SDN-based Detection Method against DoS/DDoS attacks in an IoT environment

1F10180066 | Abdul Adhim

Supervisor: Takuho Mitsunaga

1. Problem

The problem

- The growth of IoT devices.
- Just in 2021, the number of IoT devices alone is recorded to be 31 billion.
- The rise of IoT devices could cause some security issues.
  - Such as incorrect access control, overly large attack surface and lack of encryption.

To fully realize the potential of IoT in the future it is necessary to consider the security aspect of it.

These security issues include

- **Denial of Service (DoS)**
- **Distributed Denial of Service (DDoS)**

How can we secure these devices from DoS/DDoS?

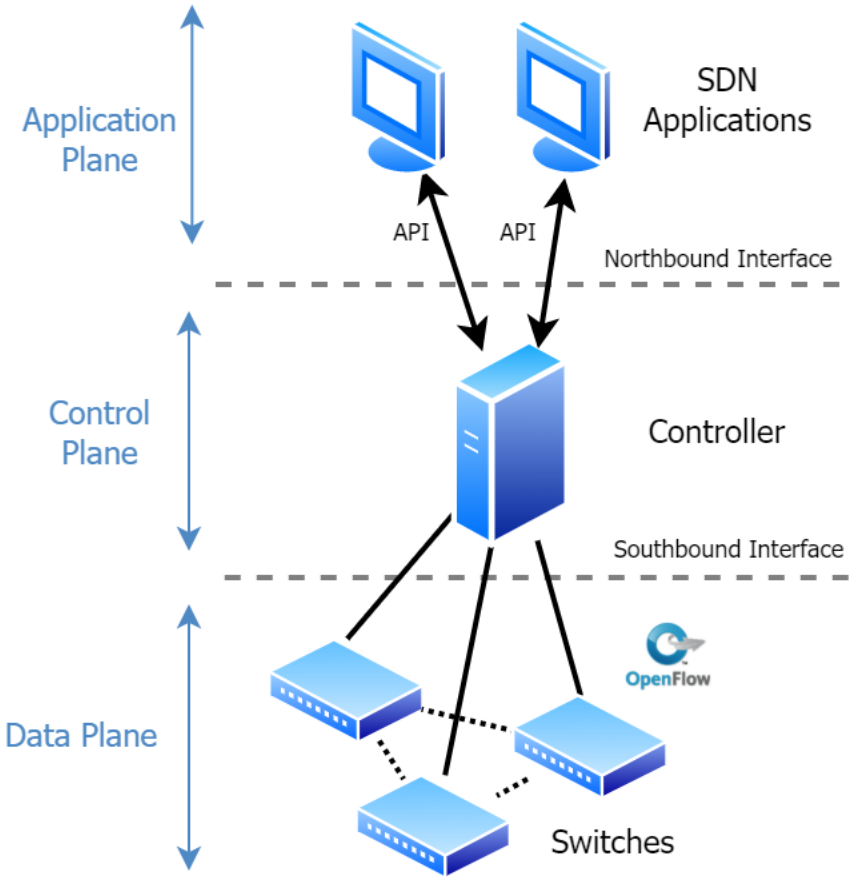
2. Solution

Solution

- (1) To use SDN
- (2) To use entropy-based detection

Solution : (1) to use SDN

- What is SDN?
  - SDN (Software-Defined Network)
  - ➢realized by *virtualizing its components*.
  - ➢can be centrally controlled with software applications.
  - ➢made up of 3 layers.
- Why SDN?
  - provide flexibility and scalability to the network.
  - more secure networking by making it easy for software updates.



Solution : (2) to use entropy-based detection

- What is entropy-based detection?
  - In information theory, entropy can be used to measure the uncertainty of information.
  - calculation can be done with Shannon entropy

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

Where there is an information source,
n = independent symbols
p_i = probability of each n
H = entropy value

Solution : (2) to use entropy-based detection

- What is entropy-based detection?
  - In information theory, entropy can be used to measure the uncertainty of information.
  - calculation can be done with Shannon entropy

$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

Where there is an information source,
n = independent symbols
p_i = probability of each n
H = entropy value

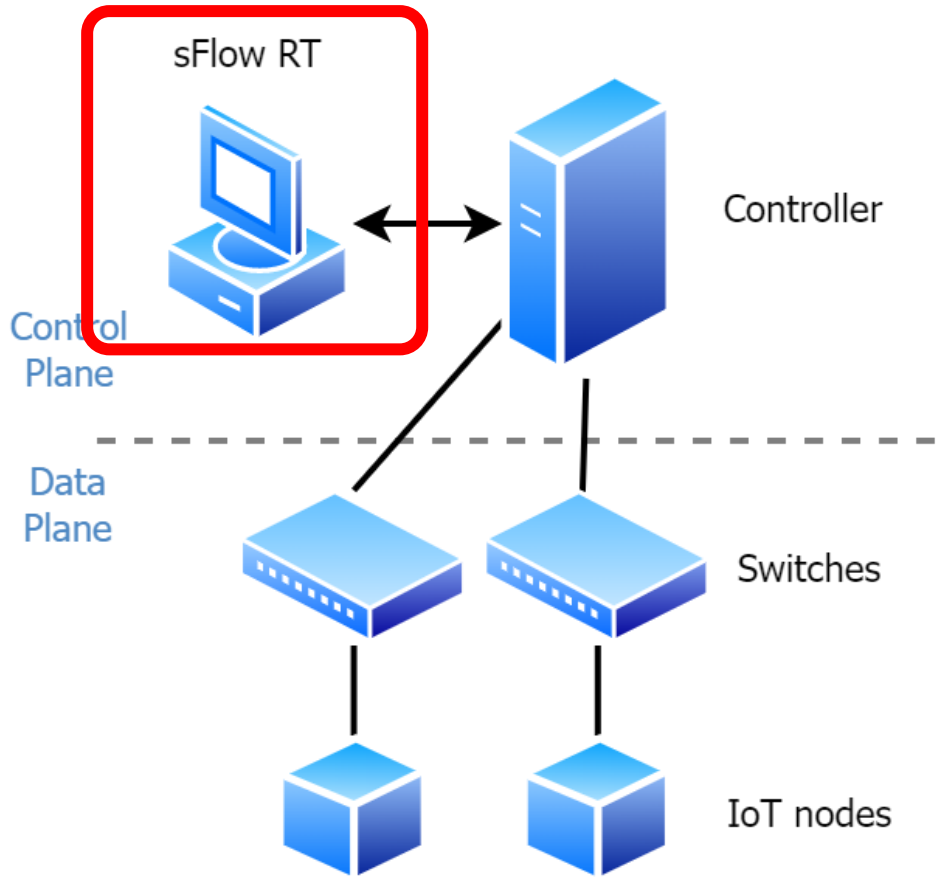
$$H = -\sum_{i=1}^{c} \left(\frac{x_i}{n}\right) \log_2 \left(\frac{x_i}{n}\right)$$

Where there is an information source,
c = total number of connections from hosts
x_i = number of travelling packets from each i^{th} connection
n = total number of travelling packets in the network
H = entropy value

3. Proposed Method

Proposed method

- Calculate entropy value of the network on the Control Plane
- Using *sFlow-RT* for calculating the entropy



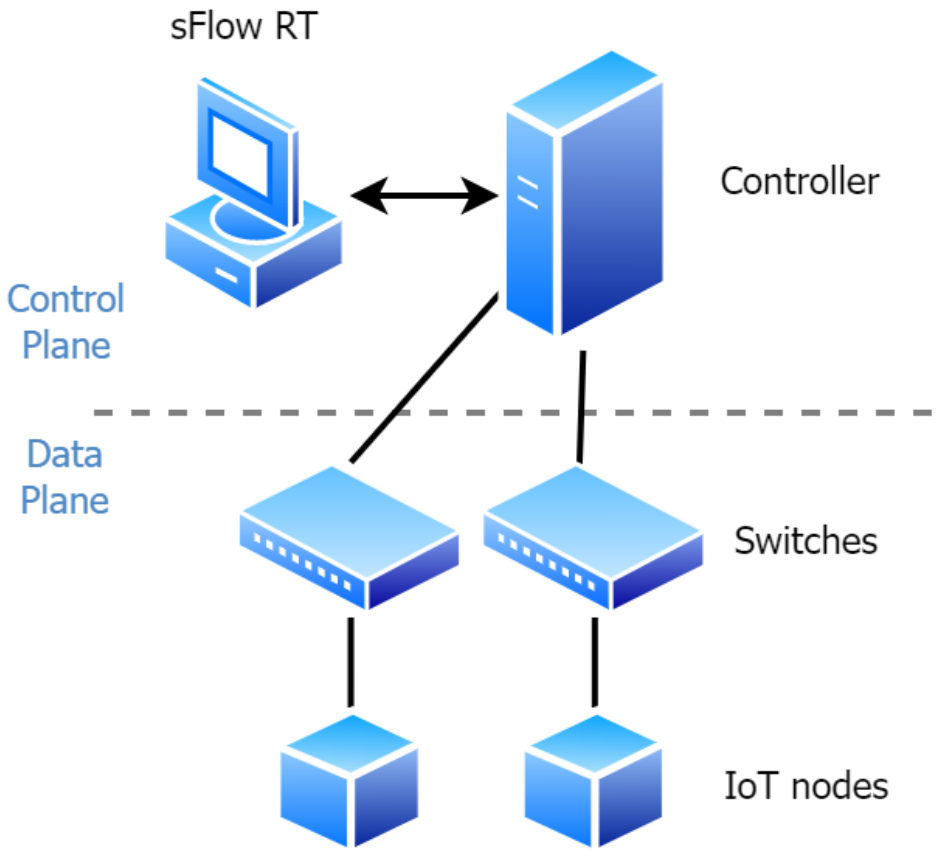
Proposed method

What is sFlow-RT?

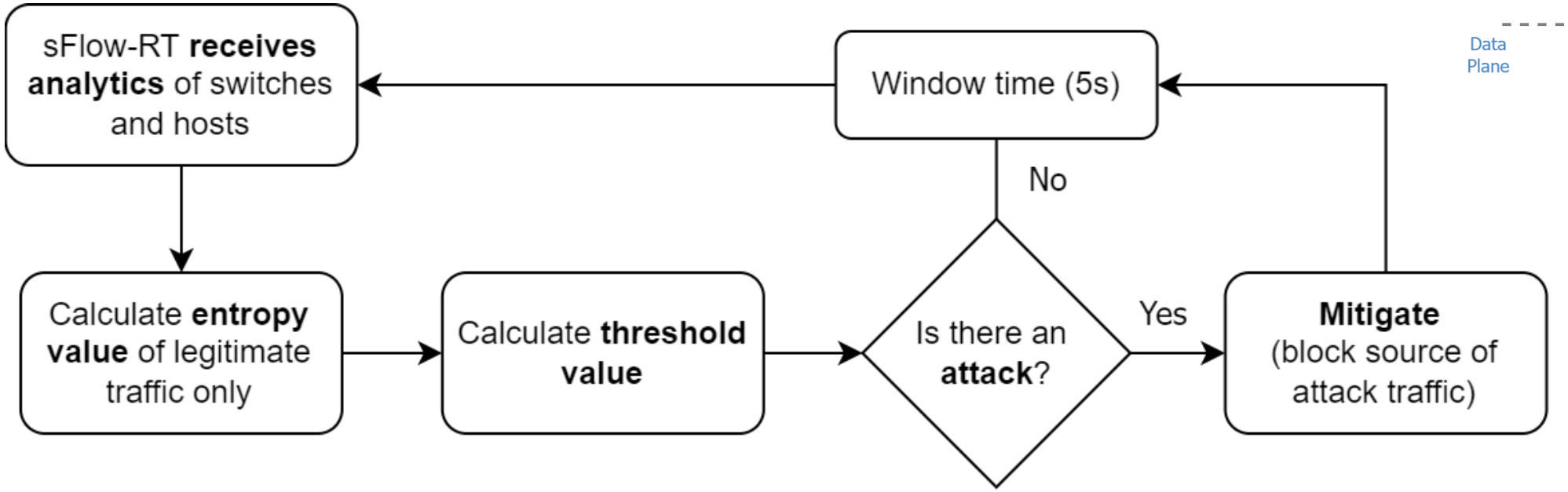
- sFlow-RT is an analytics technology that delivers real-time visibility to SDN.
- Can be used to create applications
- Here sFlow-RT is used to *calculate the entropy of the network*.

