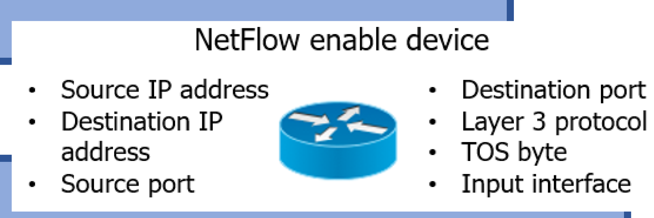
Botnet Detection by Integrating Multiple Machine Learning Models

JAPAN ADVANCED INSTITUTE OF SCIENCE AND TECHNOLOGY 1990 × Mahidol University

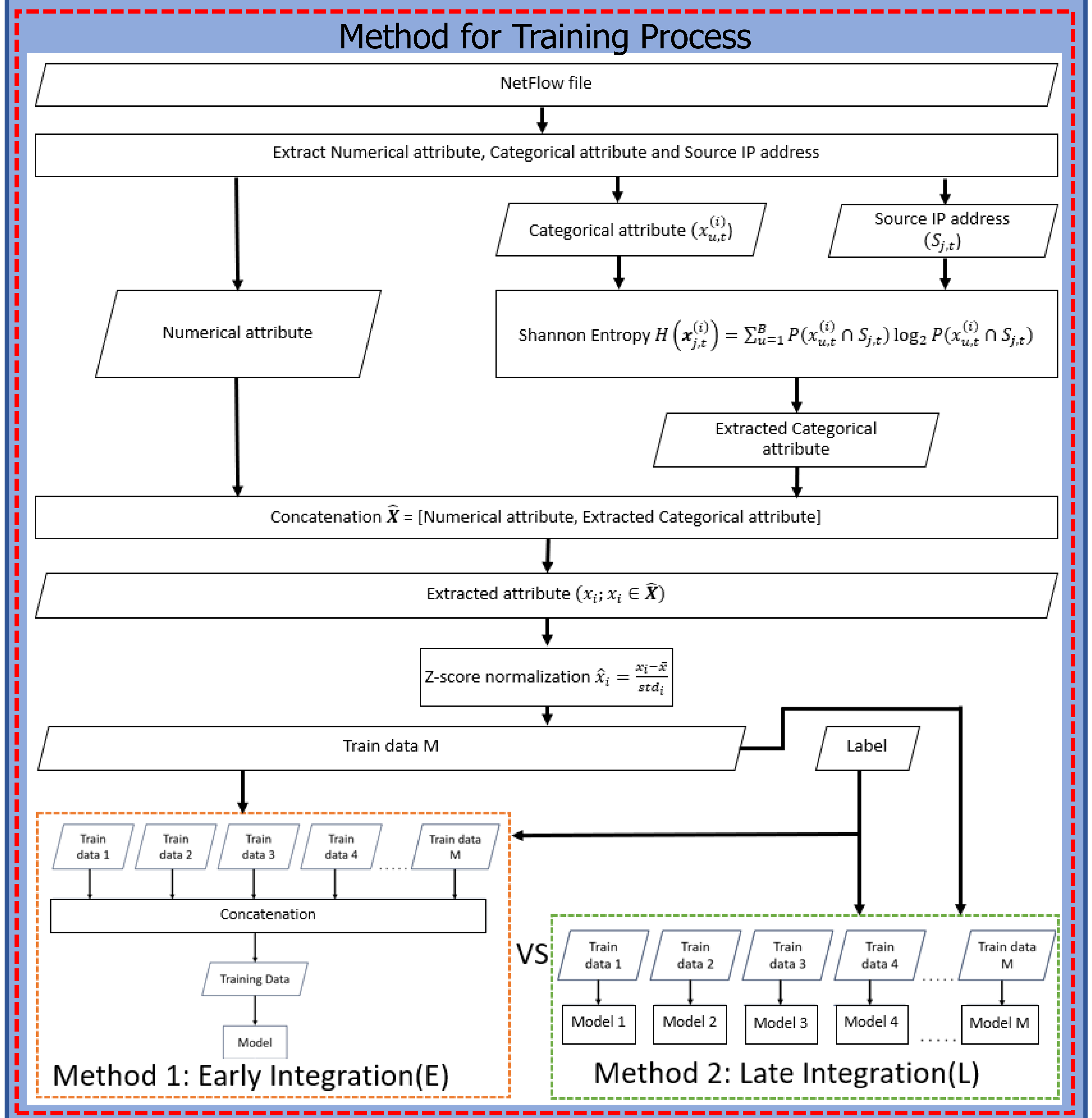
Thanawat Tejapijaya, Prarinya Siritanawan, Karin Sumongkayothin, Kazunori Kotani

