A Survey on Data-driven Network intrusion Detection

Dylan Chou, Meng Jiang

Supervisor: Prof. Jong Hyuk Park

Presented By : Bhagyashree Kakde

2022.10.10

Seoul National University of Science and Technology, Seoul, South Korea

Table of content

Abstract

- This Survey focuses on frameworks and methodologies used in previous research and Eight main challenges of data-driven anomaly-based network intrusion detection.

- Those are lack of real-world data, noisy data, redundant data, weakly correlated data, too few correlated data, imbalanced data, dynamic data, big data, small data.

1. Introduction

1.1 Intrusive Detection System (IDS)

- System which analyze, monitor, capture, network traffic and misuse of resources by internal sources in it.
- There are two types of Intrusive Detection system: 1) Host based(HIDS), 2) Network Based(NIDS)
- These Systems monitors data packets in HIDS with respect to device and in NIDS with respect to Network.
- There are two Methods /Behavior: based on pattern followed in data pocket . efficiency of system depends on traffic volume, application on network, and type of data exchange.
    1. Pattern based: compares from the database of attack pattern
    2. Anomaly based: compares and update for new attack pattern (Novel Intrusion Attack)

1. Introduction

1.2 Past Survey (studied Papers)

- As Intrusive detection was adopted by cloud computing and many authors were focused on methods and rest primarily on network intrusion datasets.

- Many variables in Big data like storage volume, velocity, variety, intrusion detection system, cost were part of problem and technical solution in Hadoop system.

  - Jeong et al. [79]: anomaly tele traffic detection.
  - Modiet al.[119]: high level intrusion.
  - Keegan et al.[88]: NID datasets, approaches, cloud environment, algorithm etc.

- Other Authors where focused on NID datasets
  - Ring et al. [147]: focused packet flow, host log files, flow-based data.

1. Introduction (Cont.)

1.2 Past Survey

Table 1: Represent past papers with their focus area and technologies

	Title	Focus	Topics
Davis and Clark. 2011. [39]	**Data preprocessing** for anomaly based network intrusion detection: A review	Data prepro-cessing	Relevant features construction using targeted content parsing and deeper network packet inspection
Jeong et al. 2012. [79]	Anomaly teletraffic intrusion detection systems on **Hadoop-based Platforms**: A survey of some problems and solutions	Framework	Hadoop and big data platforms for speed, storage volume, and cost-efficiency
Poston. 2012. [139]	A brief taxonomy of intrusion detection **strategies**	Strategies	Taxonomy of traditional network intrusion detection
Modi et al. 2013. [119]	A survey of intrusion detection techniques in **Cloud**	Framework	Incorporating IDS on host system and virtual machines
Keegan et al. 2016. [88]	A survey of **cloud-based** network intrusion detection analysis	Framework	Integrating machine learning algorithms and MapReduce to cloud computing environments
Resende and Drummond. 2018. [143]	A survey of **random forest**-based methods for intrusion detection systems	Machine learning strategy	Application of random forest methods over time
Ring et al. 2019. [147]	A survey of network-based intrusion detection **data sets**	Data collection	Categorization of 34 public datasets

Table 1: Represent past papers with their focus area and technologies

2. Data Processing

Various techniques used for data reduction

- LDA for function reduction before classification to address heavy computation done on payload-based anomaly intrusion detection.
- [65] paper presented a framework with correlation based and chi-square were applied to get important feature set. and for classification of Support vector machines(SVMs),random forest, Gradient boosted decision trees and Naive bayes were used.
- [67] paper used a combination of a multilayer perception(MLP) network and artificial(ABC) algorithm and fuzzy clustering to detect intrusion.

2. Data Processing(Cont.)

Table 2. Vertical Comparisons of Common Datasets

Dataset	Duration	Traffic Type	Method	#IPs	#Instances
KDDCup1999 [40]	N/A	Synthetic	Tcpdump	N/A	4,898,430
NSL-KDD 2009 [53]	7 weeks	Synthetic	N/A	N/A	125,973/22,544
UNSW NB15 IDS [131]	15–16 hours	Synthetic	Tcpdump/IXIA PerfectStorm	45	2,540,044
UGR'16 [51]	96 days	Real	Netflow	600M	16.9M
CIDDS'17 [130]	4 weeks	Emulated	Netflow	26	32M
CICDS'17 [54]	5 days	B profile sys.	User behavior	21	2,830,743
CSE-CIC-IDS2018 [55]	17 days	B/M profile system	CICFlowMeter	500	4,525,399
LITNET-2020 [132]	10 months	Real	Flow traces	7,394,481	39,603,674
MAWILab [95]	15 min/d	Real	Sample point collection	N/A	N/A

3. Common Public Datasets and Reproducibility

- Dataset Compared in Table 2 is newly updated, highly cited data driven NID papers .