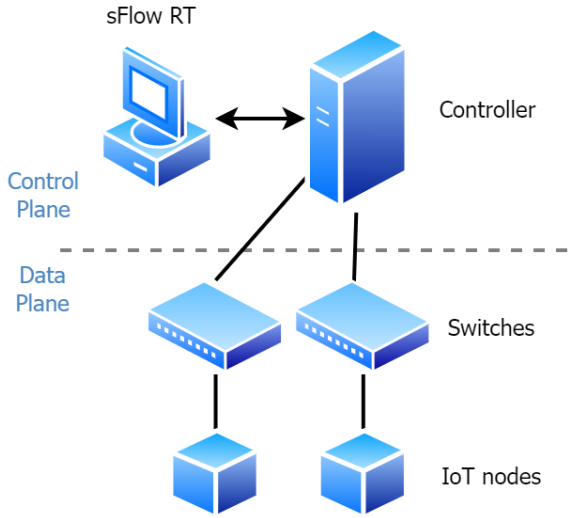
Why use sFlow-RT?

- Number of packets being analyzed do not affect the computational power of the main controller.
- Uses sampling to estimate the number of travelling packets.
  - Applicable to high-speed networks and suitable for handling large flows [1].

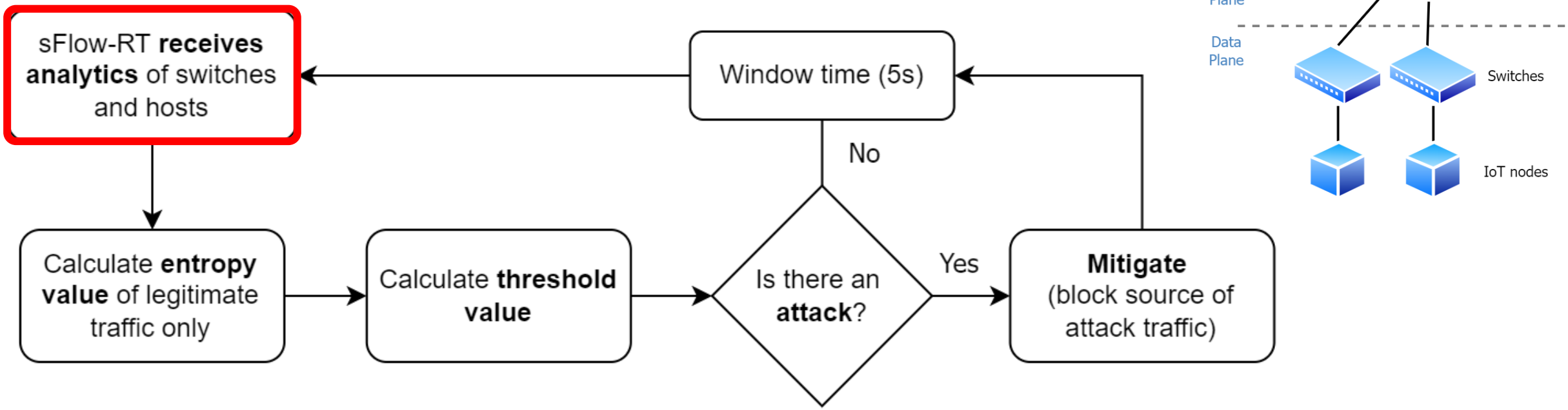


Flowchart


sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)

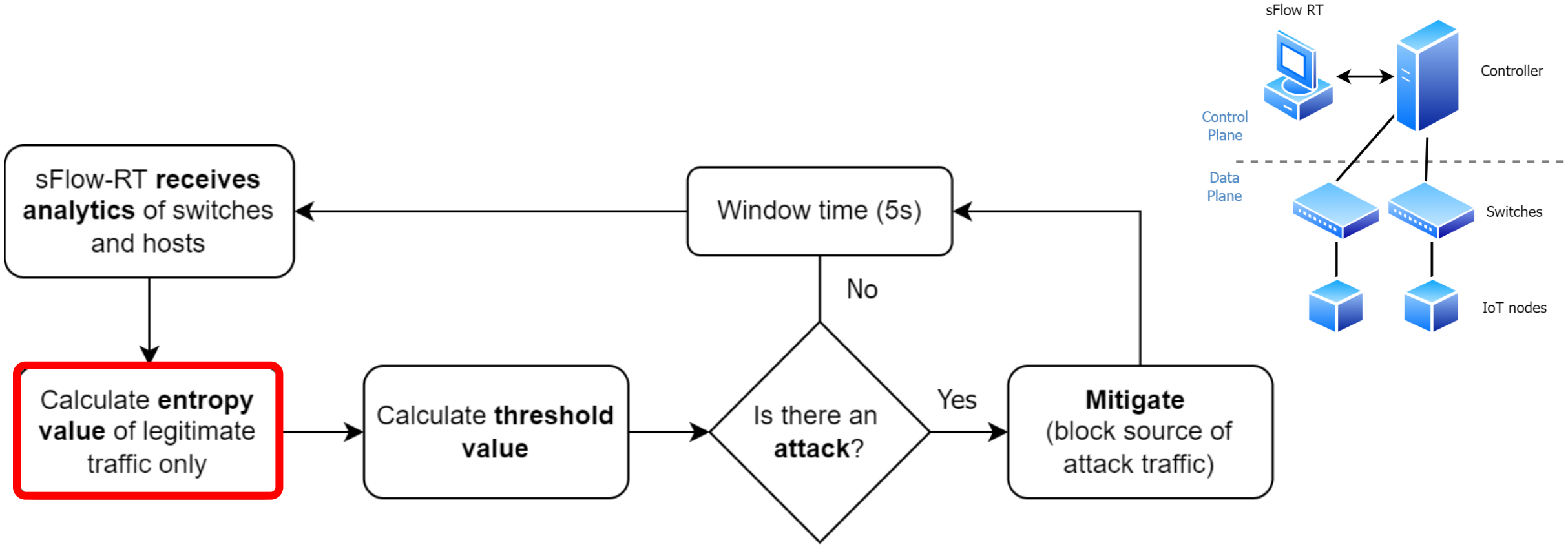

sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

Flowchart

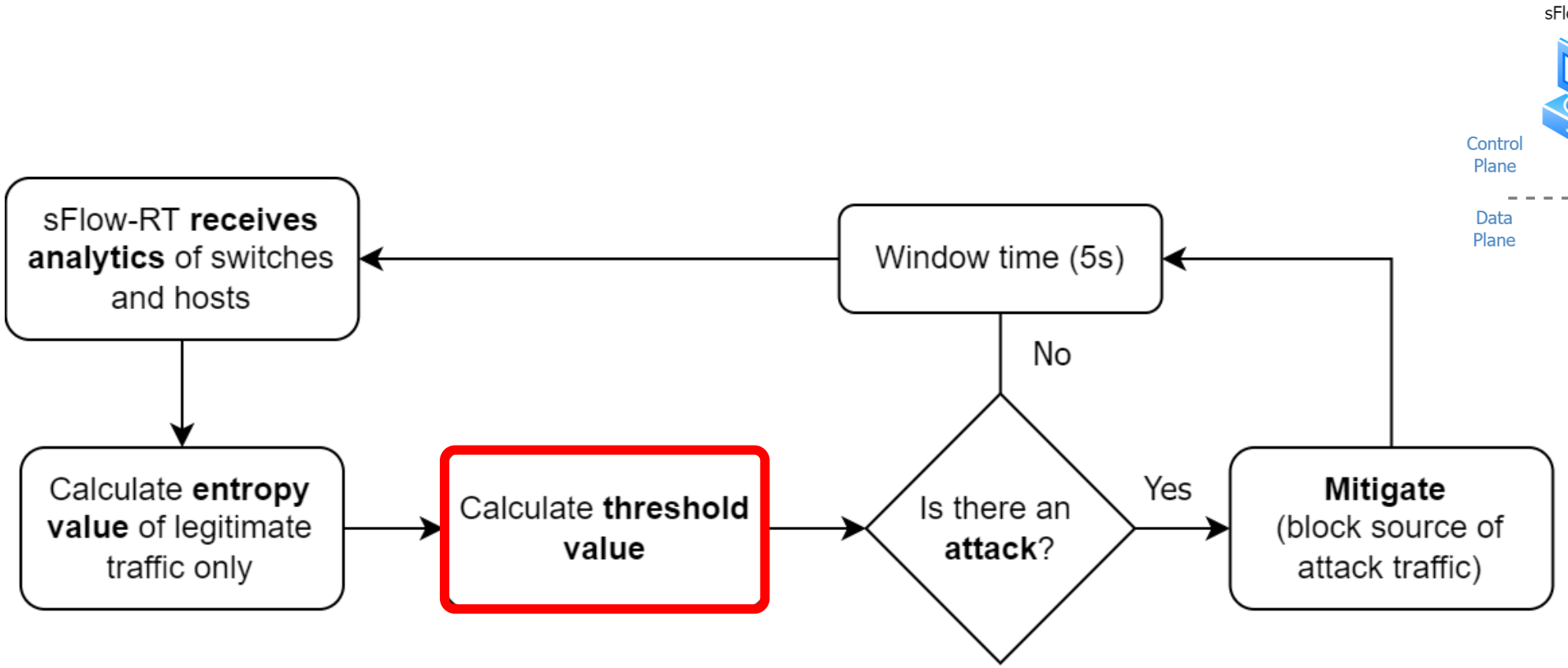

sFlow-RT **receives analytics** of switches and hosts
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)
No
Window time (5s)

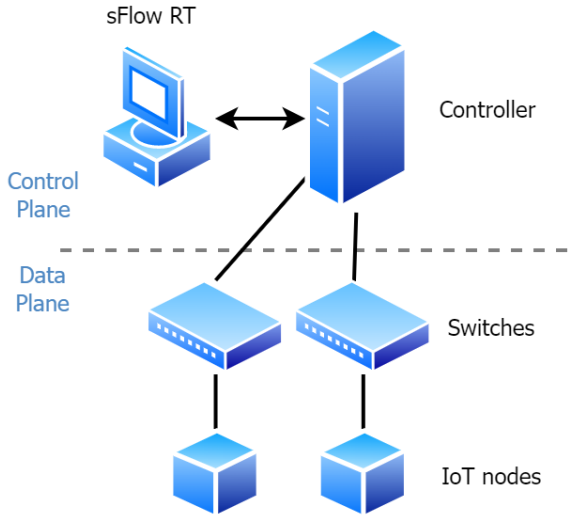
sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

