TEST Engineering & Management

May-June 2020
ISSN: 0193-4120 Page No. 8280 - 8286

IoT Traffic Classification Techniques for Attack Detection using Machine Learning Algorithms

Wesam Raad

Dijla University College, Baghdad, Iraq

Article Info
Volume 83
Page Number: 8280 - 8286
Publication Issue:
March - April 2020

Article History
Article Received*: 19 November 2019*
Revised*: 27 January 2020*
Accepted*: 24 February 2020*
Publication*: 18 May 2020*

Abstract

The raise popularity of specialized Internet devices, called Internet of things (IoT), commitments conveniences and privacy concern. The largest use of IoT these days is security. The IoT attacked have been raised recently, the attacked increased by 600% since 2016. There are many ways to detect the IoT attacks. Network traffic classification is the most techniques are used in last years. The network traffic classification has many techniques. The popular technique used in last few years is Machine Learning techniques, which have been used via many Authors and get high accuracy. In this study, we explain IoT networks traffics classifications technique and IoT dataset, after that features extraction tool will be used to extract the features from dataset traffics, after that will use four machine learning algorithms which is SVM, Naive Bays, C4.5 and K-nearest. The experiment analysis show that C4.5 classifier got a good accuracy results comparing to other classifiers.

Keywords; *Business, Social Media, Cyber Bullying, Society, Education, Mobile Devices*

1 Introduction

The first mention of Internet of Things was in 1999 by Kevin Ashton [16]. Internet of Things, usually referred as IoT is ubiquitous concept where physical objects are connected to Internet and have the abilities to communicate over the network [9].

In 2016 there were 6.4 billions devices are connected through internet, according to Gartner researchers. While were 8.4 billions connected devices were used in worldwide in 2017, increased in 2018 to be more than 10 billion devices, and it will be 20.4 billions devices by 2020. The largest used of these devices in Western Europe, China, and North America, with the regions accounting for 67% of the overall IoT install base in 2017.

The largest use case for the IoT today is security [15]. The introduction of the IoT technologies raises many security concerns. Because of tight connections between a real world and IoT, its adoption can leads to safety and security breach [15].

The IoT attacked have been raised recently, according Symantec Global Intelligence Network in 2018, the attacked increased by 600% since 2016, and the vulnerabilities have been increased by 29% in the same period time.

The volume of the network traffics is constantly rising because of a new multimedia applications and advancement in network technologies [1]. The recent innovation links to the methods, systems and computers program product for the performing the classifications of network traffics. Network operator that handle networks traffics between, for example, mobile phones and web servers, classifies the network traffics in order to get the information about the use of them networks [13].

In this matter, applications classifications become more important for the managing QoS in the Networks and the security monitoring for different ISP and another governmental and the private organizations. The efficient and accurate applications classifications are the key stone of the networks monitoring, and on the basis of classifications result networks administrators can design different policies to increase the network security. But, the challenges are to classify the applications depends on IoT traffics features because the huge data in the high speed network [1].

In this study, we will talk about the Network traffics classifications techniques. Then we will explain about four machine learning algorithms. We first use the IoT dataset collected by [17]. After that we will use the NetMate tool to extract the features of the traffics from the dataset and we will implement four machine learning classifiers.

2 Related Work

Detecting attack on modern computers by analysis the network traffic is one of the interesting topics nowadays.

