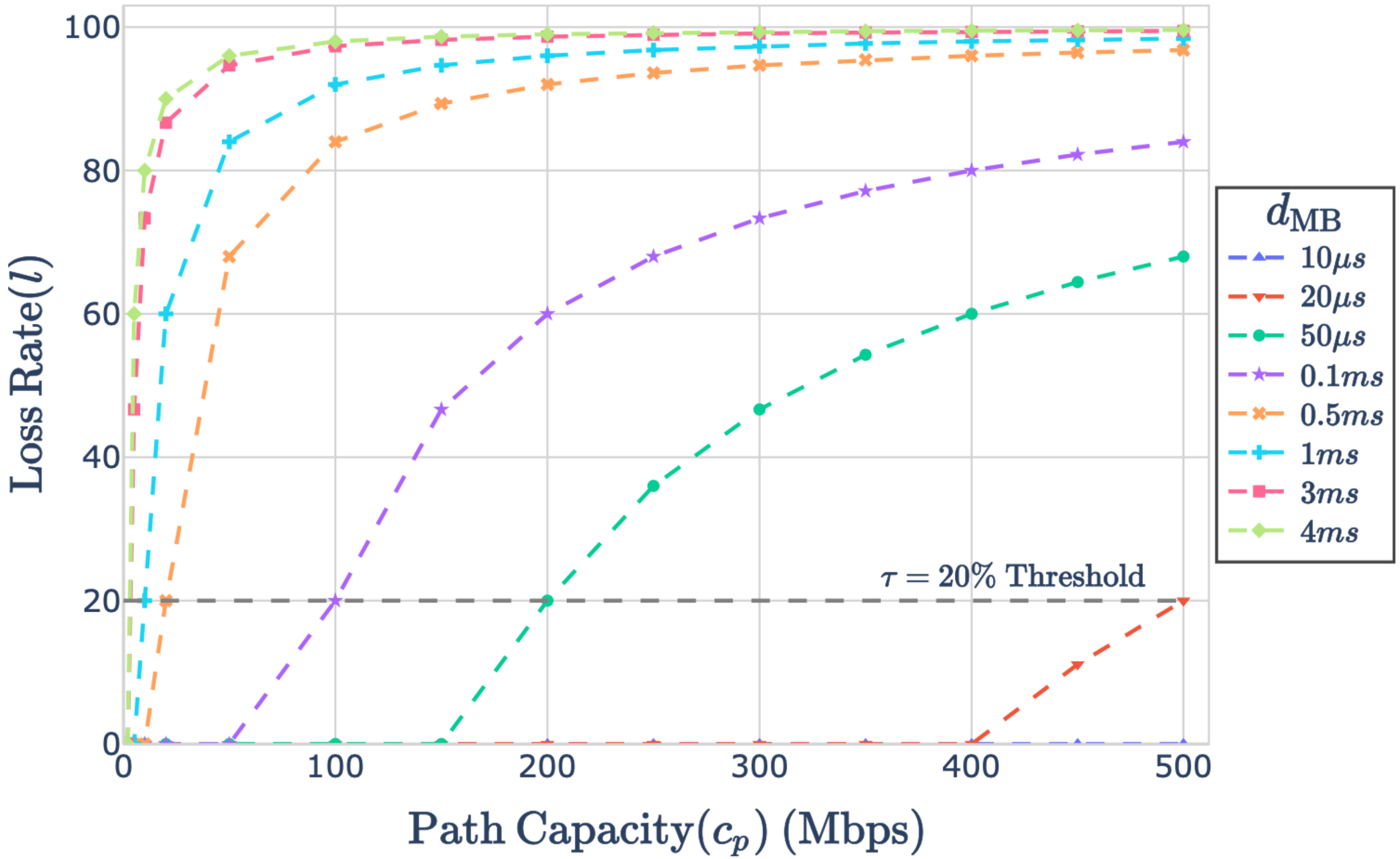
$$(P_{A_j})_{j=1}^{\rho} := \left(P_{A_1}, P_{A_2}, \ldots, P_{A_\rho}\right), (P_{B_j})_{j=1}^{\rho}$$

Discriminator is Detectable



The relationship between d_{MB} and c_p in detecting a discriminator ($\Delta l > \tau$), where ρ=1000, c_p=r