Flowchart

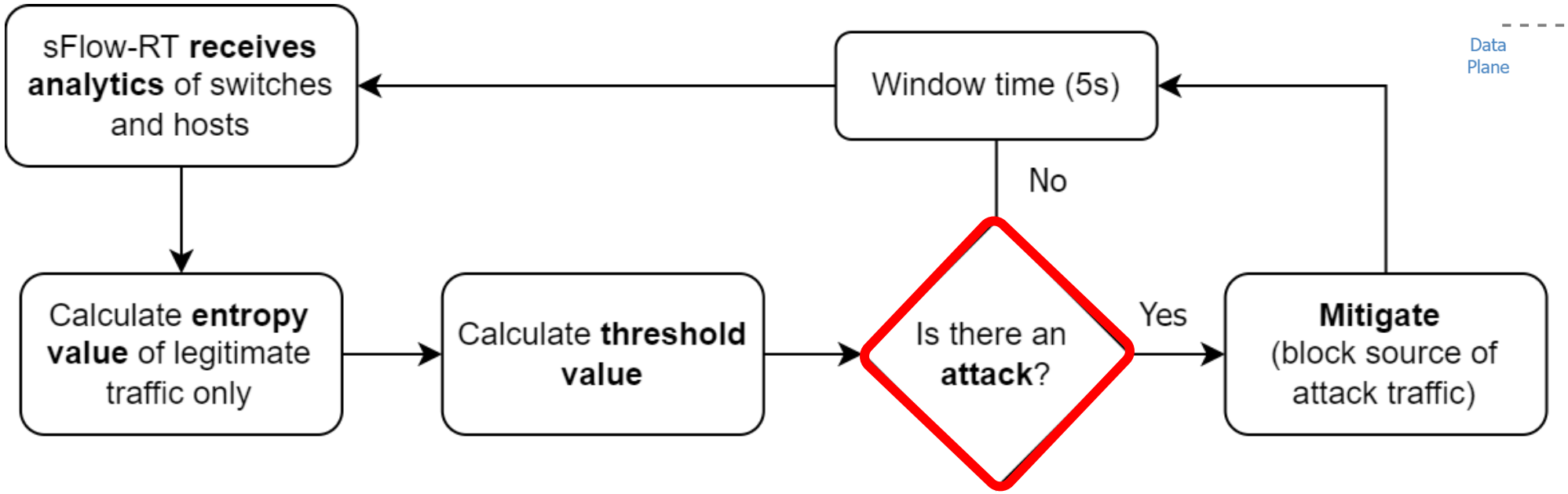

sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)
sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

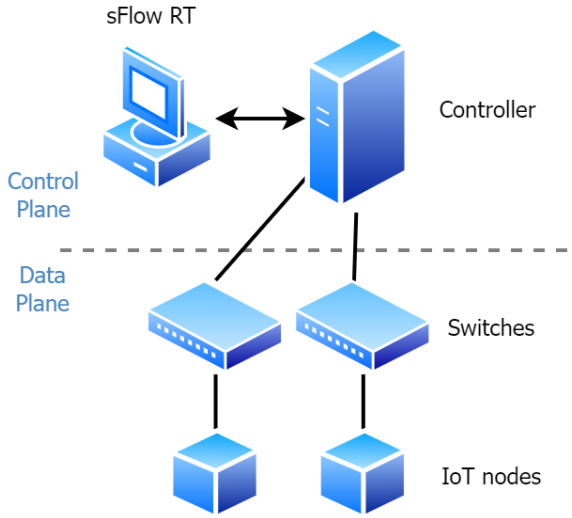
Flowchart


sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)

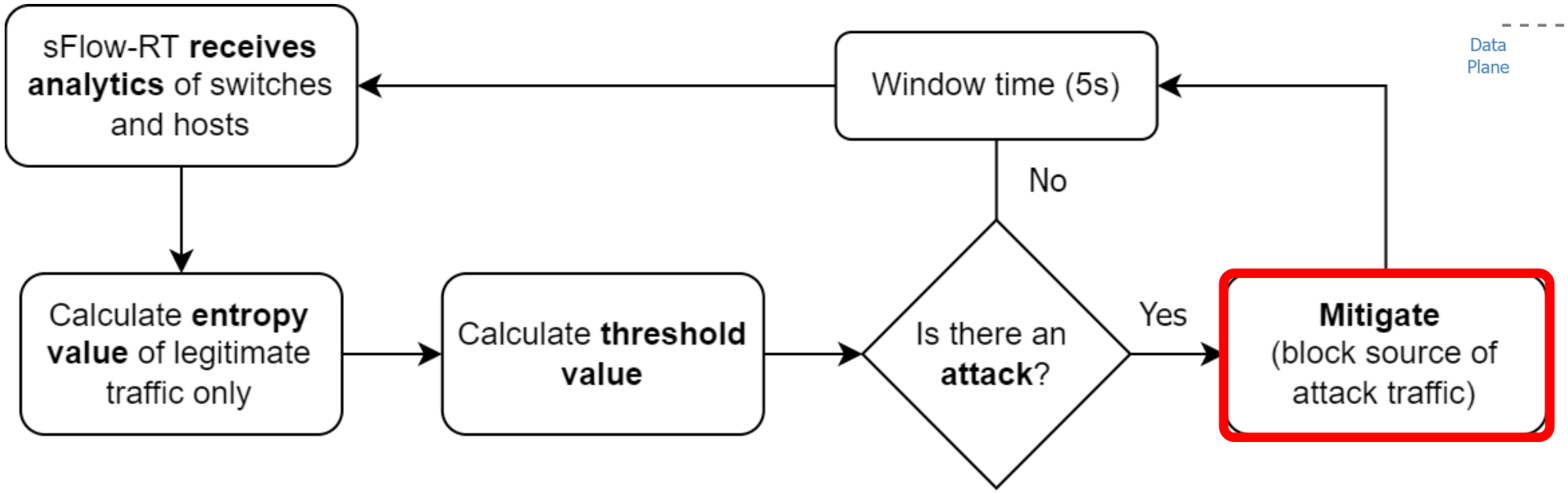

sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

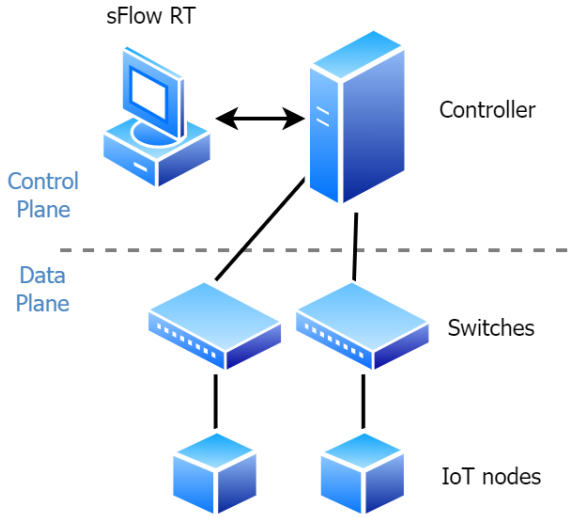
Flowchart


sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)

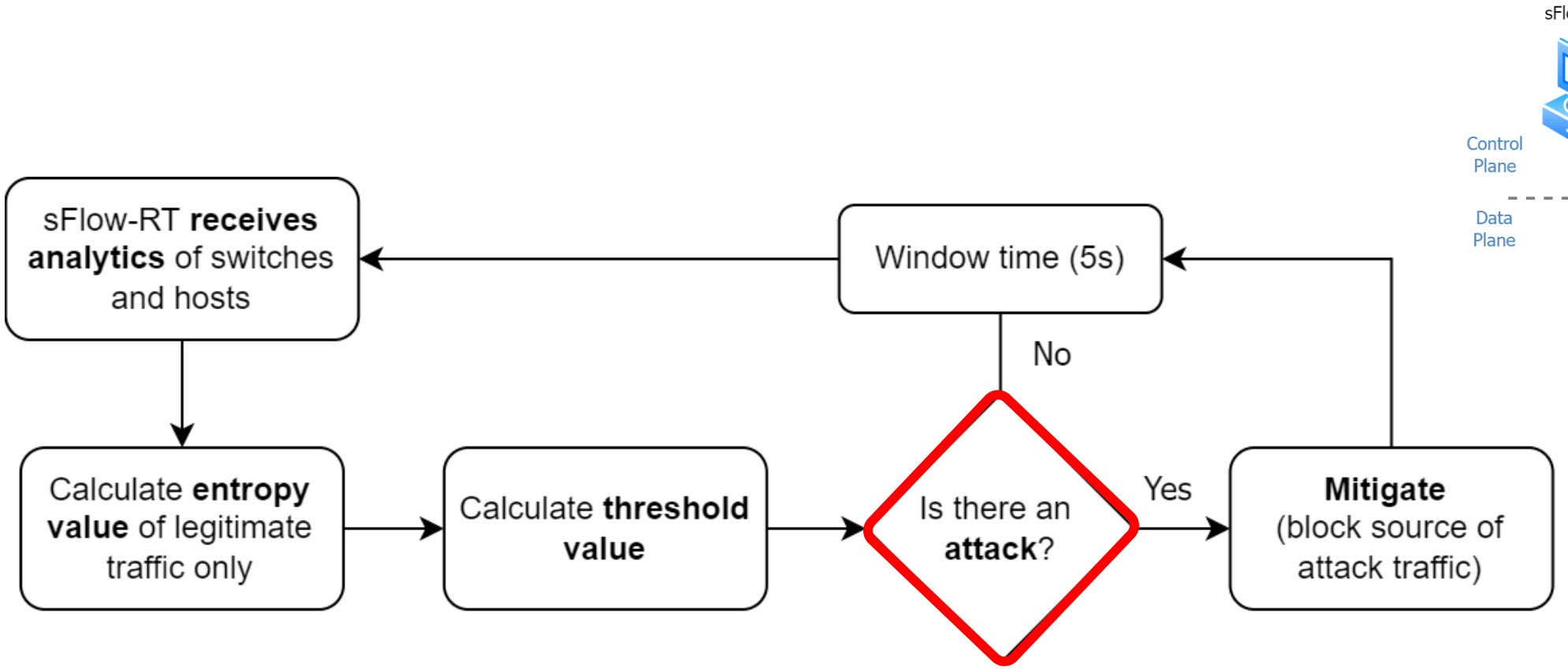

sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

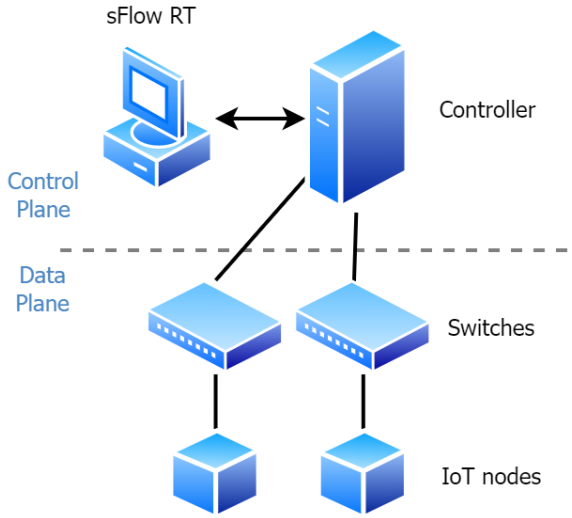
Flowchart


sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)