May-June 2020
ISSN: 0193-4120 Page No. 8280 - 8286

Many researchers try to analyzed the normal network traffic and IoT network traffic using many methods.
The [8] describes an approach for using machine learning algorithms to accurately recognize the IoT devices. The researchers first collected network traffic data from 13 different devices. The paper describes a multistage processes in which the set of the machine learning algorithms based-on classifiers are stratified to the streams of the sessions which originates from the particular device. The goal of this process was to define whether the traffics belonged to the smartphone, PC, or the specific IoT device. The paper describes some of the features used to train a model and describes the algorithms used to classify traffic, but there are not any graphs describing the traffic observed. This would have made it easier to understand how the traffic generated by each device differed. Additionally, the paper only describes the logging of network traffic of the devices. There is no mention of power consumption logging.
Later in the year, [8] published a second paper, in this paper describes the process of detecting unauthorized IoT devices on a network using machine learning techniques. The data collection portion of this paper is much more detailed than the first. Network traffic data was collected and labeled from 17 different IoT devices covering 9 different categories. The focus of the paper is machine learning techniques, so many of the finer details of the traffic patterns are lost. Additionally, the paper does not cover any power consumption statistics for the devices.
In [3], researchers found that it possible to determine IoT devices based on network traffic patterns and DNS packets being sent. Furthermore, it is possible to reveal information about a user even if all of the traffic is encrypted. Almost all of the devices that had been used in the paper, send DNS traffic to servers that no other device does, making it easy to identify where the traffic is coming from.
However, the dataset used in [3] is much smaller than [8]. The number and types of IoT devices examined are also more limited and no power consumption is taken into consideration since the goal of the paper was to identify traffic from outside of the network.
The primary purpose of [2] is to analyze networks traces from the test bed of usual IoT devices, illustrate public methods for the fingerprinting them attitude, and assess where privacy and security risks manifest. The test bed contained 14 common IoT devices. The paper shows that it is possible to associate specific devices with network traffic even if protection methods like MAC address randomization are used. This can have done with something as simple as looking at DNS requests to see what websites the devices are asking for. Some of the devices also used HTTP packets to send or receive all data, so it was possible to extract API keys and hijack devices.
The analysis performed in [2] shows the breakdown of what normal traffic looked like for 14 different common IoT devices. However, the data collected only covered a period of three weeks. Additionally, the paper did not monitor multiple devices from different manufacturers in the same category. For instance, the only smart speaker being monitored was the Amazon Echo. Additionally, there is no mention of power consumption for the monitored devices.

3 Machine Learning

There are a lot of techniques that have been used for traffics classifications (figure 1) and will be described in brief in this part.

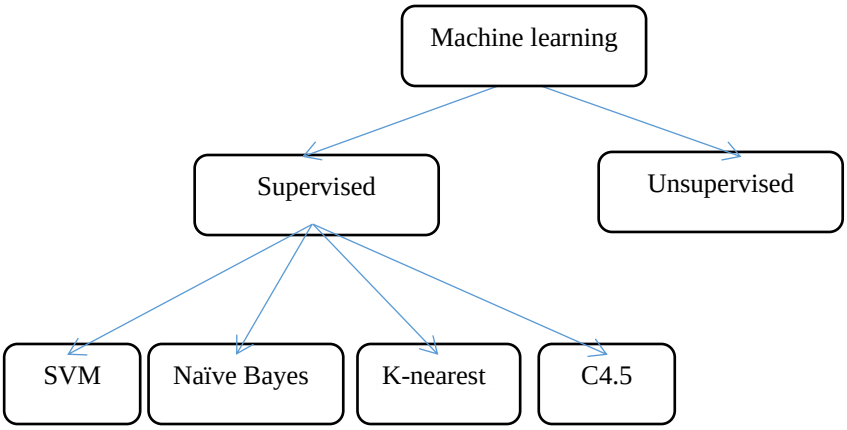

Figure 1: machine learning algorithms

Support Vector Machine (SVM)

SVM is strong algorithm that be used to resolve classifications and regression issues. In order to classify the algorithm, transform the input data to the high dimensional hyper plane, where it becomes more independent comparing to original forms [13]. This has been done via using nonlinear kernel function, and then linear classifier has been used to structure the most margins hyper plane to split the various types in the training data. The two hyper planes are established the both aspects of the hyper planes separating data that tend to the maximize spaces among two parallel hyper planes. The supposition of the distance among the parallel hyper plane is better than the generalization error of classifiers will be. The SVM's learn over the cases in forms of the data points which contributes to high accuracy classifications, other feature of this algorithm is it may handle the missing amounts and the noises effectively. But,

Published by: The Mattingley Publishing Co., Inc.

May-June 2020
ISSN: 0193-4120 Page No. 8280 - 8286

it requirements are complex and demands huge memories [10].