- Table 3 includes papers with datasets and its reproducibility method ,paper here are most cited with past decade period . These papers are relevant to network detection.

3. Common Public Datasets and Reproducibility(Cont.)

Table 3. Datasets and Papers That Used the Datasets for Evaluation

Dataset	Research works that used the dataset for evaluation
KDDCup1999	• [26, 27, 31, 34, 50, 57, 65, 72, 91, 92, 103, 112, 114, 148, 152, 158, 173, 176, 179, 181, 183, 187–189] ★ [7, 33, 44, 49, 64, 70, 89, 90, 98, 100, 101, 117, 120, 122, 141, 159, 164, 172, 182, 185]
NSL-KDD 2009	• [36, 45, 59, 73, 80, 81, 102, 105, 133, 134, 137, 150, 158, 170, 176, 181, 202] ★ [60, 75, 93, 125, 142, 149, 163, 174, 184, 190, 197, 198] ▷ [69]
UNSW NB15 IDS	• [16, 20, 73, 161] ★ [81, 89, 90, 122, 154, 174, 182, 184, 191] ▷ [69]
UGR'16	• [109] ▷ [110]
CIDDS-001	• [2, 144, 148]
CICIDS'17	• [6, 11, 29, 45, 157] ★ [63, 140, 191] ▷ [46, 69, 180, 194, 195, 201]
CSE-CIC-IDS2018	• [91]
LITNET-2020	• [38]
MAWILab	▷ [201]

"•" for those that are general methods or frameworks.
"★" for those that pseudo code and implementation details are available.
"▷" for those that have documented code associated with the paper.

3. Common Public Datasets and Reproducibility(Cont.)

3.1 Dataset Description

3.1.1 KDD cup 1999 : With this datasets Attacks fall into 4 main categories sync flood , unauthorized access to a remote machine (R2L), unauthorized access to a local superuser(U2R), probing . It runs into problem of duplication between training and testing , missing IP addresses (source and destination) even after provision of TCP attributes. it has bias due to synthetic generation.

3.1.2 NSL-KDD 2009 : This dataset was developed to removed some duplication of data problem from KDD cup 1999 .

3.1.3 UNSW NB 15 IDS : This Database explains detection through structure .This dataset was created from traffic generator.

3.1.4 UGR'16 : Data set is split into calibration and training set. Long term evolution and periodicity in data is advantage but network traffic is labeled as "Background".

3. Common Public Datasets and Reproducibility(Cont.)

- 3.1.5 CIDDS-001 : This dataset is primarily used for benchmarking.
- 3.1.6 CICIDS'17 : The data collection occurred over the course of five days where Monday was benign activity, Tuesday was brute force, Wednesday was DoS, and Thursday was web attacks where the afternoon saw Botnet, Port Scan, and a DDoS LOIT.
- 3.1.7 CSE-CIC-IDS2018 : since environment is supported in AWS, the network topology includes an attack network of 50 machine . 5 department holds 100 machines each and server with 30 machine.
- 3.1.8 LITNET-2020 : it contains Real-network attacks in Lithuanian- wide Network .
- 3.1.9 MAWI Lab : It records data 15 minutes of network traces everyday.

3. Common Public Datasets and Reproducibility(Cont.)

3.2 Reproducibility on datasets

- [69] Code Repository contains separate files to load data and runs a model for result. Train and test dataset paths are defined according to author's environment along with naming schemes.
- [110] code repository implemented logistic regression, random forest and SVM in python. Implementation of the partial least squares regressions are not provided. hence code is partially reproducible for some model performances.
- [46] code repository were made public. For some package python 2 version was used ,restricts python 3.7 . Hence follow up on package versions and fixes with author is needed.
- [201] Contains a demo that consolidates the code for parsing the network flow packets, extracting features, training the deep belief network autoencoder and LSTM and plotting anomaly score for indication of network packets. Code is able to completely recreate the result.