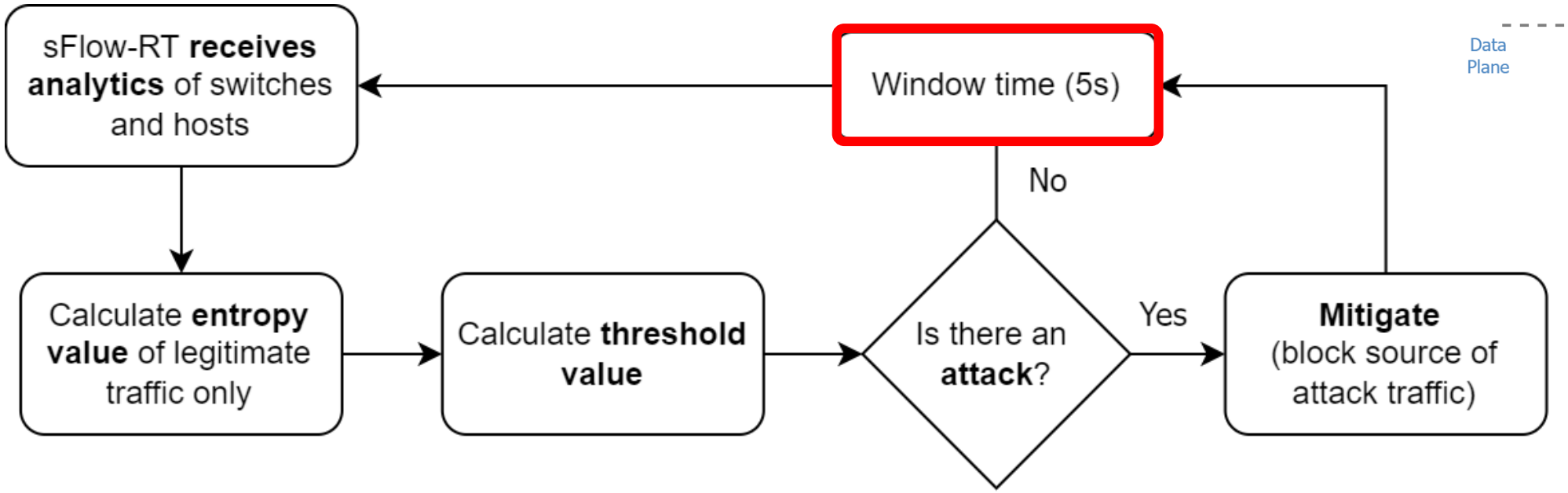

sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

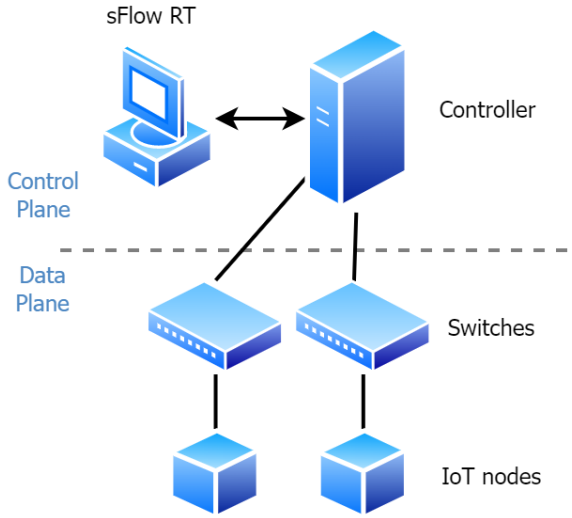
Flowchart


sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)

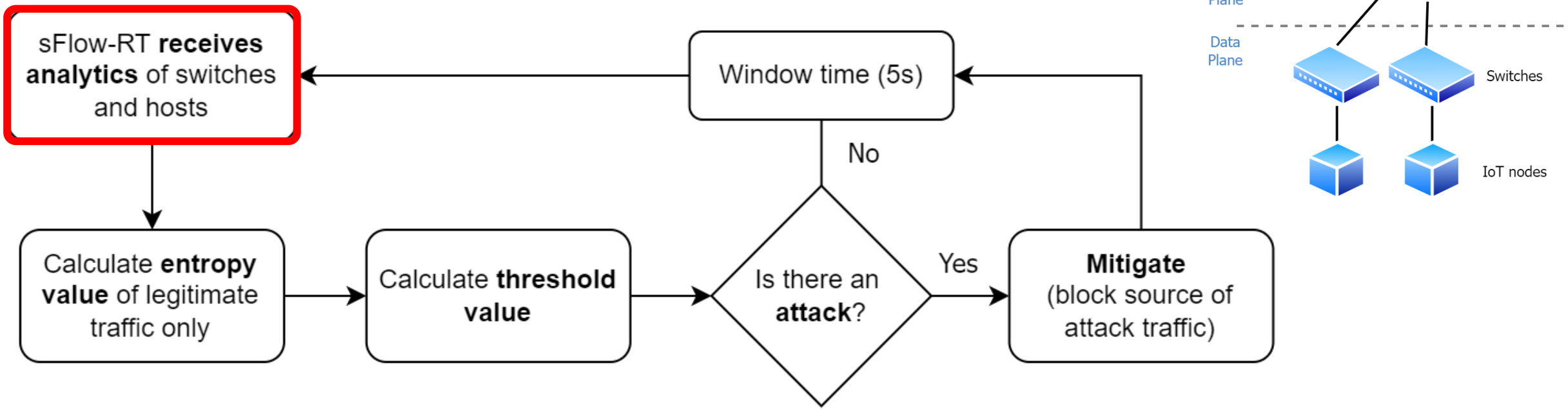

sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

Flowchart


sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)

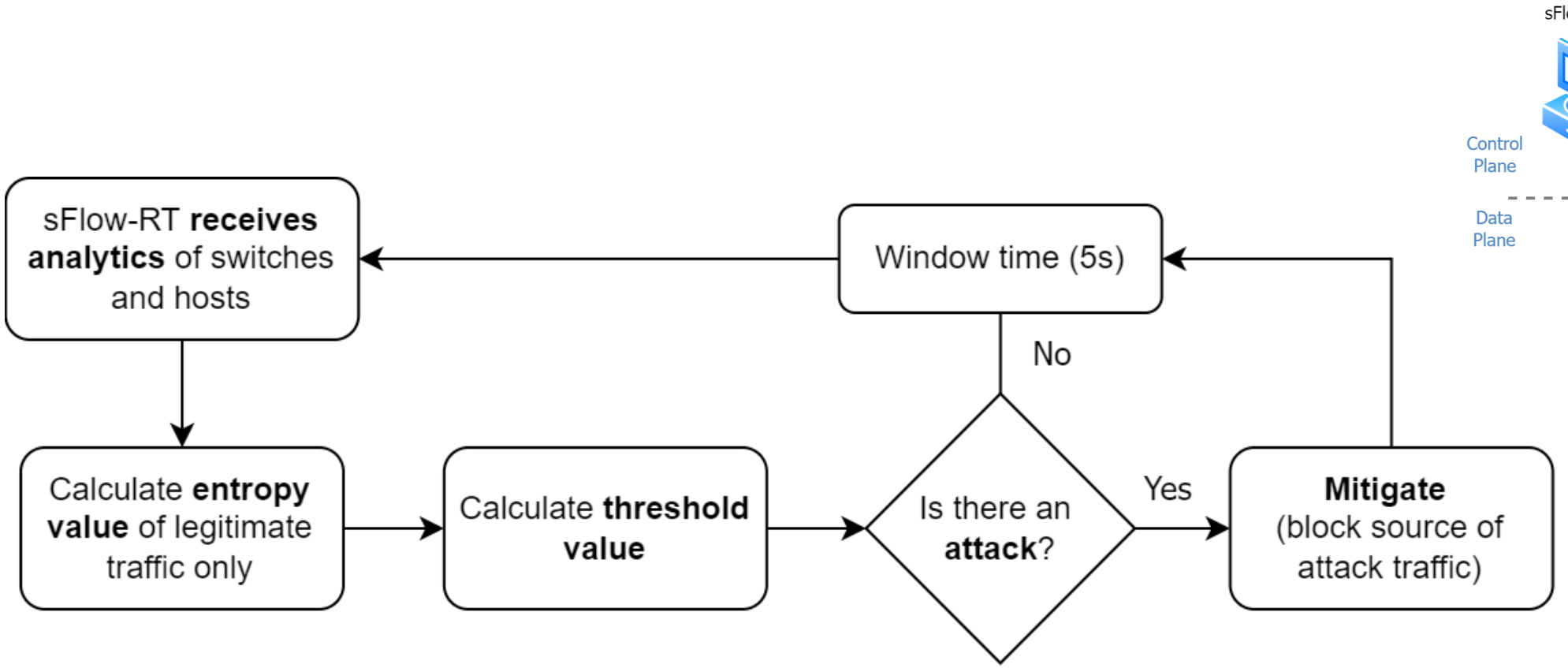

sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

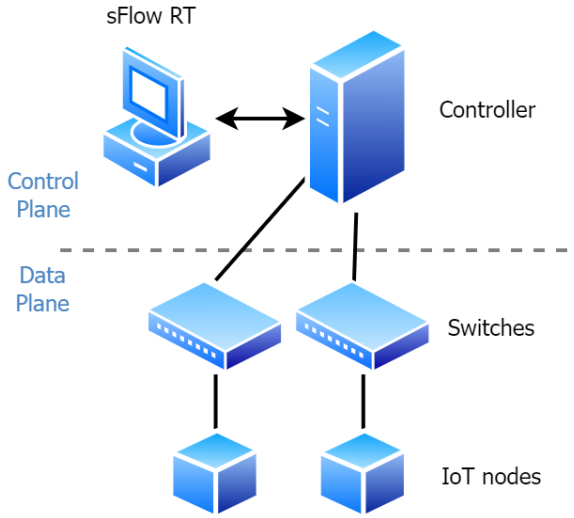
Flowchart


sFlow-RT **receives analytics** of switches and hosts
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
No
Yes
Window time (5s)
Mitigate (block source of attack traffic)

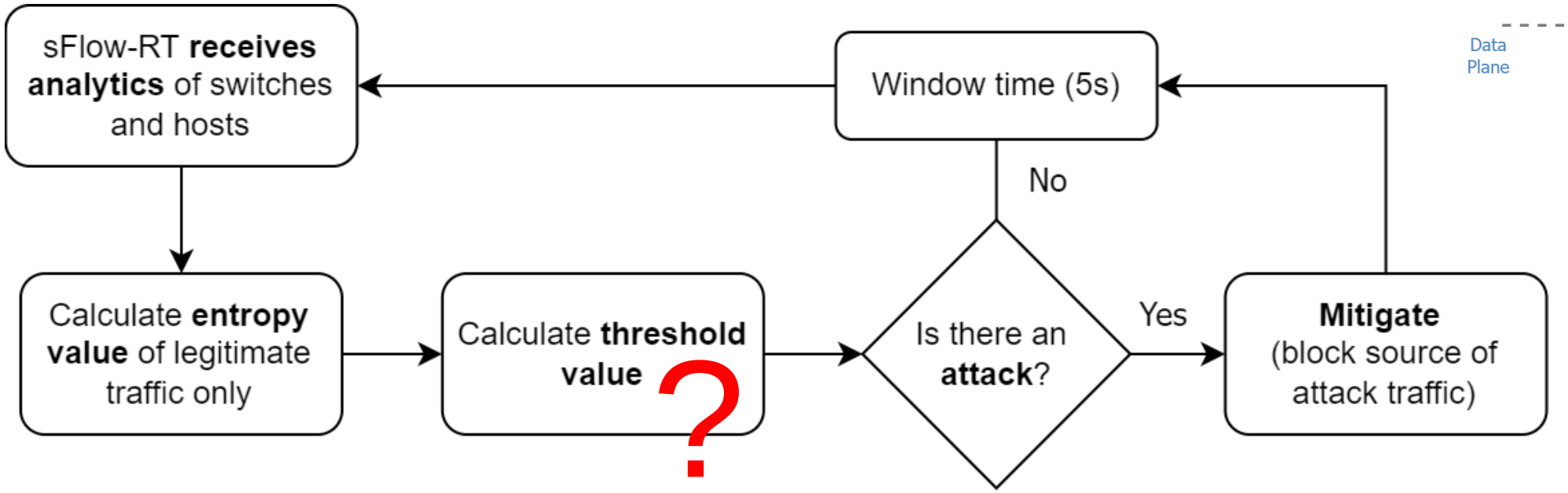
sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

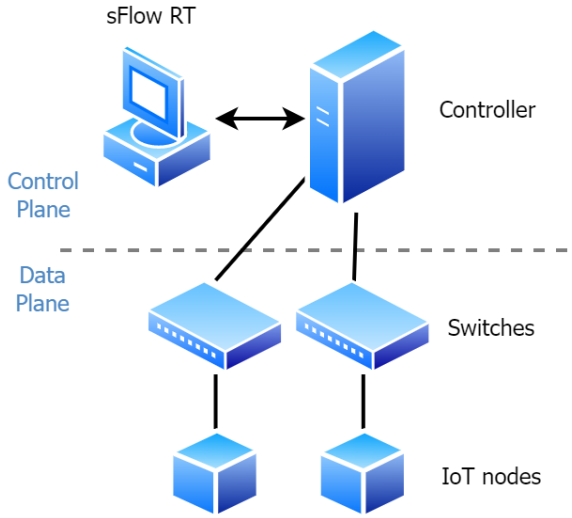
Flowchart


sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)


sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

Flowchart


sFlow-RT **receives analytics** of switches and hosts
Window time (5s)
No
Calculate **entropy value** of legitimate traffic only
Calculate **threshold value**
?
Is there an **attack**?
Yes
Mitigate (block source of attack traffic)


sFlow RT
Controller
Control Plane
Data Plane
Switches
IoT nodes

Setting a threshold

- Threshold is important for *entropy-based detection.*
  - To judge whether the calculated entropy is anomalous or not, it is necessary to set a threshold.
  - If the calculate entropy *drops below the threshold*, then the traffic can be considered as an attack.