Naïve Bayes

This algorithm is the simplest probabilistic algorithm depends on implementing the Bayes' theory with the powerful independent supposition. The most illustrative expression for underlying expectation model could be the self-determine the features model. Simply, the Naive Bayes presume that the presence of the specific features of the category is not related to presence of another features [11]. This classifier perform rationally good even if the underlying supposition is false. The advantages of this algorithm is require a small value of the data to rate the mean and difference of value that necessary for the classifications. Because of the independents value are undetermined, only the differences of values for every label that needs to be defined and not all covariance matrix. In contrast to this algorithm operators, its Kernel process could be implementing on the numerical features.

K-Nearest

It is a non-parametric decision execution for the machine learning and the data mining jobs. K-Nearest is considered as the most effectives algorithm in the Machine Learning algorithms, and top algorithms in the data mining [7, 14]. This algorithm assigning to the un seen sample x, category of the nearest training data according to many distance metrics. This algorithm, with $k > 1$, is the popularization of NN method where predicted classes of the x is set as similar to classes represented the plurality of its KNN in training set.

But, k-NN suffers from many issues like the huge memory requirements, huge computational complexity in an operational stage, and the low tolerance to the noise because of considering the whole instances as pertinent during training set might include noise or mislabeled instance. Various algorithms have suggested alleviating those issues. One of the techniques, known as prototype selection, comprises of the selecting the suitable subset of data which yields the same or larger classifications accurate. The prototype selection techniques could be classified in to three various types. First, algorithms eliminate noises instance from the original training set in order to develop classifications accurate. Second, condensation methods select the sufficiently small subsets of the training examples that lead to similar effectiveness of single nearest neighbor rule, via deleting examples which won't effect on the classifications accuracy. Third, hybrid technique selects the small subsets of the training instances which combine the targets of the previous two algorithms [7, 6].

C4.5 Decision Tree

This algorithm used and generated tree depends on the structure that could be used for the classifications that's why it also known statistically method. It used concepts of the entropy method for the classifications, for example, we have data M= {m1, m2…….mx.} where m1, m2 ….mx represent the training sample of the dataset that are described via various characteristics, let say {K1, K2…} are corresponding characteristics consisting goal category. The C4.5 algorithm selects specific features of the dataset on all nodes, that used to divide those samples to various categories. The aim of selecting features relies on normalized obtain data from samples, features with huge normalized obtain are selected and decision had made [5]. There are a lot of features of using the algorithm which are:

- Self-Explanatory and easy to follow
- Can handle both numeric and nominal input attributes
- Can handle a data set with many errors including missing values

But, many decision trees request the goal variable to have separated amounts; they tend to the perform well with noncomplex characteristics. Moreover, they are so sensitive to training datasets any corrupt amounts close to roots nodes could changes the all structure of the tree [12]

4. Methodology

In this part, we clarify IoT network traffics classifications models, which include the processes as shown in Fig.2. This process methods shows how to use IoT networks traffics classifications techniques to classify unknown IoT network traffics classes by using machine learning techniques.

Published by: The Mattingley Publishing Co., Inc.

May-June 2020
ISSN: 0193-4120 Page No. 8280 - 8286

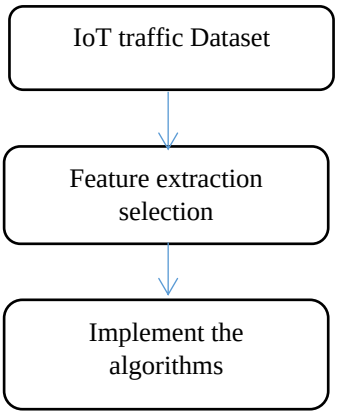

Figure 2: IoT network traffic model

IoT Traffic Dataset

The dataset that will be used in this paper is collected by Mirsky et al. (2018). The dataset consists of two deployments of four HD surveillance cameras each. The cameras in the deployments are powered via PoE, and are connected to the DVR via a site-to-site VPN tunnel. The DVR at the remote site provides users with global accessibility to the video streams via a client-to-site VPN connection. Figure 3 shows the topology of the network that used in this paper.

The dataset has many attacks, for example, a SYN flood on a target camera, or a man in the middle attack involving video injection into a live video stream. Table III summarizes the attack in the dataset.