3. Common Public Datasets and Reproducibility(Cont.)

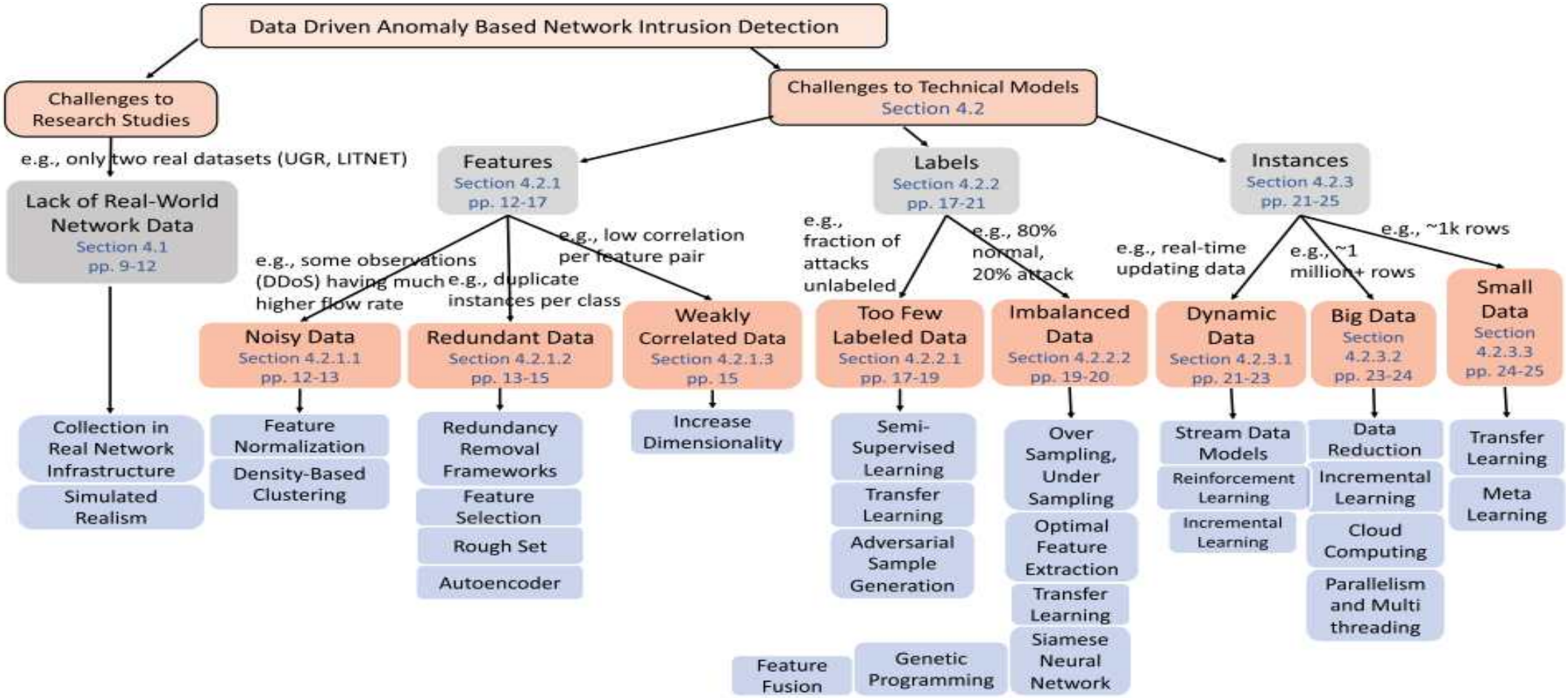


Fig. 1. Hierarchical chart of categorized high-level challenges and recent methods to resolve them. The section and page numbers are included in the boxes so readers can easily skip around different sections.

4. A taxonomy : Challenges and Methods

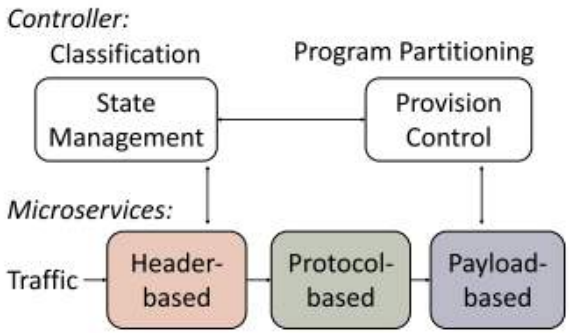
4.1 Lack of Real-world Network Data

- To overcome the challenge researchers simulated realistic network with synthetic data generation or with simulated virtual network and for that Honeypots were used but honeypot can sense for its own not otherwise.