- To set a threshold, ***adaptive threshold algorithm*** is used [2].
- What is adaptive threshold algorithm?
  - David et al. [2] suggested a method to calculate the *mean* and *standard deviation* of *previous entropy values* for a period of time and set the threshold base on it.

4. Experimental Results

Experimental results

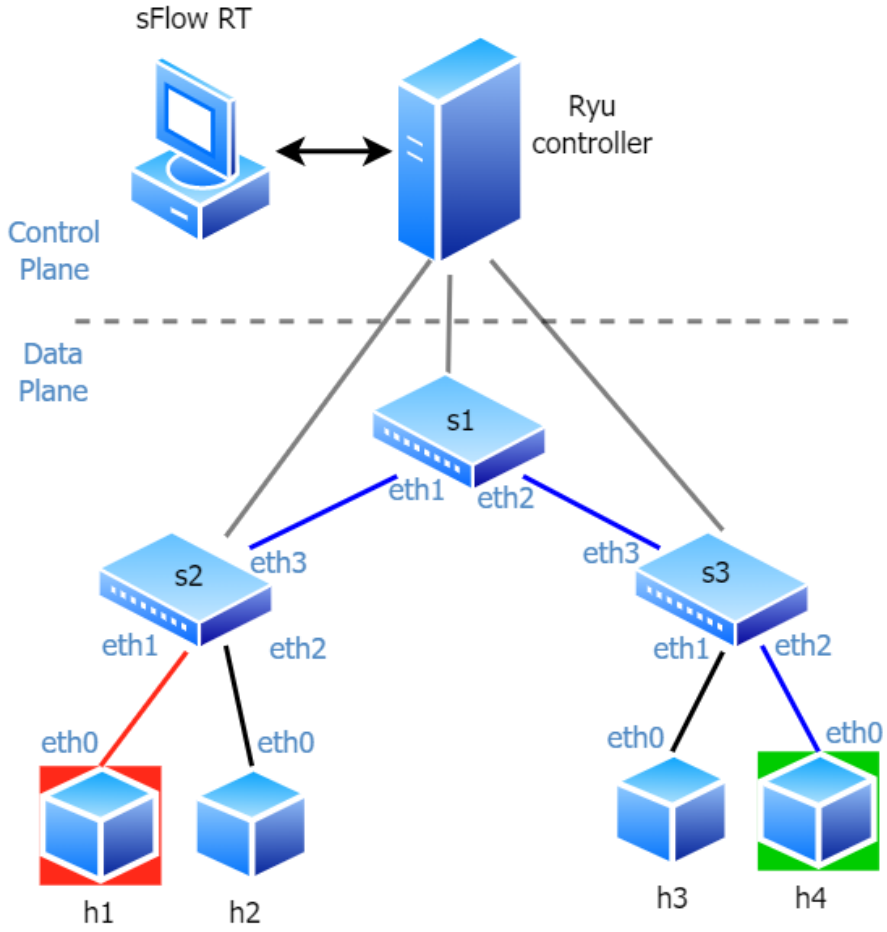
Experiment this proposed method in 2 scenarios

- DoS scenario.
- DDoS scenario.

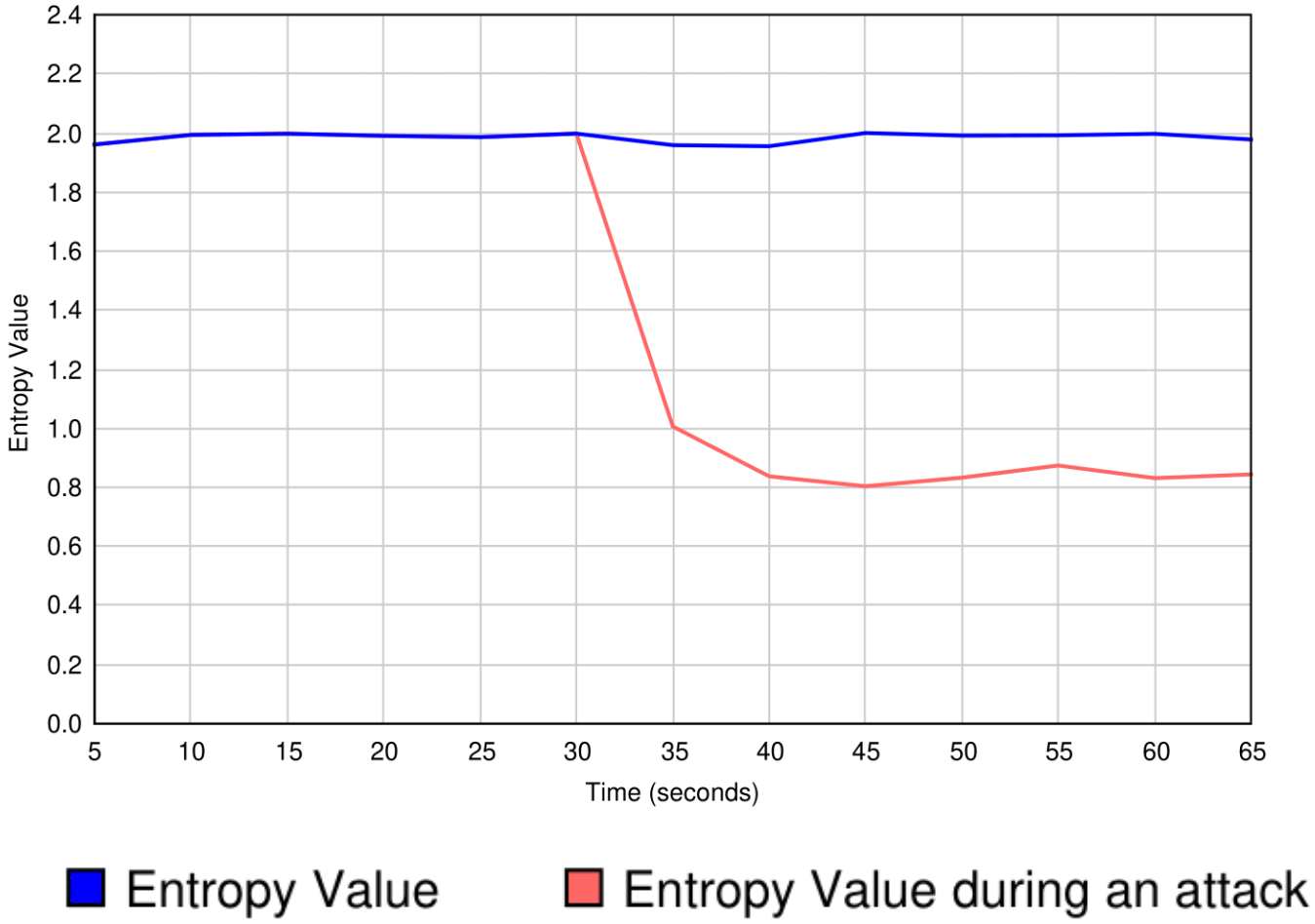
Experimental results (*DoS Scenario*)

DoS scenario

- Red box: attacking host
- Green box: target host
- Red lines: affected connections
- Blue lines: mitigated connections



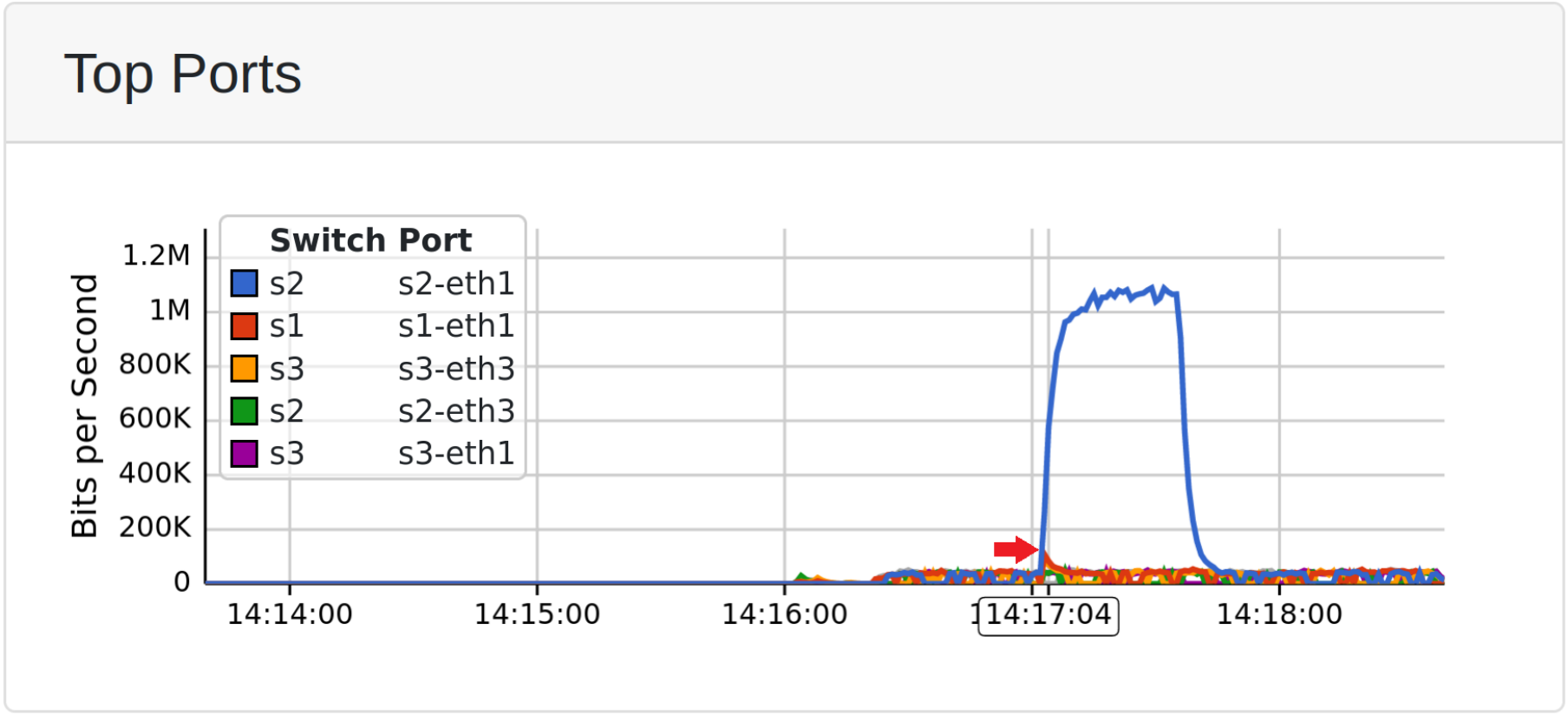
Experimental results (*DoS Scenario*)

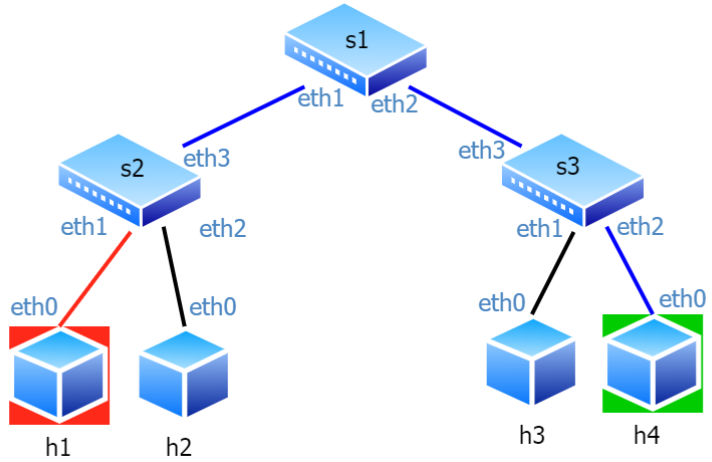


This significant drop alerts the detection that an attack is occurring.