Feature extraction selection

Features selections and extractions step follow. In this part, the features will extract from the dataset like packets duration, packets length; inter arrival packets time protocol and so on. The extracted features will be used to test the Machine Learning classifiers. For features extraction, Perl script could be used to extract the features from dataset. But in this work we will use Netmate tool for the features extraction and we will extract 23 features. We use MS Excel for saving the data set for the Weka tool as CSVformat.

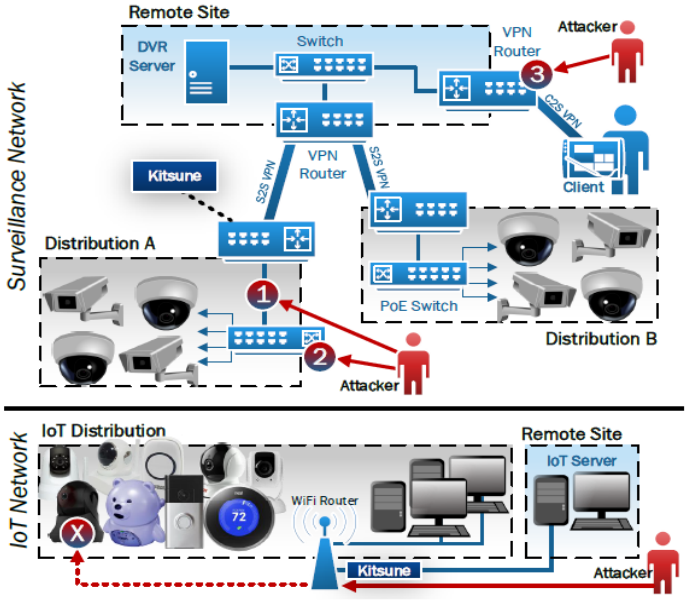

Fig. 3: The network topologies used in the experiments

Published by: The Mattingley Publishing Co., Inc.

May-June 2020
ISSN: 0193-4120 Page No. 8280 - 8286

Table III: the attacks in Mirsky et al. (2018) dataset

Attack Type	Attack name	Description
Recon.	OS Scan	The attacker scans the network for hosts, and their operating systems, to reveal possible vulnerabilities.
	Fuzzing	The attacker searches for vulnerabilities in the camera's web servers by sending random commands to their cgis.
Man in the Middle	Video Injection	The attacker injects a recorded video clip into a live video stream.
	ARP MitM	The attacker …intercepts all LAN traffic via an ARP poisoning attack.
	Active Wiretap	The attacker intercepts all LAN traffic via active wiretap (network bridge) covertly installed on an exposed cable
Denial of Service	SSDP Flood	The attacker overloads the DVR by causing cameras to spam the server with UPnP advertisements.
	SYN DoS	The attacker disables a camera's video stream by overloading its web server
	SSL Renegotiation	The attacker disables a camera's video stream by sending many SSL renegotiation packets to the camera
Botnet Malware	Mirai	The attacker infects IoT with the Mirai malware by exploiting default credentials, and then scans for new vulnerable victims network

Feature extraction selection

Features selections and extractions step follow. In this part, the features will extract from the dataset like packets duration, packets length; inter arrival packets time protocol and so on. The extracted features will be used to test the Machine Learning classifiers. For features extraction, Perl script could be used to extract the features from dataset. But in this work we will use Netmate tool for the features extraction and we will extract 23 features. We use MS Excel for saving the data set for the Weka tool as CSVformat.

Implementation

In this stage, implementing the machine learning algorithms or the classifiers. For the implementation of the machine learning algorithms, there are a lot of tools on internet used to implement the classifiers, but MatLab and Weka classification tools is the most tools are using these days. In this work, Weka tools has been used to implement four machine learning algorithms which are C4.5, SVM, K-nearest, and Naïve Bayes to build the classifications model.