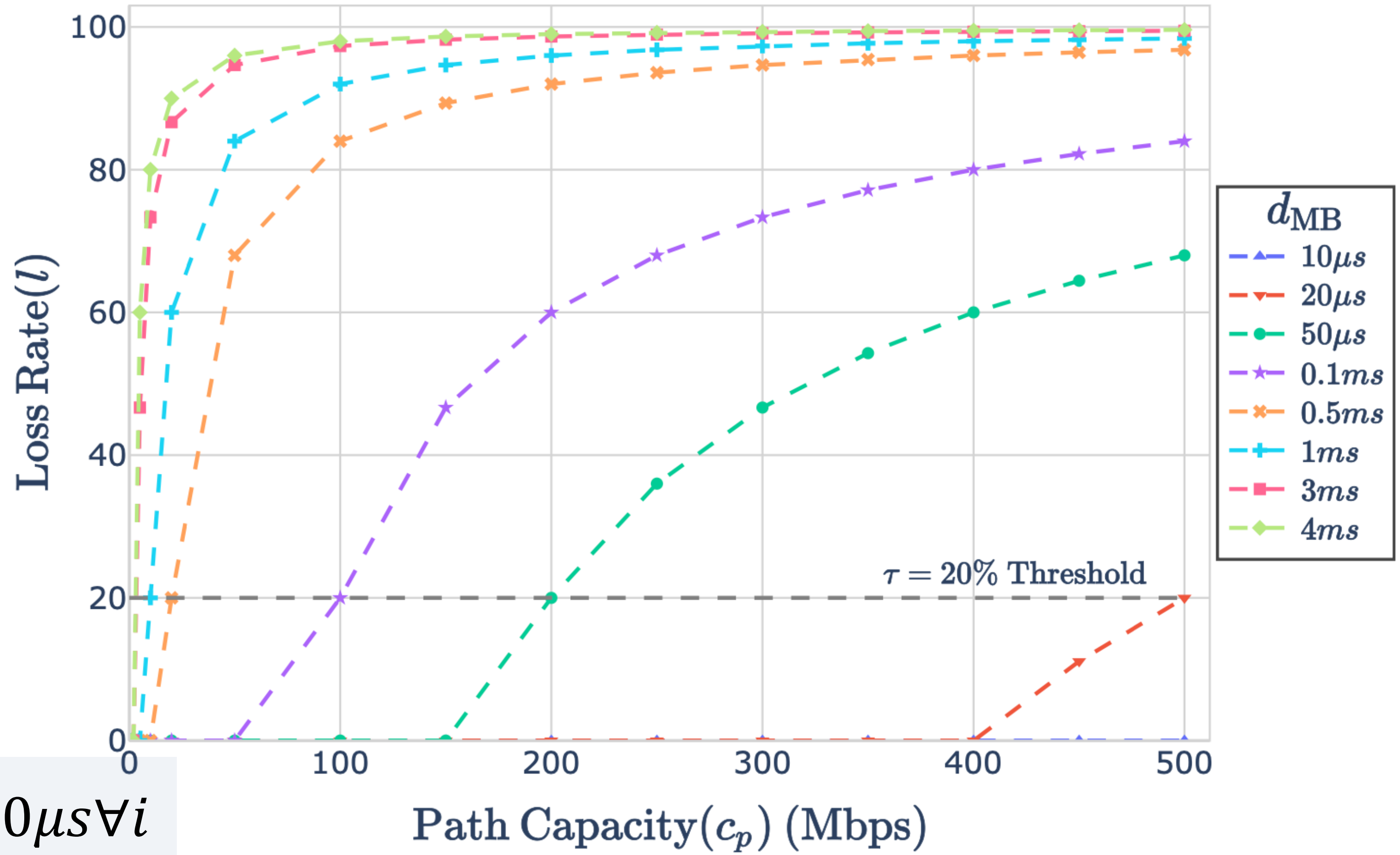
Discriminator is Detectable



$\delta_{min} = \Theta \geq 1\text{ms}$

The relationship between d_{MB} and c_p in detecting a discriminator ($\Delta l > \tau$), where ρ=1000, c_p=r

Discriminator is Detectable



$\delta_{min} = \Theta \geq 1\text{ms}$

$d_{MB}(P_{B_i}) < \theta = 10\mu s \forall i$

The relationship between d_{MB} and c_p in detecting a discriminator ($\Delta l > \tau$), where ρ=1000, c_p=r

Detection Framework

To show that a particular middlebox MB is detectable, it is sufficient to show:

1. MB is a discriminator. I.e., there exists a pair of non-empty sets (P_A, P_B) that is delay-discriminatory.

2. $\delta_{min} \geq \Theta = 1ms$, where $\delta_{min} := min\{\delta_i : 1 \leq i \leq \rho\}$.

3. $\forall i, d_{MB}(P_{B_i}) \leq \theta$ = 10μs

Detection Framework

To show that a particular middlebox MB is detectable, it is sufficient to show:

1. MB is a discriminator. I.e., there exists a pair of non-empty sets (P_A, P_B) that is delay-discriminatory.

2. $\delta_{min} \geq \Theta = 1ms$, where $\delta_{min} := min\{\delta_i : 1 \leq i \leq \rho\}$.

3. $\forall i, d_{MB}(P_{B_i}) \leq \theta$ = $10\mu s$

Detection Framework

To show that a particular middlebox MB is detectable, it is sufficient to show:

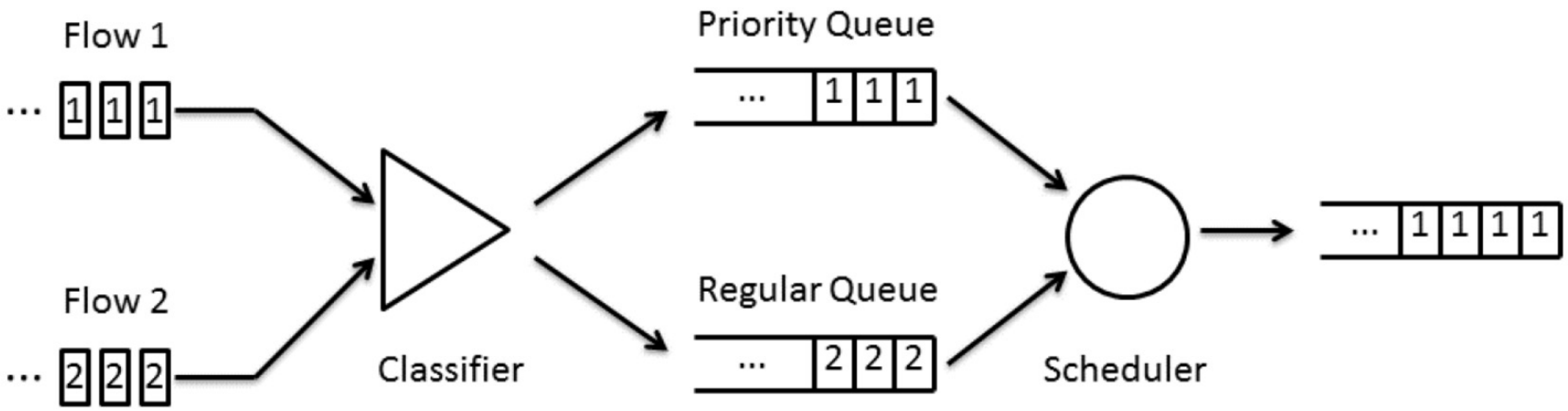
1. MB is a discriminator. I.e., there exists a pair of non-empty sets (P_A, P_B) that is delay-discriminatory.