Experimental results (*DoS Scenario*)

Affected ports

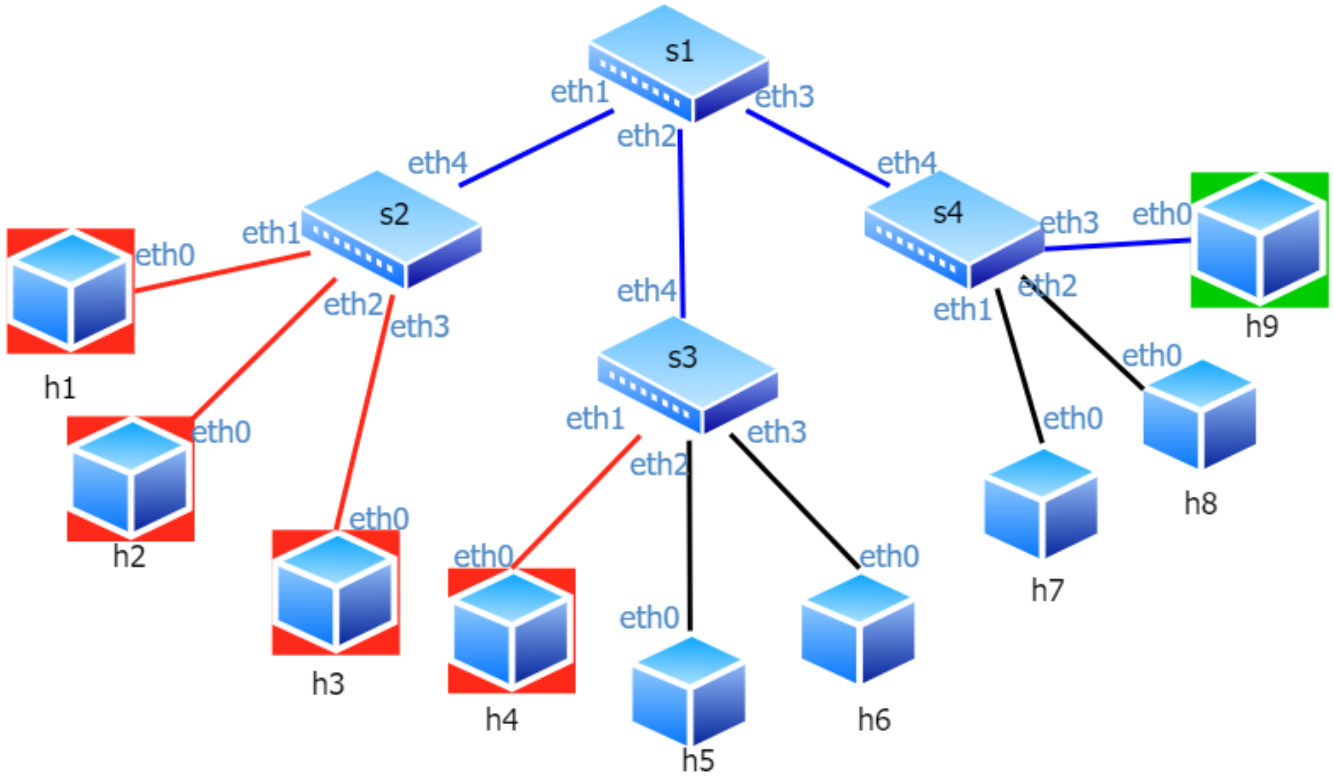




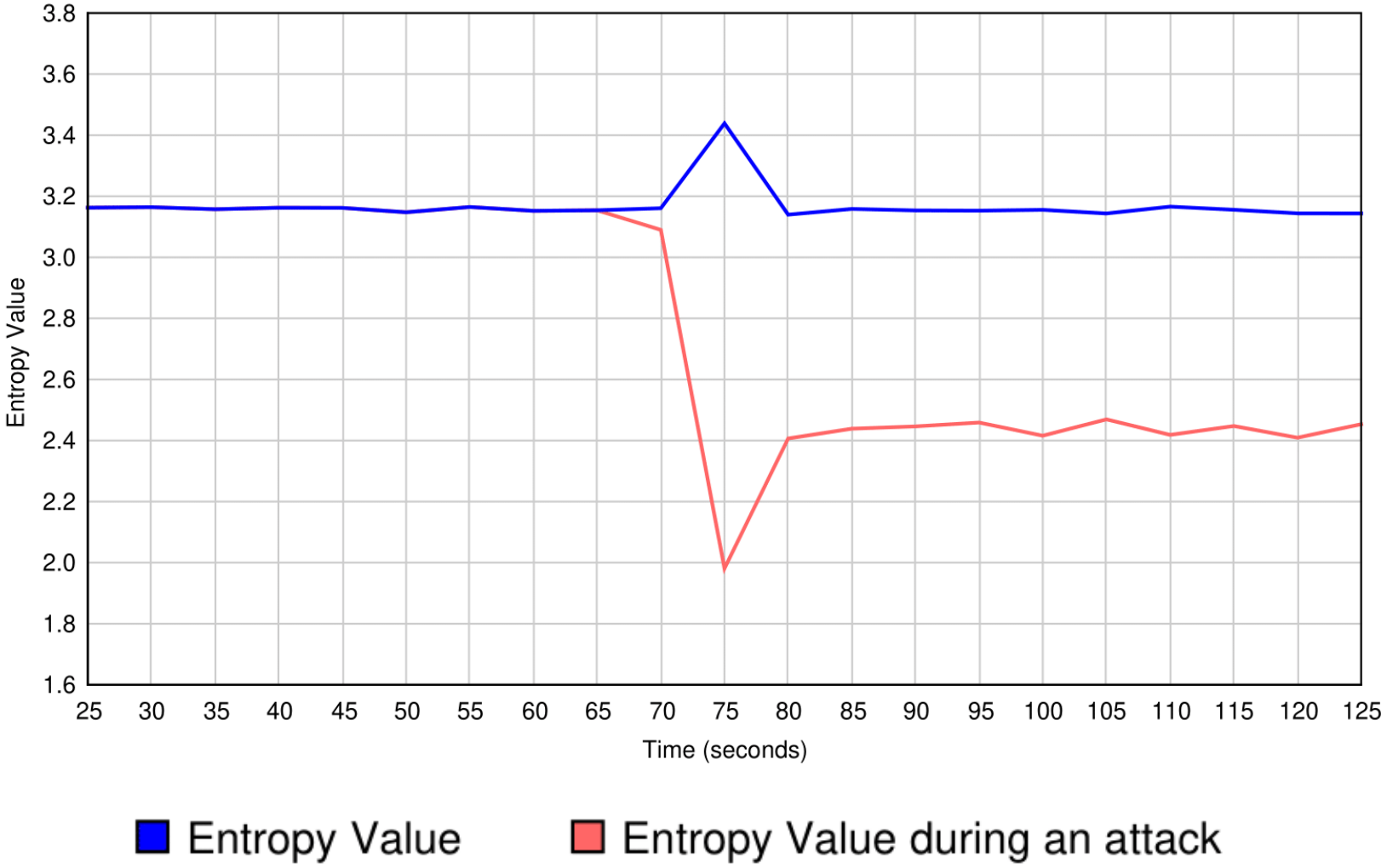
Experimental results (*DDoS Scenario*)

DDoS scenario

- Red box: attacking hosts
- Green box: target host
- Red lines: affected connections
- Blue lines: mitigated connections



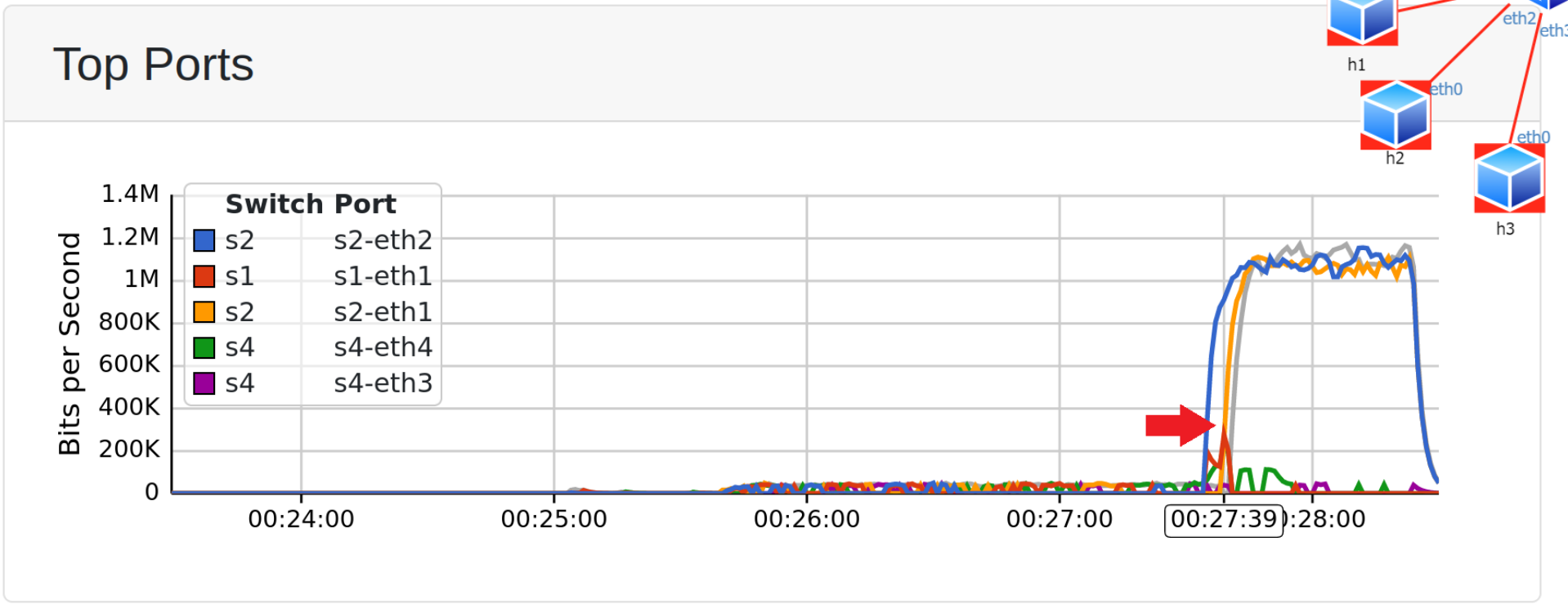
Experimental results (*DDoS Scenario*)

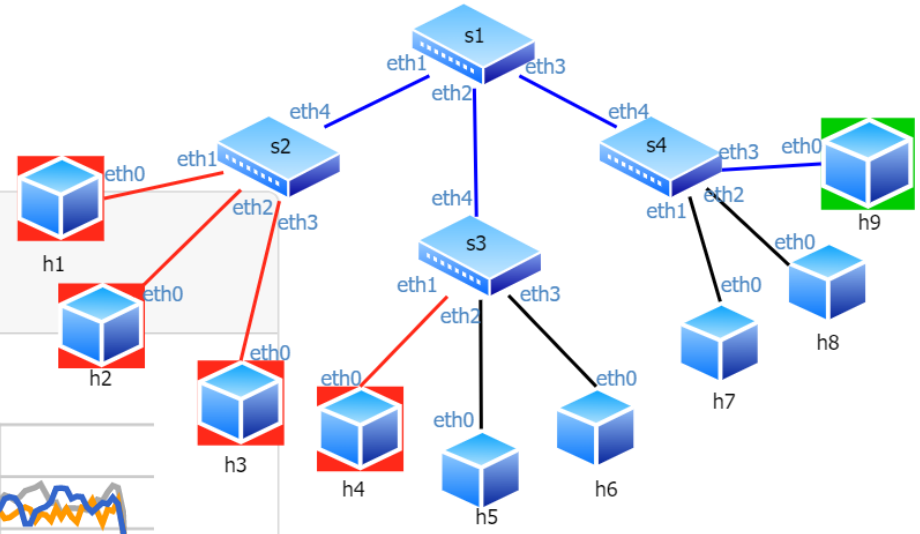


This significant drop alerts the detection that an attack is occurring.