Results

In this section, we present the results of several machine learning algorithms applied on dataset collected by Mirsky et al. (2018).
After the implementations of several machine learning algorithms on the dataset, we tested the accuracy and the execution times of each algorithm.
The results show that C4.5 algorithm gave high accuracy more than SVM, K-nearest and naive Bayes. The C4.5 detect 78.9% of the attack in the dataset, SVM detect 74.2% of the attack, k-nearest detect 70.4% of the attack and naïve Bayes detect 72.7% of the attack (table 2 & figure 4).

Table 2: Accuracy and training time results of classifications methods

Classifier	Accuracy (%)	Time (mins)
C 4.5	78.9	0.3
SVM	74.2	0.36
K-nearest	70.4	0.57
NaiveBayes	72.7	0.41

Figure 4 shown the comparison of accuracy results of using four machine learning algorithms.

Published by: The Mattingley Publishing Co., Inc.

May-June 2020
ISSN: 0193-4120 Page No. 8280 - 8286

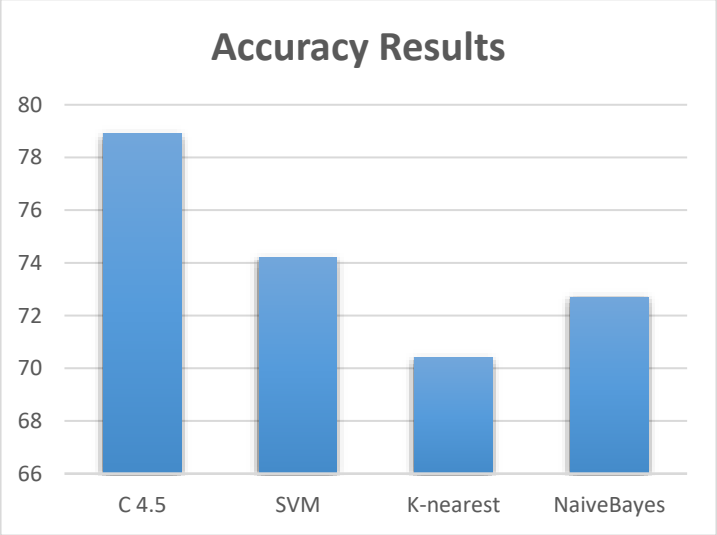

Figure 4: accuracy results of machine learning algorithms.

For the execution time, the C4.5 gave the minimum time less than other algorithms. The executed time that needed to execute C4.5 was 0.3 minutes, while the SVM needed 0.36 minutes to execute, K-nearest needed 0.57 minutes to execute and naïve Bayes needed 0.41 minutes to execute (table 2 & figure 5)

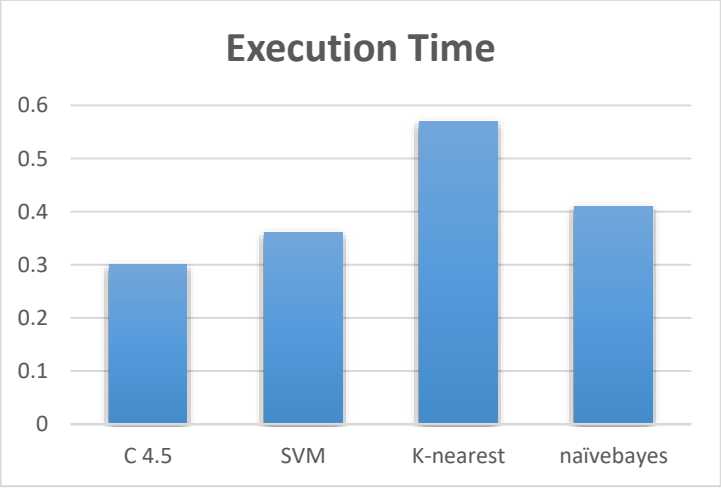

Figure 5: execution time results of machine learning algorithms.

From the results above, we can see the C4.5 algorithm gave better accuracy and less execution times than SVM, K-nearest and Naïve Bayes. Because C4.5 can handling both continuous and discrete attributes, C4.5 creates a threshold and then splits the list into those whose attribute value is above the threshold and those that are less than or equal to it. Also the C4.5 can handling training data with missing attribute value, Handling attributes with differing costs and Pruning trees after creation.