2. $\delta_{min} \geq \Theta = 1ms$, where $\delta_{min} := min\{\delta_i : 1 \leq i \leq \rho\}$.

3. $\forall i, d_{MB}(P_{B_i}) \leq \theta$ = $10\mu s$

Detecting Traffic Prioritization

- Strict Priority Queueing (SPQ)



Detecting Traffic Prioritization

To show that SPQ is detectable, it is sufficient to show:

1. SPQ is a discriminator. I.e., there exists a pair of non-empty sets (P_A, P_B) that is delay-discriminatory.

2. $\delta_{min} \geq \Theta = 1ms$, where $\delta_{min} := min\{\delta_i : 1 \leq i \leq \rho\}$.

3. $\forall i, d_{MB}(P_{B_i}) \leq \theta$ = $10\mu s$

Detecting Traffic Prioritization

To show that SPQ is detectable, it is sufficient to show:

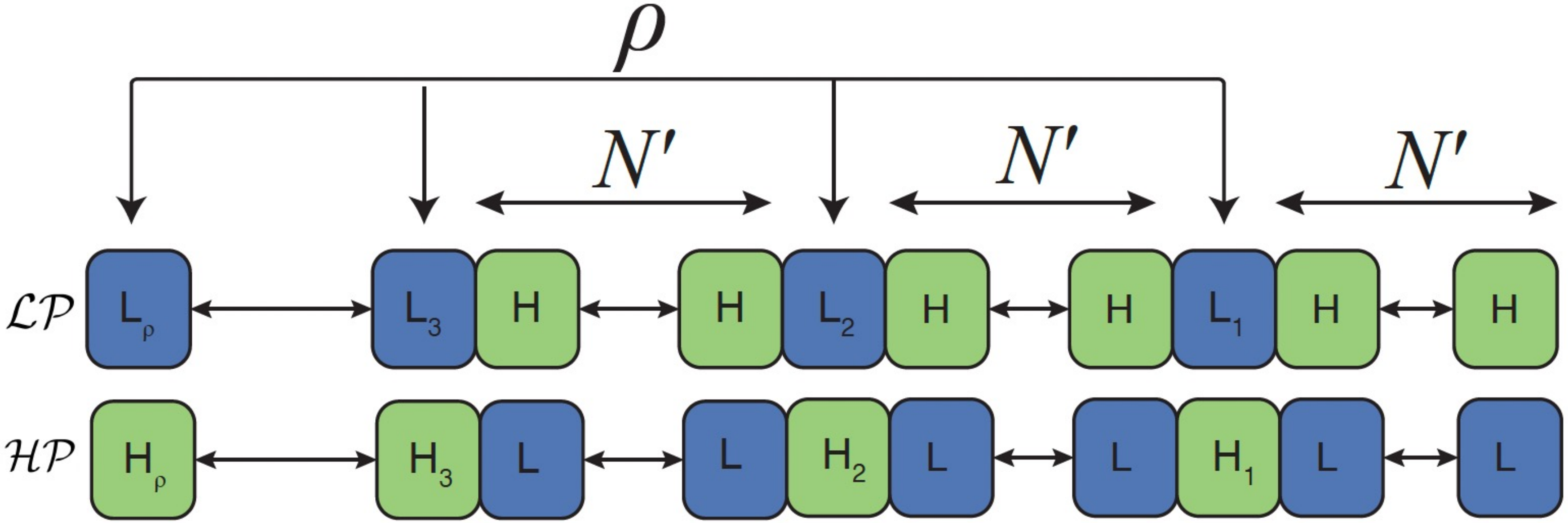
1. SPQ is a discriminator. I.e., there exists a pair of non-empty sets (P_A, P_B) that is delay-discriminatory.

