Leveraging In-band Network Telemetry for Automated DDoS Detection in Production Programmable Networks: The AmLight Use Case

Hadi Sahin, Jeronimo Bezerra, Italo Brito, Renata Frez, Vasilka

Chergarova, Luis Fernandez Lopez, Julio Ibarra

November 11, 2024

Center for Internet Augmented Research & Assessment

Florida International University
Division of Information Technology

Table of Contents

Motivation

Introduction

- In-band Network Telemetry (INT) has been available since 2015, providing rich network state information.
- While INT holds promise for enhanced network monitoring and security applications, research and practical deployments of INT for Distributed Denial of Service (DDoS) threat detection remain limited:
  - Existing studies primarily rely on data generated from simulation environments (e.g., Mininet), lacking real-world validation.
  - There is a lack of comparative analysis among different network monitoring tools, such as the performance and accuracy of INT-based approaches versus traditional sFlow-based monitoring.

FIU

Key Contributions of This Paper

Introduction

In this work, we leverage the In-band Network Telemetry (INT) technology implemented in the AmLight network to enhance Distributed Denial of Service (DDoS) attack detection:

- Utilize real-world production INT data to detect and characterize DDoS attacks.
- Compare the DDoS attack predictions from INT-based analysis with those from traditional sFlow-based monitoring.
- Propose an automated, machine learning-driven approach for robust and accurate DDoS attack detection.

Table of Contents

In-band Network Telemetry (INT) and sFlow

Background and Related Work

- INT technology combines data packet forwarding with network measurement.
- It embeds telemetry information into packets as they traverse the network
- sFlow captures and samples packets across network devices.
- The sFlow agent collects data from switches and routers, and the sFlow collector processes this data.

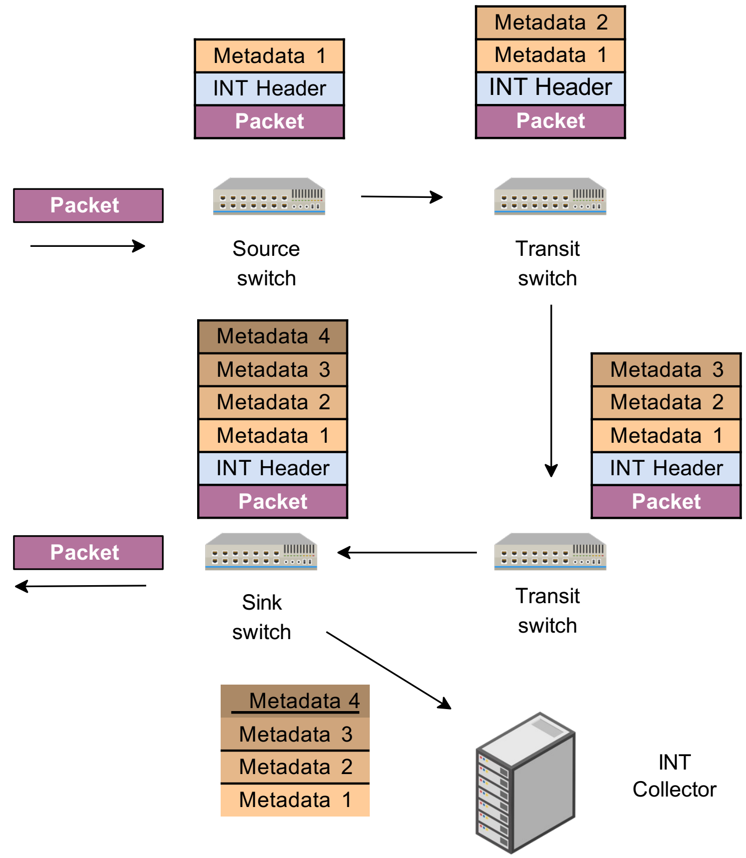
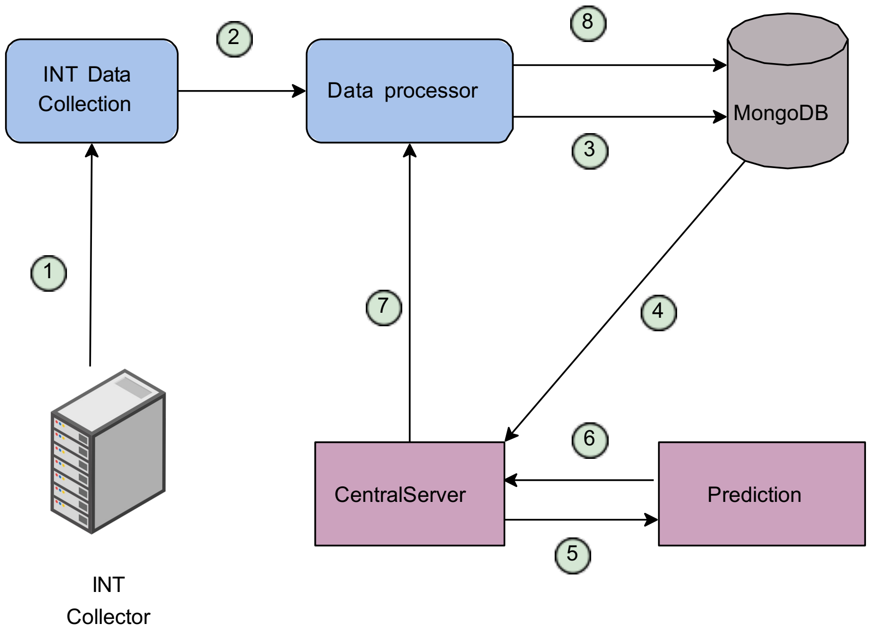