Conclusion

In this paper, we explained IoT Network traffics classifications techniques and explained how the researchers applied the network traffics classifications techniques by using Machine Learning algorithms to classify IoT attacks. And then we perform comparative analysis between four machine learning classifiers. First we used an IoT traffics dataset collected by (Mirskey et al, 2018) and then we used Netmate tool to extract and select 23 features. After that, traffics are classified using four machine learning algorithms. The experimental results show that the C4.5 algorithm gave highest accuracy results comparing with other Machine learning algorithms. The C4.5 got 78.9% within 0.3 minutes while the SVM 74.2% within 0.36 minutes.

Published by: The Mattingley Publishing Co., Inc.

May-June 2020
ISSN: 0193-4120 Page No. 8280 - 8286

References

1. Al-araji, Z.J. et al., 2019. Network Traffic Classification for Attack Detection Using Big Data Tools : A Review. Intelligent and Interactive Computing, Lecture Notes in Networks and Systems 67, pp.355–363.
2. Amar, Y. et al., 2018. An Analysis of Home IoT Network Traffic and Behaviour. ArXiv e-prints.
3. Apthorpe, N., Reisman, D. and Feamster, N., 2017. A Smart Home is No Castle : Privacy Vulnerabilities of Encrypted IoT Traffic. arXiv preprint arXiv:1705.06805.
4. Meidan, Y., Bohadana, M., Shabtai, A., et al., 2017. ProfilIoT : A Machine Learning Approach for IoT Device Identification Based on Network Traffic Analysis. ser. SAC ’17. New York, NY, USA: ACM, 2017,, pp.506–509.
5. A. Zhu, "A P2P Network Traffic Classification Method Based on C4. 5 Decision Tree Algorithm." pp. 373-379.
6. Brighton, H., Mellish, C.: Advances in Instance Selection for Instance-Based Learning Algorithms. Data Mining and Knowledge Discovery 6(2), 153–172 (2002). DOI 10.1023/A:1014043630878
7. Garcia, S., Derrac, J., Cano, J.R., Herrera, F.: Prototype selection for nearest neighbor classification: taxonomy and empirical study. IEEE transactions on 32 pattern analysis and machine intelligence 34(3), 417–35 (2012). DOI 10.1109/ TPAMI.2011.142
8. Meidan, Y., Bohadana, M., Tippenhauer, N.O. and Guarnizo, J.D., 2017. Detection of Unauthorized IoT Devices Using Machine Learning Techniques. arXiv preprint arXiv:1709.04647.
9. Ndibanje, B., Lee, H.J. and Lee, S.G., 2014. Security analysis and improvements of authentication and access control in the internet of things. Sensors (Switzerland).
10. P. Pinky, and S. V. Ewards, “A Survey on IP Traffic Classification Using Machine Learning.”
11. Patrick Breheny, Kernel density classification, STA 621: Nonparametric Statistics October 25.
12. R. Alshammari, and A. N. Zincir-Heywood, “Identification of VoIP encrypted traffic using a machine learning approach,” Journal of King Saud University-Computer and Information Sciences, vol. 27, no. 1, pp. 77-92, 2015.
13. V. D'Alessandro, B. Park, L. Romano, and C. Fetzer, "Scalable network traffic classification using distributed support vector machines." pp. 1008-1012.
14. Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G.J., Ng, A., Liu, B., Yu, P.S., Zhou, Z.H., Steinbach, M., Hand, D.J., Steinberg, D.: Top 10 algorithms in data mining. Knowledge and Information Systems 14(1), 1–37 (2007). DOI 10.1007/s10115-007-0114-2
15. Yang, Y. et al., 2017. A Survey on Security and Privacy Issues in Internet-of-Things. IEEE Internet of Things Journal, 4(5), pp.1250–1258.
16. Ashton, Kevin. "That ‘internet of things’ thing." RFID journal22.7 (2009): 97-114.
17. Yahalom, Ran; Steren, Alon; Nameri, Yonatan; Roytman, Max(2018), “Small versions of the extracted features datasets for 9 attacks on IP camera and IoT networks generated by Mirskey et al (2018).

Published by: The Mattingley Publishing Co., Inc.