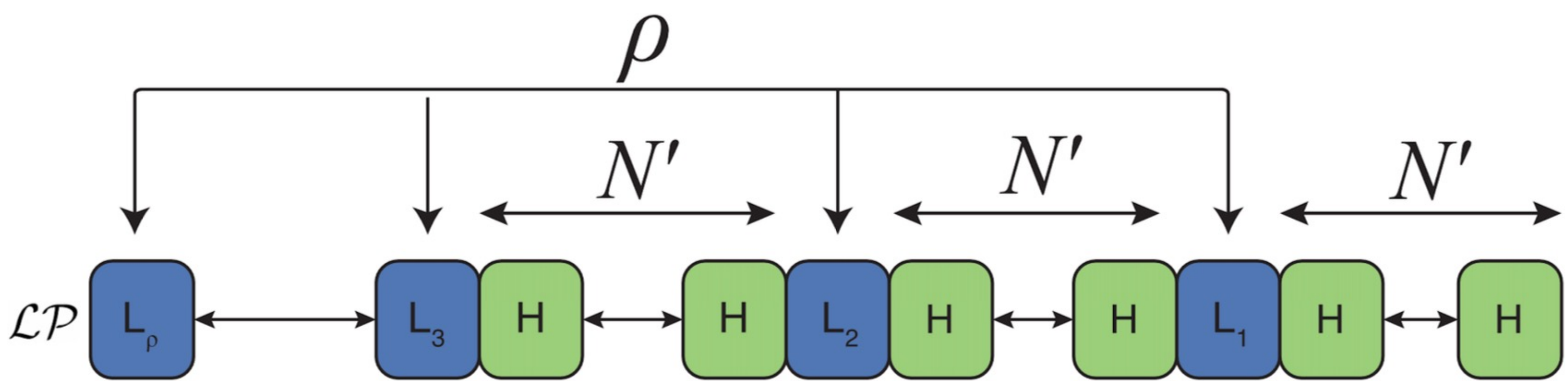
2. $\delta_{min} \geq \Theta = 1ms$, where $\delta_{min} := min\{\delta_i : 1 \leq i \leq \rho\}$.

3. $\forall i, d_{MB}(P_{B_i}) \leq \theta = 10\mu s$

Detecting Traffic Prioritization

- P_H: The packet that is prioritized.
- P_L: The packet that is not prioritized.





Assuming $r > Z$, to accomplish these objectives, the following conditions must hold:

(i) High priority packets in HP are sent at the rate of $\frac{r}{N'+1}$. Hence, for Q_H to never build up, $\frac{r}{N'+1} \leq Z$ must hold.

(ii) High priority packets in LP are sent at the rate of $\frac{N'}{N'+1}r$. Therefore, for Q_H to never be empty, the condition $\frac{N'}{N'+1}r \geq Z$ must be satisfied.

Combining conditions (i) and (ii), we have $\frac{r}{N'+1} \leq Z \leq \frac{N'}{N'+1}r$. We can now solve for N':

$$N' \geq \left\lceil max\left\{\frac{r-Z}{Z}, \frac{Z}{r-Z}\right\}\right\rceil$$

Detecting Traffic Prioritization

To show that SPQ is detectable, it is sufficient to show:

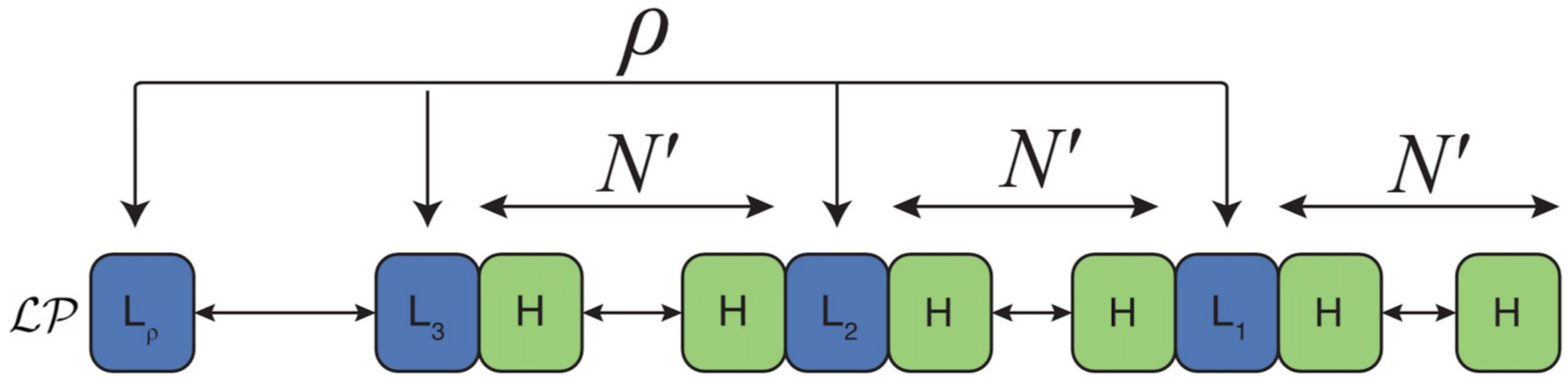
1. SPQ is a discriminator. I.e., there exists a pair of non-empty sets (P_A, P_B) that is delay-discriminatory.

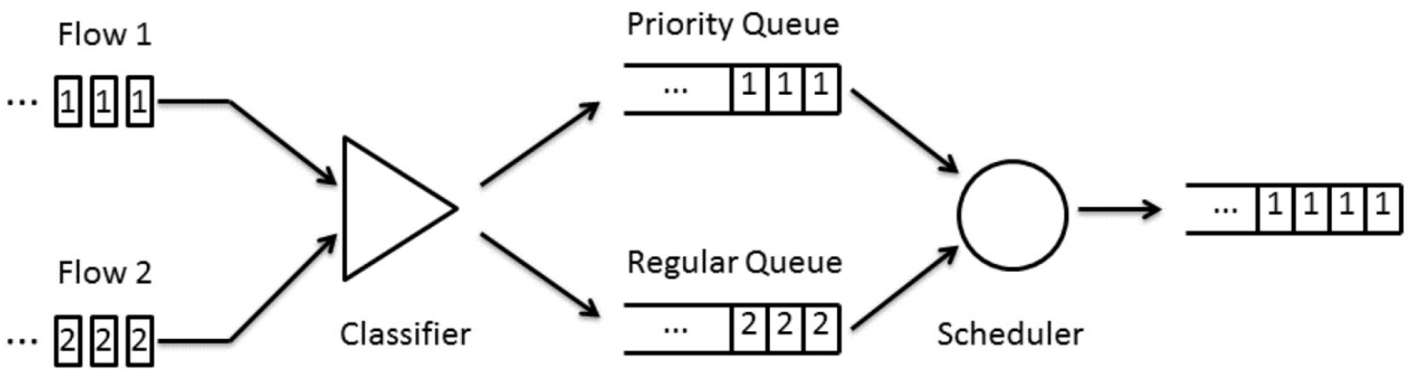
2. $\delta_{min} \geq \Theta = 1ms$, where $\delta_{min} := min\{\delta_i : 1 \leq i \leq \rho\}$.