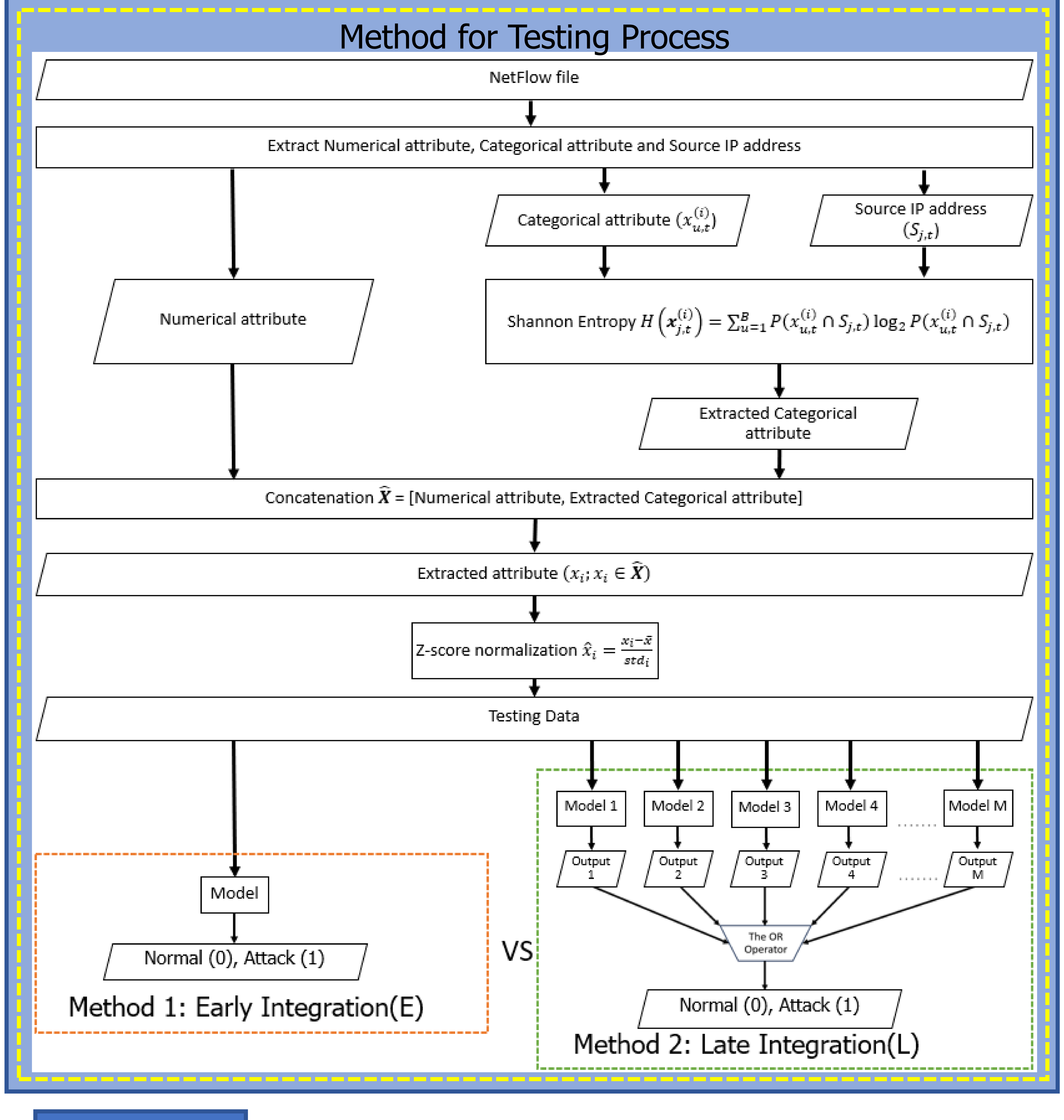
Introduction

A botnet is a collection of computers that have fallen victim to malware. It enables a single, malevolent individual, often referred to as a "botmaster," to manipulate the computers from a distance. Botnet represents one of the most aggressive cyber-attack threats. They are characterized by their elusive nature and evolving behaviors.



Method





Conclusion

- Our proposed method (Late Integration) showed remarkably low false negatives (less than 10), a high recall score (higher than 0.98).
- It is a plug-and-play method that can be maintained or updated at any time.

Problem Statement

Challenges persist in achieving low false negative rates for detecting various botnet behaviors.

Experiment Setup

We used the CTU13 dataset, which contains 13 different scenarios in NetFlow format. Each scenario corresponds to certain types of botnet attack which are IRC, spam, click fraud, port scan, DDoS, Command and control, P2P, and HTTP. The CTU 13 dataset consists of approximately 20 million data samples in total. We had conducted 2 experiments for each integrating technique as follows:

1. **Known scenarios approach:** All 13 scenarios are used as input for the training process. To train each model, we divide the data for each scenario into a training set (80%) and a test set (20%).
2. **Unknown scenarios approach:** 12 out of the 13 scenarios are used as input for the training process. One scenario is unknown and used as a test set.