Table of Contents

Automated DDoS Detection

Proposed Mechanism



1. Gather INT data.
2. Send INT data to the *Data processor*:
   - Flow ID: src/dst IP, src/dst ports, protocol.
   - Flow-level features (e.g., *Packets per second*, *Flows per second*).
3. Save processed data to the database.
4. Retrieve processed data.
5. Send data to the prediction model.
6. Receive predictions.
7. Send predictions to the *Data processor* for aggregation.

Table of Contents

Evaluation Metrics

Experimental Evaluation

True Positives (TP), True Negatives (TN), False Positives (FP), and False Negatives (FN):

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{F1-score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Confusion matrix: a 2×2 table of actual and predicted Positives (P) and Negatives (N)

Data Source

Experimental Evaluation

- Data were collected from a subnet of a AmLight network from June 6 to June 11, 2024
- We also simulated various attack types

Attack Type	Date	Attack Episode
SYN Scan	06.10.2024	13:24:02 - 13:57:03
SYN Scan	06.10.2024	16:30:51 - 16:35:20
UDP Scan	06.10.2024	16:36:20 - 16:53:00
UDP Scan	06.10.2024	16:56:45 - 16:59:99
SYN Flood	06.10.2024	20:48:01 - 20:49:01
SYN Flood	06.10.2024	20:52:11 - 20:54:12
SYN Flood	06.11.2024	20:13:31 - 20:15:31
SYN Flood	06.11.2024	20:16:41 - 20:17:01
SYN Flood	06.11.2024	20:17:17 - 20:17:37
SlowLoris	06.11.2024	20:27:37 - 20:28:37
SlowLoris	06.11.2024	20:29:12 - 20:31:12

Feature Selection

Experimental Evaluation

Features	INT	sFlow
Protocol	√	√
Packet Size*	√	√
Number of packets	√	√
Queue Occupancy*	√	×
Hop Latency*	√	×
Inter Arrival Time*	√	√
Flow rate (Gbit/s)	√	√
Packet rate (Packet/s)	√	√

- * Includes packet-level, cumulative, average, and standard deviation of the variables.
- The cumulative inter-arrival time denotes flow duration.

Machine Learning Models

Experimental Evaluation

We employ the following machine learning (ML) models for DDoS attack detection:

- Random Forest (RF)
- K-Nearest Neighbors (KNN)
- Gaussian Naive Bayes (GNB)
- Neural Network (NN) with three hidden layers of 32, 16, and 8 neurons

To train the ML models, we use a 90:10 train-test split ratio, reserving 10% of the data for model evaluation.