3. $\forall i, d_{MB}(P_{B_i}) \leq \theta$ = $10\mu s$

Detecting Traffic Prioritization

$$d_{MB}(P_H) = 0$$

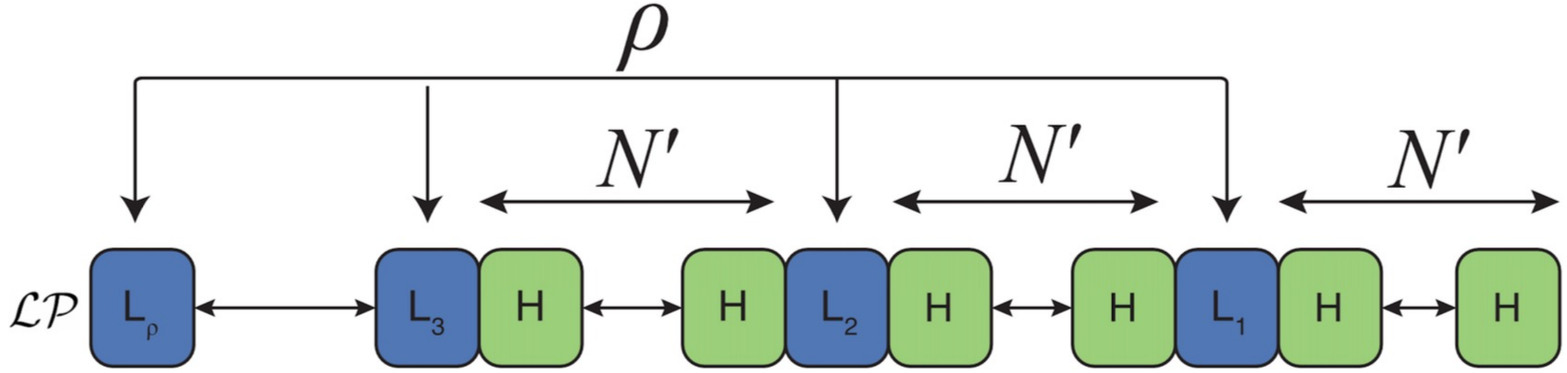


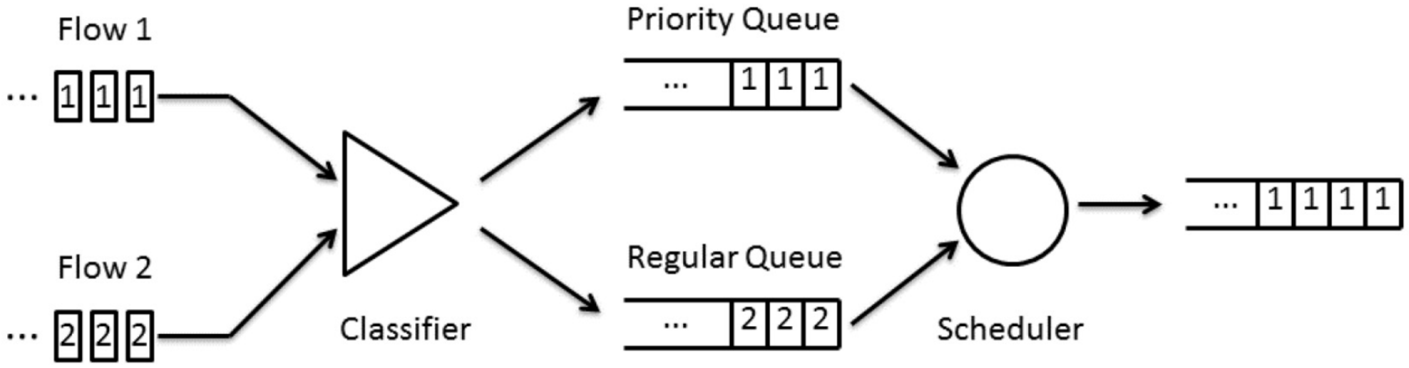


Detecting Traffic Prioritization

$$d_{MB}(P_H) = 0$$