Experimental results (*DDoS Scenario*)

Affected ports





Result comparison

Result comparison of detecting DoS and DDoS attack.

Type of Attack	DoS (1 source of attack)	DDoS (4 sources of attack)
No. of connections made	4	9
Detection speed (no. of time window)	1	1
Difference in Entropy at the start of the attack	0.985	1.175

Conclusion

- Proposes a solution to detecting an incoming DoS/DDoS attack in an SDN-based IoT network.
- Entropy-based detection was proven to be effective. (attacks were detection within 1 time window)
- Attacks that were detected were successfully mitigated (blocked) and did not affect the whole network.

References

[1] E. Jasinska, “sFlow I can feel your traffic,” Amsterdam Internet Exchange, 2006.

[2] J. David and C. Thomas, “DDoS attack detection using fast entropy approach on FLOW- based network traffic,” Procedia Computer Science, vol. 50, pp. 30–36, 2015.

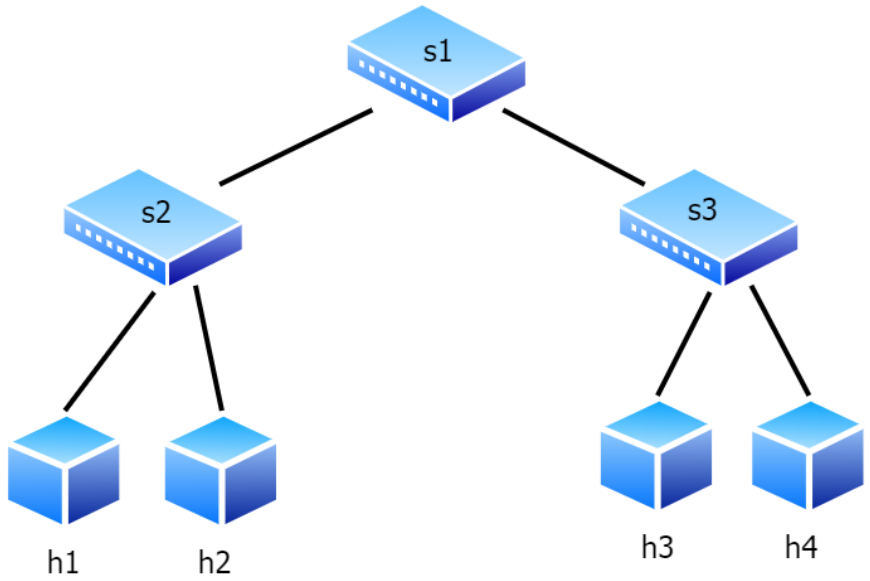
Thank you

THE END

Appendix

Solution : (2) to use entropy-based detection

- Example of an entropy calculation in a network.



At a time interval, there are connections from
h1 → h2 (200 travelling packets)
h2 → h3 (200 travelling packets)
h3 → h4 (200 travelling packets)
h4 → h1 (200 travelling packets)

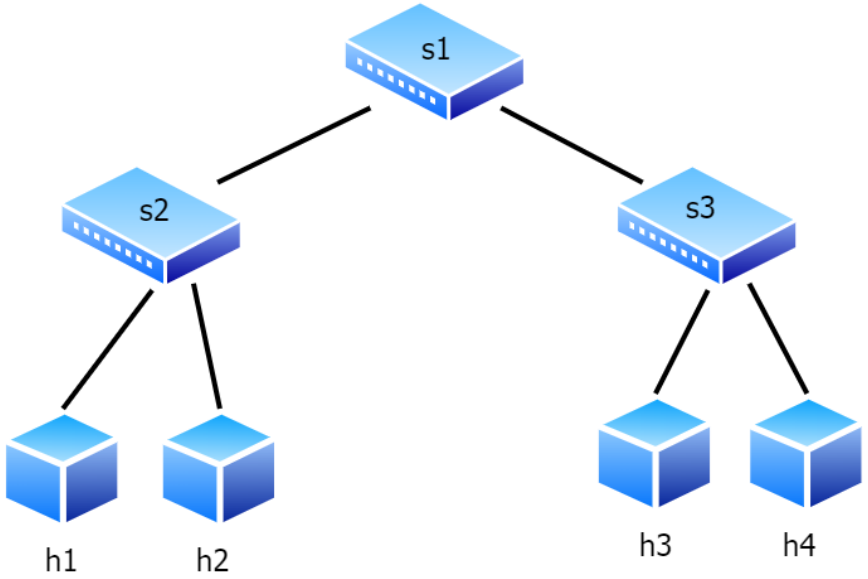
$$H = -\sum_{i=1}^{c} \left(\frac{x_i}{n}\right) \log_2 \left(\frac{x_i}{n}\right)$$

c = total number of connections from hosts
x_i = number of travelling packets from each i^{th} connection
n = total number of travelling packets in the network
H = entropy value

c = 4
$x_1 = 200$, $x_2 = 200$, $x_3 = 200$, $x_4 = 200$
n = 800
H = 2

Solution : (2) to use entropy-based detection

- Example of an entropy calculation in a network.

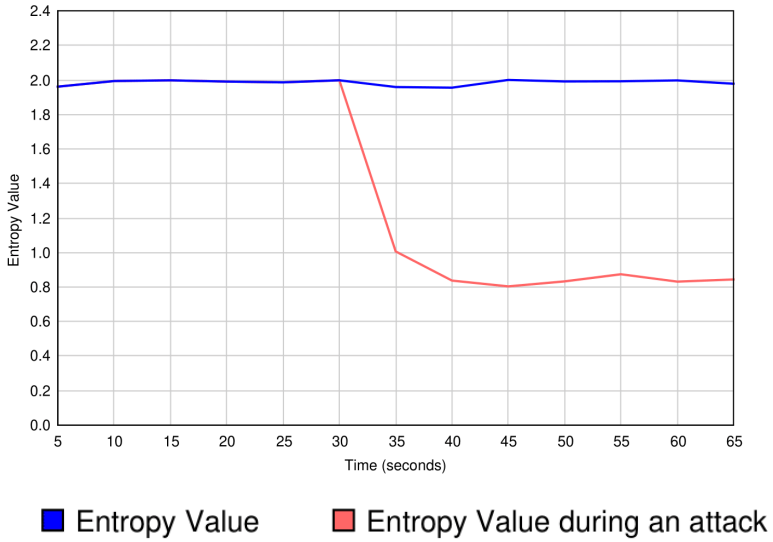


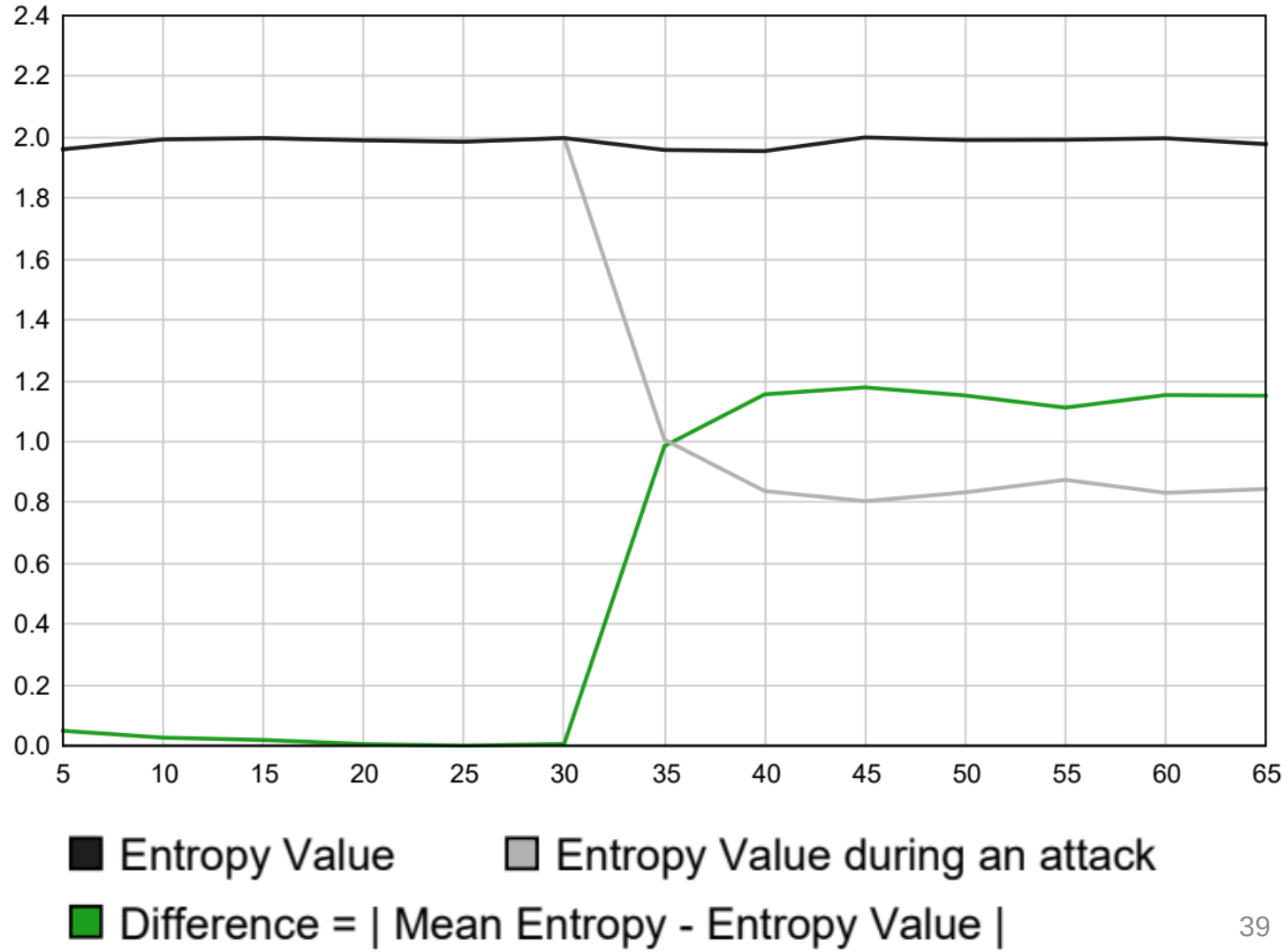
At a time interval, there are connections from
h1 → h2 (200 travelling packets)
h2 → h3 (200 travelling packets)
h3 → h4 (200 travelling packets)
h4 → h1 (200 travelling packets)

$$H = -\sum_{i=1}^{c} \left(\frac{x_i}{n}\right) \log_2 \left(\frac{x_i}{n}\right)$$