Result

F1-Score for Integration method

F1-score	Integration method			
Testing	Late integration		Early integration	
Scenario ID	KnownL	UnknownL	KnownE	UnknownE
1	0.684	0.306	0.999	0.202
2	0.667	0.550	0.999	0.109
3	0.684	0	0.999	0
4	0.735	0.359	0.969	0
5	0.668	0.039	0.994	0
6	0.707	0.789	0.999	0
7	0.664	0	0.933	0
8	0.667	0	0.975	0
9	0.700	0.589	0.999	0
10	0.659	0.101	0.999	0
11	0.646	0.112	1	0
12	0.693	0	0.98	0
13	0.687	0.007	0.999	0

Precision Score for Integration method

Precision	Integration method			
Testing	Late integration		Early integration	
Scenario ID	KnownL	UnknownL	KnownE	UnknownE
1	0.619	0.967	1	0.999
2	0.614	0.989	1	0.996
3	0.619	0	0.999	0
4	0.626	0.277	1	0
5	0.612	0.022	1	0
6	0.646	0.734	1	0
7	0.605	0	1	1
8	0.640	0	0	0.999
9	0.645	0.993	0.999	0
10	0.596	0.561	0.999	0
11	0.588	0.953	1	0
12	0.634	0	0.989	0
13	0.625	0.003	1	0.894

Recall Score for Classification by individual models

Recall	scenario ID for training model												
Testing Scenario ID	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0.999	0.164	0	0	0	0	0	0	0	0	0	0	0.005
2	0.498	0.999	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0.997	0	0.017	0.001	0	0	0	0	0	0	0
4	0	0	0	0.952	0.094	0.153	0	0	0	0.306	0	0	0.001
5	0	0	0	0	0.997	0	0	0	0.019	0	0	0	0.207
6	0	0	0	0.853	0	0.999	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0.923	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0.958	0	0	0	0	0
9	0	0	0	0	0.003	0	0	0	0.999	0	0	0	0.387
10	0	0	0	0	0.039	0	0	0	0	0.999	0	0	0
11	0	0	0	0.006	0	0	0	0	0	0.058	0.999	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0.977	0
13	0	0	0	0	0.045	0	0	0	0.020	0	0	0	0.999

Recall score for Classification by individual models shows that using one scenario as the training input for an individual model proved to be ineffective, resulting in a recall score of 0 for most scenarios. However, certain scenarios contain similar botnet behaviors like scenario 1 and 2, where 1 can detect almost half of 2.

Recall Score for Integration method

Recall	Integration method			
Testing	Late integration		Early integration	
Scenario ID	KnownL	UnknownL	KnownE	UnknownE
1	0.990	0.182	**0.998**	0.112
2	0.990	0.381	**0.999**	0.057
3	0.974	0	**0.998**	0
4	**0.989**	0.509	0.941	0
5	0.984	0.197	**0.989**	0
6	0.990	0.852	**0.998**	0
7	**0.981**	0	0.875	0
8	**0.978**	0	0.953	0
9	0.985	0.419	**0.999**	0
10	0.989	0.556	**0.999**	0
11	0.984	0.059	**1**	0
12	**0.980**	0	0.970	0
13	0.981	0.061	**0.999**	0

Recall score for early integration method is higher than late integration method in the known approach. Unfortunately, in the unknown approach, most scores are 0. Therefore, in real scenarios where new attacks can occur at any time, the late integration method is preferable since it can detect more unknown scenarios, as shown as UnknownL in the Recall Score for Integration method table.

Notation of parameters

Notation of parameters in our feature extraction algorithm based on Shannon Entropy method

Notation	Description
t	Window index
B	Number of unique event $x_t^{(i)}$ in window t; $x_t^{(i)} \in \mathbf{x}_t^{(i)}$
u	Unique event index; $u \in \{0,1,2,\ldots,B\}$
i	Attribute index; $i \in$ {Proto, Sport, Dir, DstAddr, Dport, Sate, sTos, dTos}
$x_{u,t}^{(i)}$	Unique event x_u of categorical attribute i in window t
j	SrcAddr index; $j \in \{0,1,2,\ldots,J\}, J = \lvert\mathcal{S}_t\rvert$
$\mathcal{S}_t$	A set of all unique SrcAddr in window t
$S_{j,t}$	Unique SrcAddr j in window t; $S_{j,t} \in \mathcal{S}_t$
$x_{u,t}^{(i)} \cap S_{j,t}$	Unique event x_u of categorical attribute i that has same SrcAddr j in window t
$\mathbf{x}_t^{(i)}$	Vector of attribute i in window t
$H(\mathbf{x}_{j,t}^{(i)})$	Shannon Entropy of $\mathbf{x}_{j,t}^{(i)}$
$\bar{x}_i$	mean of x_i
std_i	standard deviation of x_i