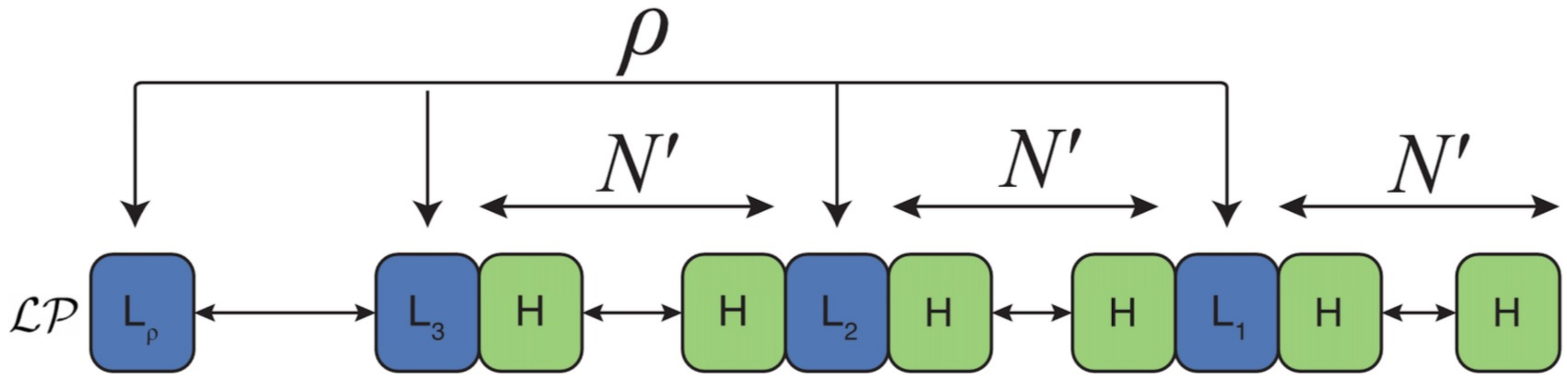
$$\delta_i = d_{MB}(P_{L_i})$$



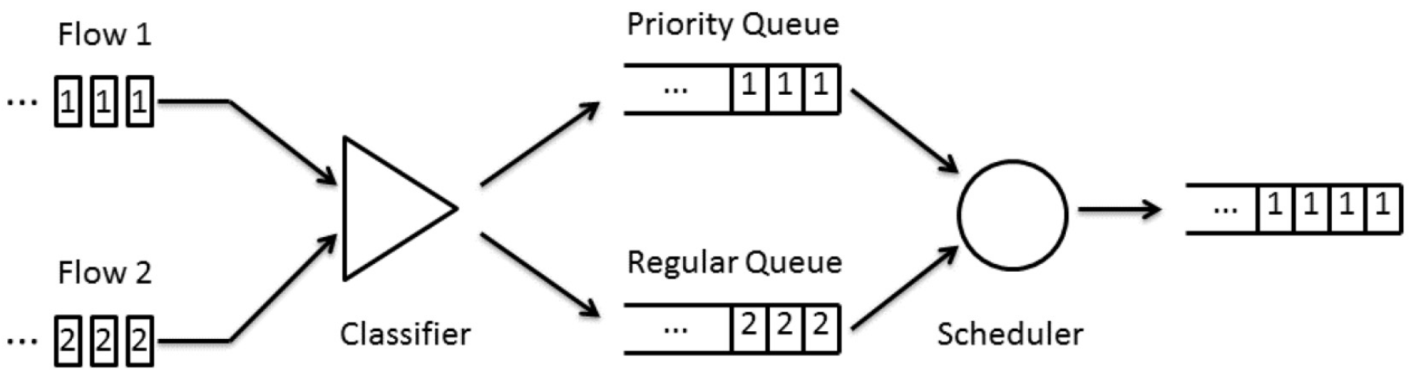


Detecting Traffic Prioritization

$$d_{MB}(P_H) = 0$$



$$\delta_i = d_{MB}(P_{L_i})$$



$$\delta_i = +\infty$$

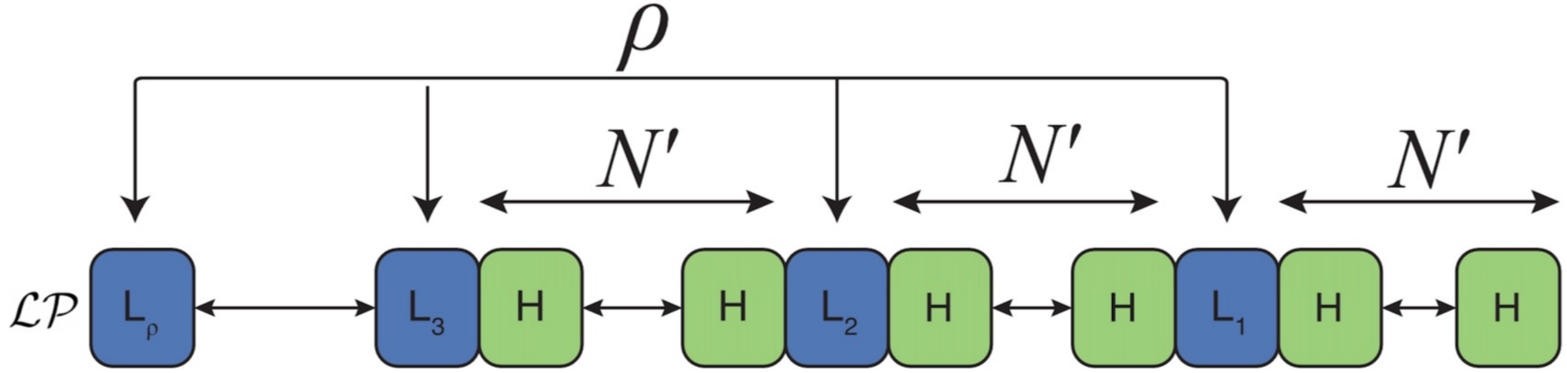