DDoS Predictions Using INT vs sFlow Data

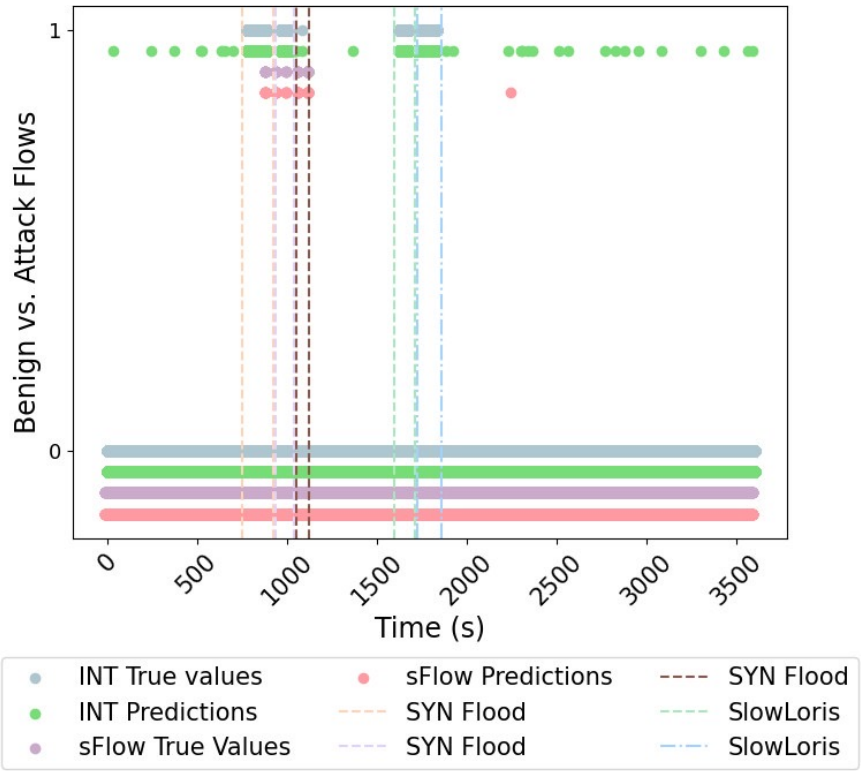
Experimental Evaluation

- We use data flows from June 11, 2024 as the test set to evaluate the models.
- We consider the *SlowLoris* attack as a zero-day scenario, where the models have not been trained on this specific attack type.

Data	Model	Accuracy	Recall	Precision	F1-score
INT	RF	1.0000	1.0000	0.9999	1.0000
sFlow	RF	0.9999	1.0000	0.9907	0.9953
INT	GNB	0.9919	1.0000	0.9959	0.9959
sFlow	GNB	0.9959	1.0000	0.6057	0.7544
INT	KNN	0.9988	0.9993	0.9984	0.9988
sFlow	KNN	0.9997	1.0000	0.9550	0.9770
INT	NN	0.9996	1.0000	0.9992	0.9996
sFlow	NN	0.9937	0.0000	0.0000	0.5000

A Closer Look at Predicted Data

Experimental Evaluation



- sFlow may not capture all attack flows due to sampling limitations.
- As a result, predictions using sFlow data could miss certain threats.

Top Five Most Important Features

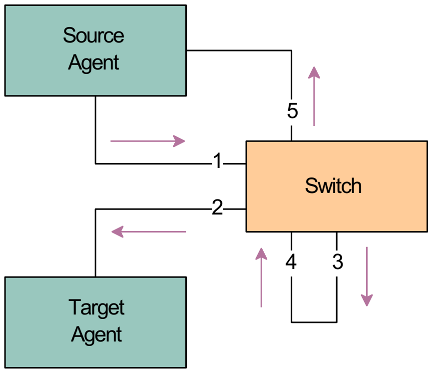
Experimental Evaluation

Features	RF	GNB	KNN	NN
Inter Arrival Time$_{cum}$	√	√	-	√
Inter Arrival Time$_{std}$	√	-	√	√
Packet Size	-	√	√	-
Packet Size$_{avg}$	√	√	√	√
Packet Size$_{std}$	√	-	-	-
Queue Occupancy$_{avg}$	√	-	√	√
Queue Occupancy$_{std}$	-	√	-	-
Protocol	-	√	√	√

- The most important features for detecting DDoS attacks are: *Inter-Arrival Time*, *Packet Size*, *Queue Occupancy*, and *Protocol*.
- The variants of these features, such as individual values, cumulative statistics, averages, or standard deviations, can differ in importance across the ML models.