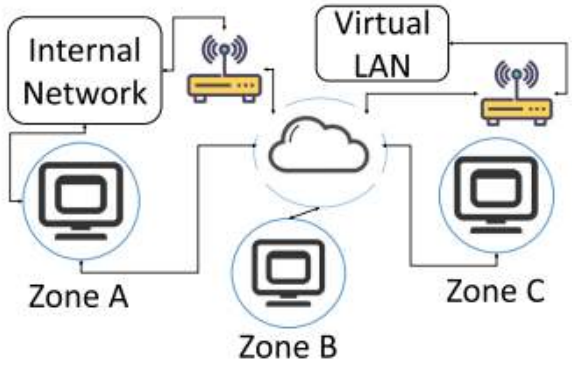
- UNSW-NB 15 was created where TCP dump in IXIA traffic generates synthetical data for simulating realistic intrusion network .it was updated by generating same traffic via IXIA perfect storm and collection of host logs. Realism of dataset was also verified using sugeno fuzzy interface engine.

- Second approach was collecting it into network infrastructure . A mixture of virtualization and cloud intrusion detection with hypervisors were used to solve the issue of small datasets by increasing power of network traffic data.

Fig.2 Paradigms of systems used towards real-world network data collection (part 1)

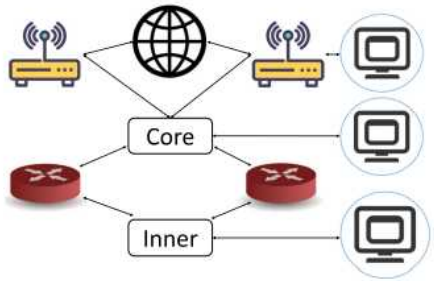


(a) vNIDS [99]: Network traffic arrives into the detection system and is passed through three types of microservices as shown in the figure. Then data is passed into the vNIDS controller, which contains state management that is responsible for detection state classification and provision control responsible for partitioning detection logic programs into header-and payload-based DLPs).

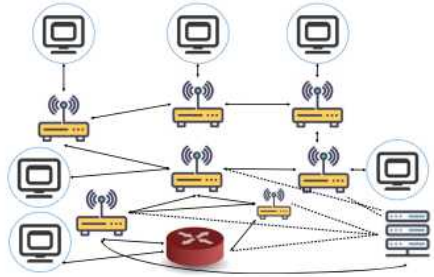


(b) ISOT-CID [9]: The three computer icons represent three hypervisor nodes (A, B, C) that hold 10 virtual machine instances. The yellow icons represent routers, and the cloud is the ISOT cloud network. Internal depicts the internal network that zone A's hypervisor is connected to and VLAN is connected, to zone B's hypervisor.

Fig.2 Paradigms of systems used towards real-world network data collection (part 1)



(c) UGR'16 [109]: The topology of the network begins with the internet represented through the globe icon, which have two routers, two yellow icons, connected to it. The attacker and victims' networks are depicted via computer icons. The two routers are called BR1 and BR2, which stand for border routers. The second border router is connected to the attacker network (five machines). The core network has 5 victim machines used in data collection, which has two firewalls represented by the red icons. The inner network holds 15 victim machines where five machines are placed in each of three distinct existing networks.



(d) LITNET [38]: The yellow icons are routers. The top three connecting nodes are CITY2 (Klaipeda University), CITY3 (Siauliai University), and CITY4 (KTU Panevezys Faculty of Technologies and Business), from left to right. The middle three are CITY1 (Kaunas–Vytautas Magnus University and Kaunas Technological University), KTU University 2, and CAPACITY (Vilnius Gediminas Technical University), from left to right. The lower left router is KTU University 1. The red icon is a firewall and the lower-right icon depicts a netflow server. The four nodes KTU UNIVERSITY 1, CAPACITY, KTU UNIVERSITY 2, CITY1 along with the firewall are netflow exporters that catch new traffic.

4. A taxonomy : Challenges and Methods(cont.)

4.1.1 Ability to Transfer

- Its hard to train data anomaly detection system in a real-network setting because it may introduce undetected intrusion on the security of network.

- Solution here is by transfer learning of intrusive detection system in a simulated environment to in-use network.

- *IDS methods* : Container based traffic collection is a model which is more efficient and scalable . This architecture divides network traffic to different detection logic program instances based on network header information. this reduces resource consumption in the virtualization of NIDS.

- [35] used containerized network for lack of ground truth traffic. Container isolates specific running applications. This gives benign traffic data .results into DOS .which can be relayed in virtual machine. It is not reliable in real-word-network.

Thank you for your attention

Bhagyashree Kakde
bhagyashreekakde27@gmail.com

SEOULTECH
SEOUL NATIONAL UNIVERSITY OF SCIENCE & TECHNOLOGY