Detecting Traffic Prioritization

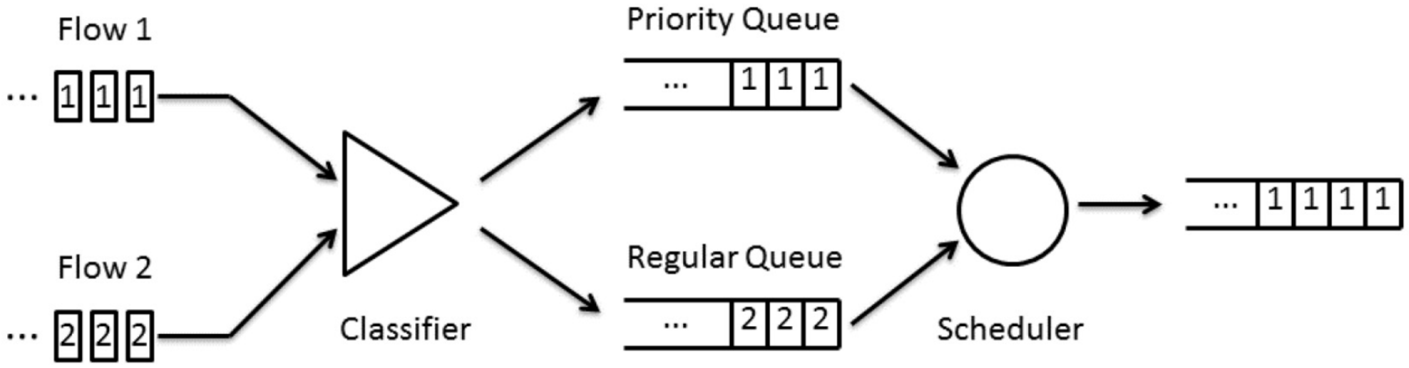
$$d_{MB}(P_H) = 0$$

$$\delta_i = d_{MB}(P_{L_i})$$

$$\delta_i = +\infty$$

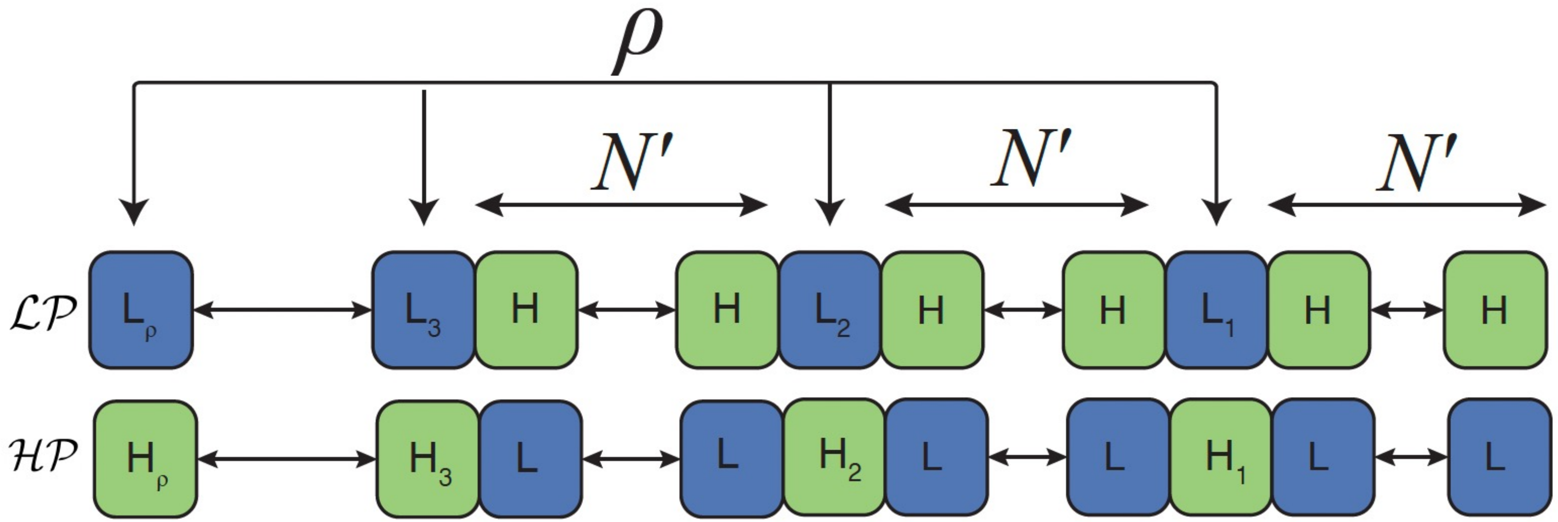
$$\delta_i \geq N' \cdot \rho \cdot t_T(P_H)$$