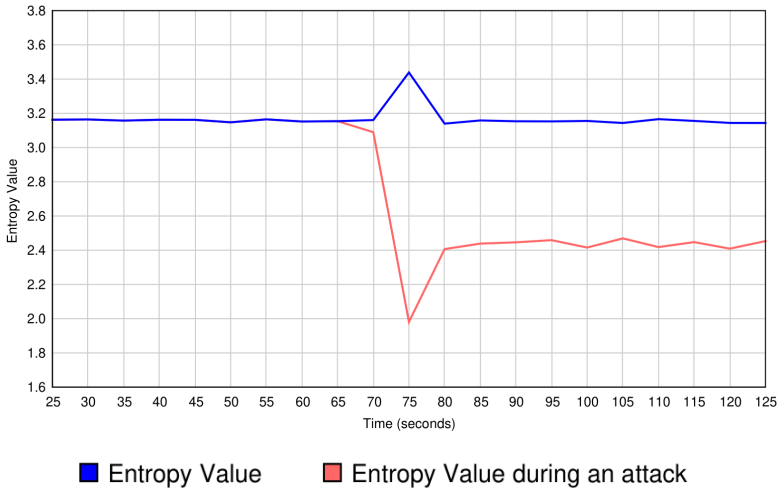
c = 4
$x_1 = 200, x_2 = 200, x_3 = 200, x_4 = 200$
n = 800
H = 2

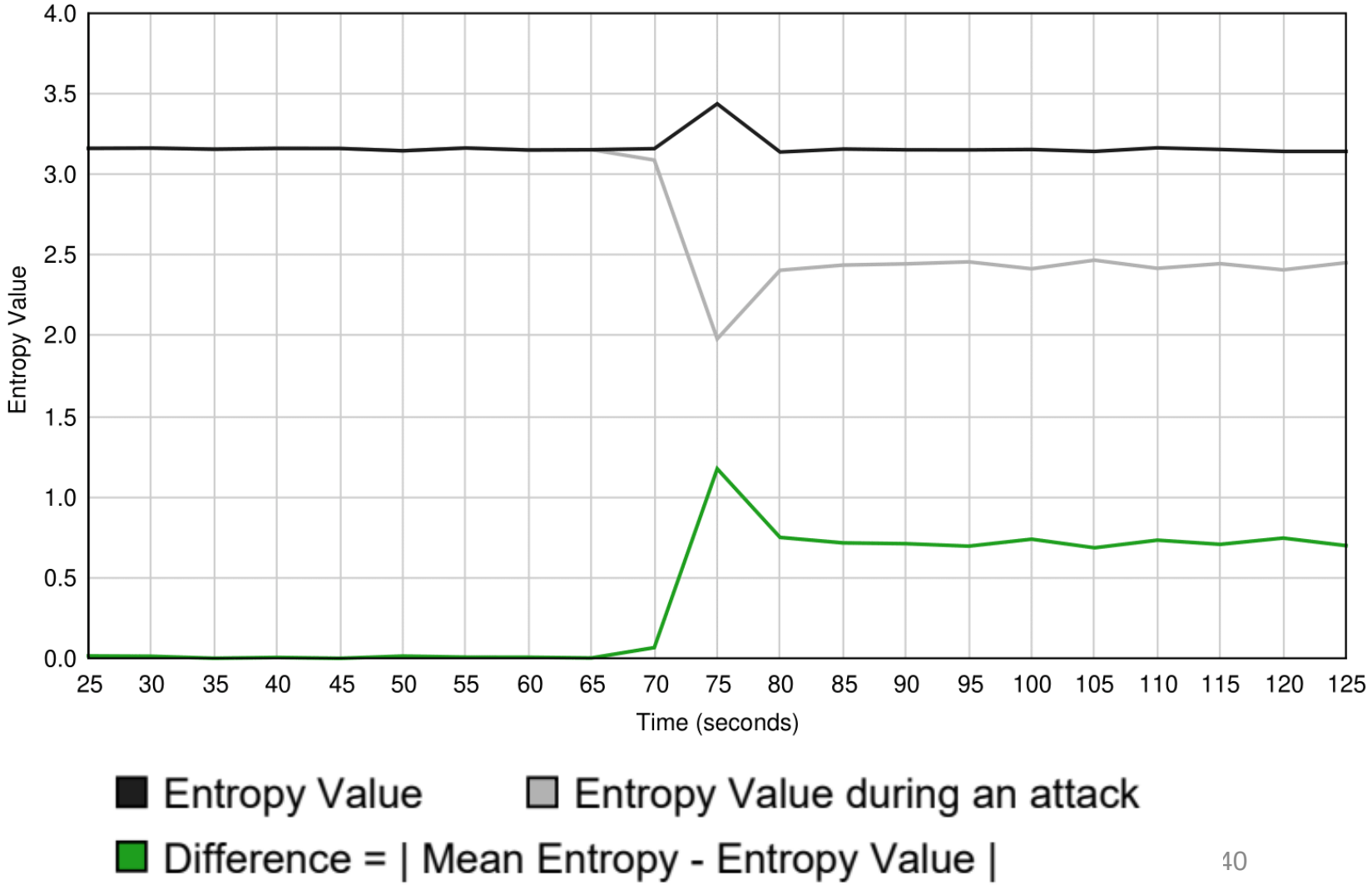
Experimental results (*DoS Scenario*)





Experimental results (*DDoS Scenario*)





How to determine the threshold [2]

Why adaptive threshold algorithm?

- Where the threshold of the detection is updated according to the state of the traffic.
- It is suitable for detecting small and stealthy attack.

Entropy in DoS/DDoS

- Anomalous traffic are usually created from different spoofed source IPs.
- The higher the randomness of the traffic, the higher the entropy.
- Conversely, the lower the randomness of the traffic (with the redundant appearance of single source IP) the lower the entropy.

Solution : (2) to use entropy-based detection

- What is entropy-based detection?
  - In information theory, entropy can be used to measure the uncertainty of information.
  - calculation can be done with Shannon entropy

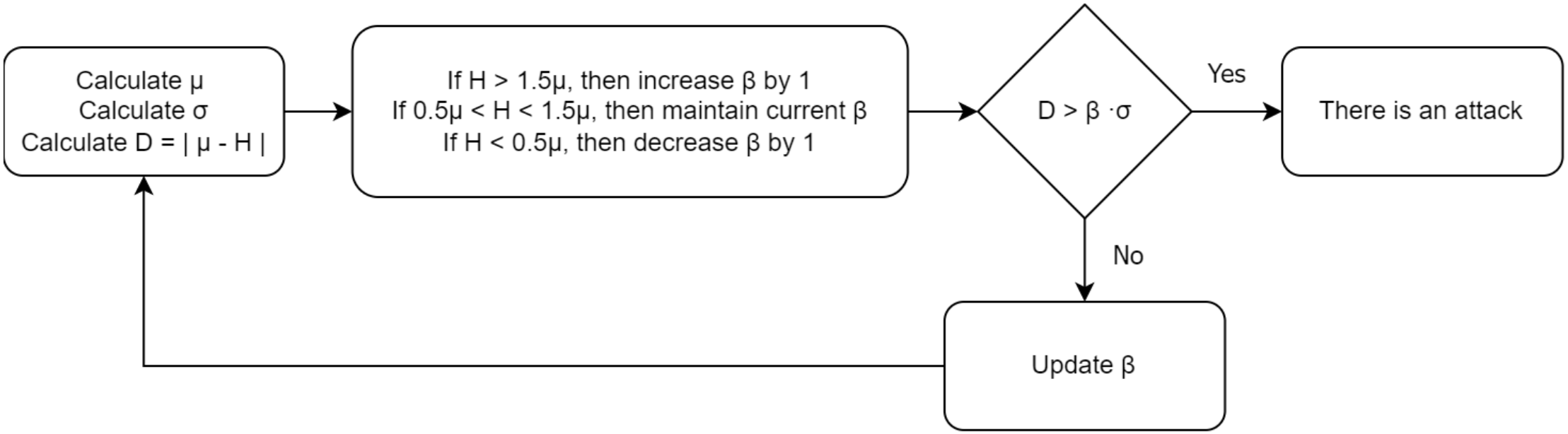
$$H = -\sum_{i=1}^{n} p_i \log_2 p_i$$

Where there is an information source,
n = independent symbols
p_i = probability of each n
H = entropy value

$$H = -\sum_{i=1}^{c} \left(\frac{x_i}{n}\right) \log_2 \left(\frac{x_i}{n}\right)$$

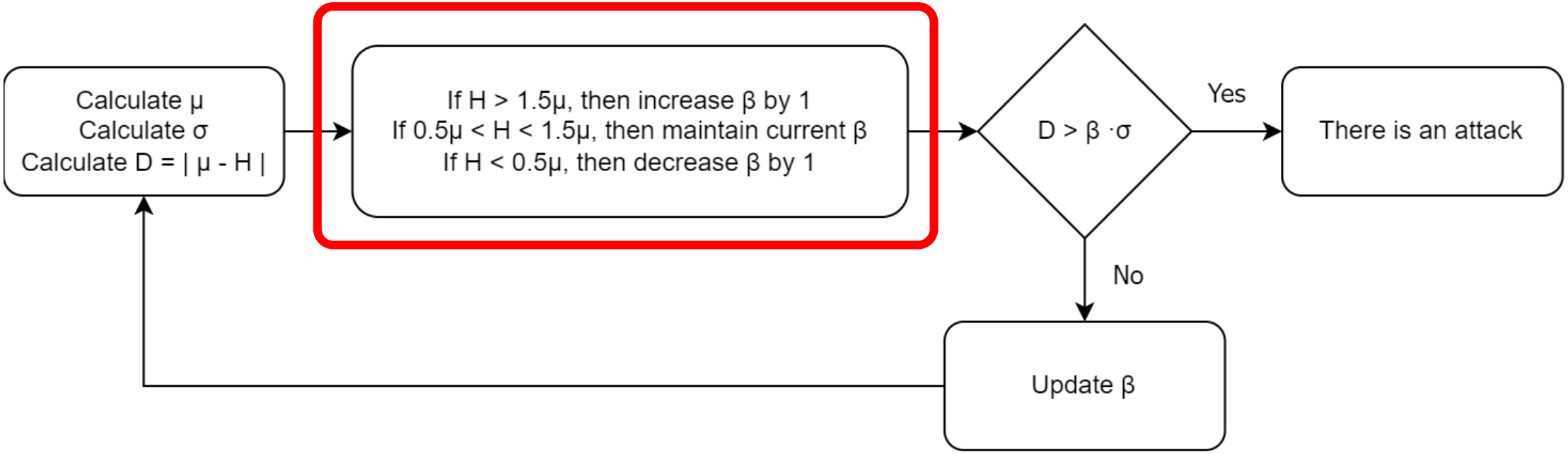
Where there is an information source,
c = total number of connections from hosts
x_i = number of travelling packets from each i^{th} connection
n = total number of travelling packets in the network
H = entropy value

How to determine the threshold [2]



H = current entropy value
μ = mean of previous entropies
σ = standard deviation of previous entropies
D = | μ - H |
β = threshold multiplication factor

How to determine the threshold [2]



H = current entropy value

μ = mean of previous entropies

σ = standard deviation of previous entropies

D = | μ - H |

β = threshold multiplication factor