The INT Testbed

Experimental Evaluation



- The source and target servers powered by dual AMD EPYC 7451 24-core processors and 128GB of RAM. Each server utilizes a Mellanox ConnectX-5 network card capable of 100Gbps throughput.
- The switch is an Edgecore Wedge DCS800
- *tcpreplay -i ⟨interface⟩ -p ⟨number of packets⟩ ⟨pcap file path⟩*

Experimental Results II

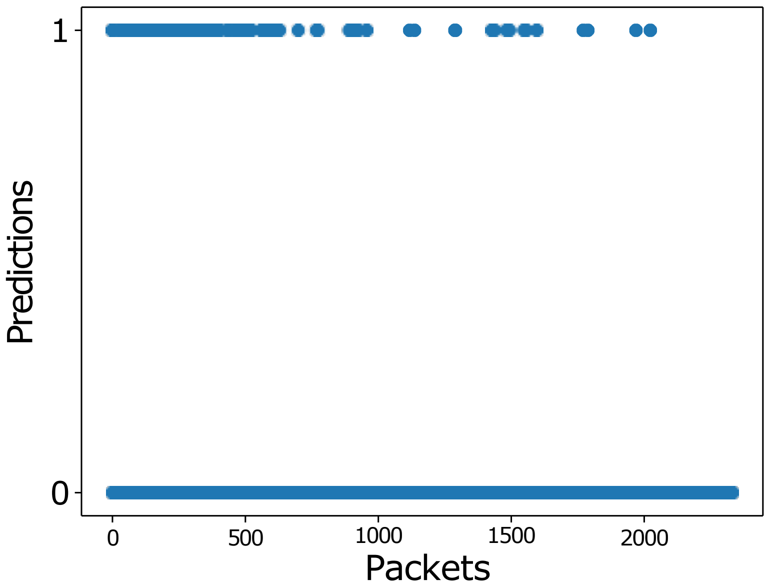
Experimental Evaluation

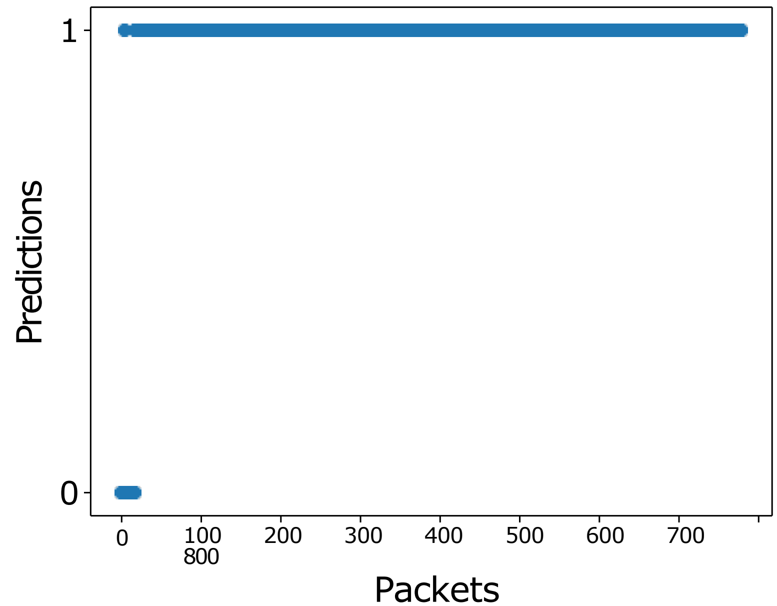
Attack Type	Accuracy	Misclassified/ Number of Predicted Packets	Average Prediction Time (s)	Max Prediction Time (s)
UDP Scan	0.9947	14/2628	0.12	0.73
SYN Scan	0.9961	10/2542	0.44	1.81
SYN Flood	0.9984	27/2814	0.09	0.4
SlowLoris	0.9795	16/779	0.05	130.85
Benign	0.9417	136/2331	103.14	734.55*

- We achieved over 97% accuracy in predicting most attack types, with an average response time of under 2 seconds.
- The creation of new flows appears to introduce bottlenecks and increase prediction time.

A Closer Look at Predictions

Experimental Evaluation





- Misclassifications occur in the initial instances of a new flow.

FIU

Table of Contents

Conclusion

Discussion and Conclusion

Conclusion

- INT data proved effective in detecting DDoS attacks for both known and novel attack patterns.
- sFlow performs similarly but may miss data due to its sampling approach.
- Automated detection, addressing bottlenecks, can be achieved in under 2 seconds.
- Efficiently storing, processing, and analyzing INT data requires substantial computational resources and optimized techniques.
- Establishing precise timestamps remains challenging.
- With our network capacity of 100 Gbps, the simulated attack did not cause significant congestion, limiting our ability to observe the effects on *queue occupancy*.

Questions

Conclusion

Thank you for your attention.
Questions are welcome.