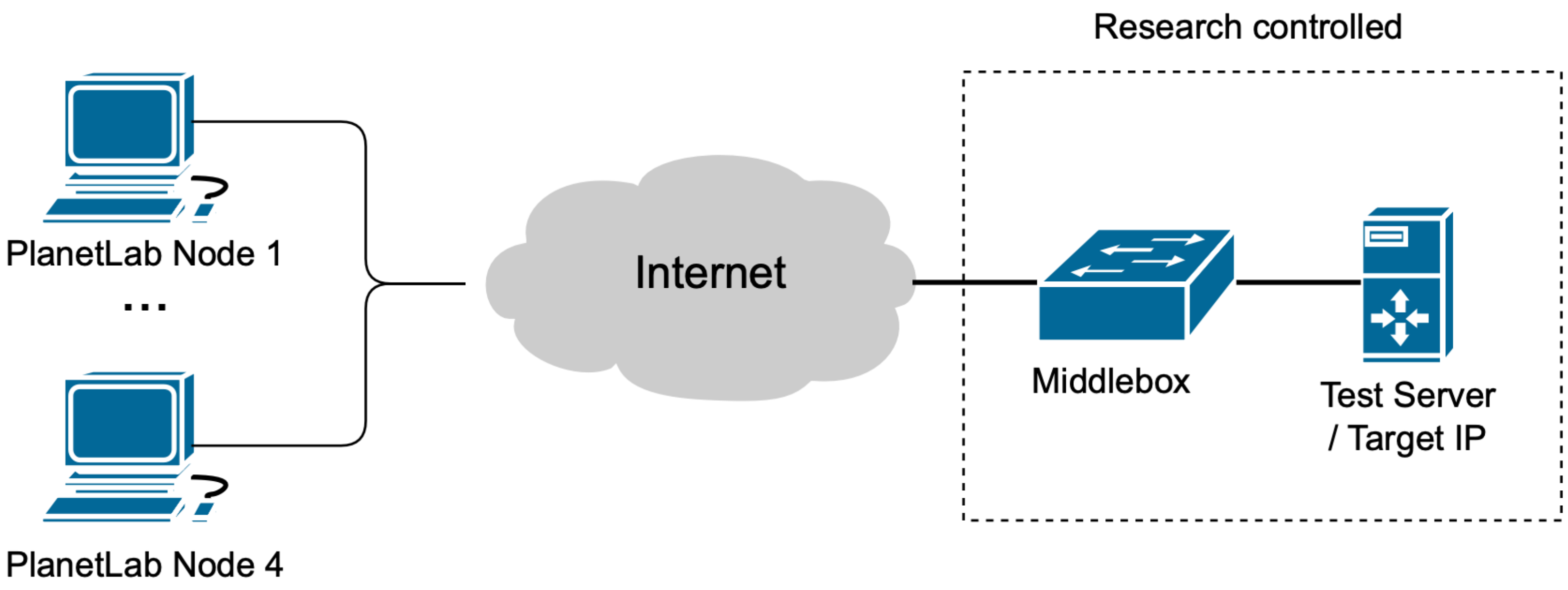


Detection Tool Development

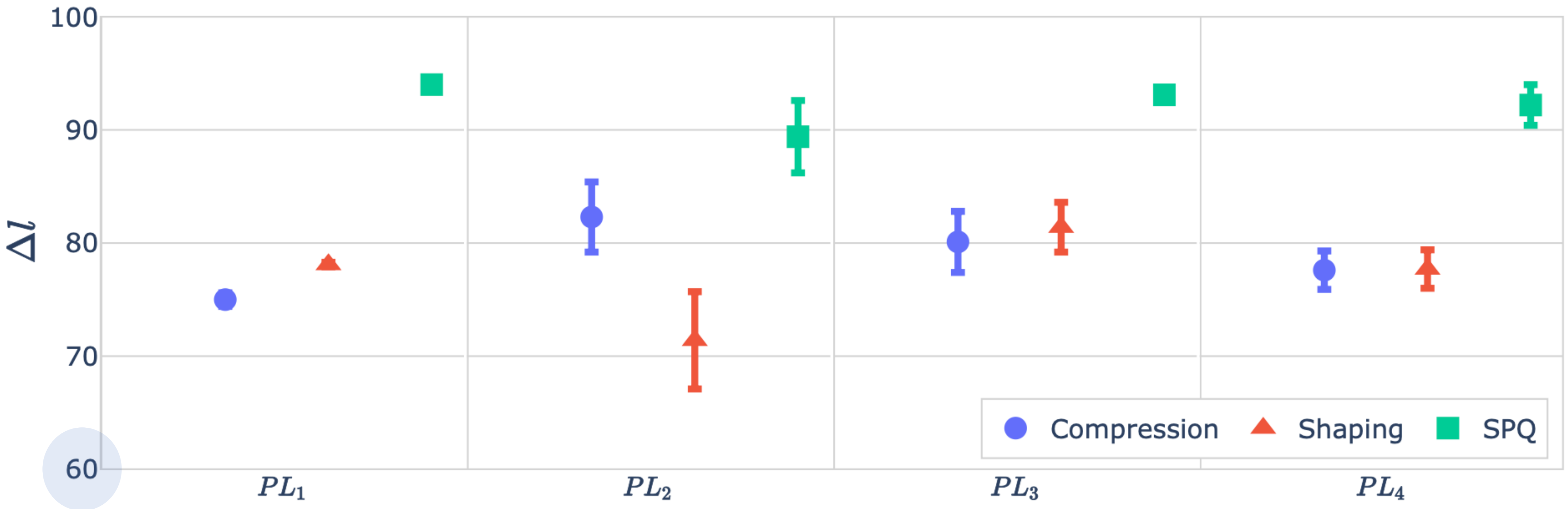
- The analytical reduction inspired us to develop a tool to detect the middleboxes
- Traffic prioritization:



Results: Environment Setup



Results



100
90
80
70
60
Δl
PL_1
PL_2
PL_3
PL_4
Compression
Shaping
SPQ

Ongoing and Future Work

- We are/will
  - Formulate additional abstract classes
  - Explore alternative solutions to detect discriminators
  - Identify more middleboxes whose interference can be described as discriminators
- We are investigating detecting middleboxes in dynamic environment

Summary

- We showed
  - An example of an abstract class: discriminators
  - A loss-based solution to detect discriminators
  - Three middleboxes---network compression, traffic shaping and prioritization--- whose interference are described as discriminators