Machine learning for IoT network monitoring

Mustafizur R Shahid, Gregory Blanc, Zonghua Zhang, Hervé Debar

Machine Learning for IoT Network Monitoring

Mustafizur R. Shahid, Gregory Blanc, Zonghua Zhang, Hervé Debar
CNRS SAMOVAR UMR 5157, Institut Mines-Télécom, France
{mustafizur.shahid, gregory.blanc, herve.debar}@telecom-sudparis.eu
zonghua.zhang@imt-lille-douai.fr

***Abstract*—The growing Internet of Things (IoT) market introduces new security challenges for network administrators. Most IoT devices are poorly configured making them a target of choice for attackers. Mirai botnet illustrates the threat posed by IoT devices. In this context, Machine Learning techniques can be leveraged to detect attacks in IoT networks. Indeed, contrary to desktop computers or laptops, IoT devices are used for very specific tasks. Therefore, the generated network traffic follows a predictable pattern making data analysis techniques well suited to detect a deviation from the expected behavior. In this paper, we present machine learning based techniques for IoT network monitoring. We first built an experimental smart home network to generate network traffic data. The network traffic is described using features, such as the size of the first N packets sent and received along with the corresponding inter-arrival times. We then train and test classification algorithms for devices recognition purposes. We also describe how to use autoencoders for anomaly detection in IoT networks.**

I. Introduction

The total number of IoT devices is expected to reach 75 billion by 2030 [1]. The growing IoT market introduces new challenges for network administrators [2]. Mirai botnet infected more than 600,000 devices around the world [3]. IoT botnet are primarily used for DDoS attacks. Legacy network monitoring methods are not tailored to cope with the ever growing IoT network.

Given the huge diversity of IoT devices (thermostat, camera, smart bulb, etc), device type recognition is critical. It will help to enforce security by applying device specific filtering rules. For example, knowing that a device is a security camera from a specific manufacturer can help the network administrator to configure the network so that the camera will not be allowed to do anything else than what it is expected to do. Device type recognition can also be used to block the access to the network of devices considered to be vulnerable.

Machine learning can also be leveraged to perform intrusion detection. Indeed, unsupervised machine learning algorithms can be leveraged to define the legitimate networking behavior of each IoT device in the network. To this purpose, autoencoders, an unsupervised neural network architecture, can be used. Any deviation from the expected behavior would trigger an alert indicating malicious activities.

In this paper, we present machine learning based approaches for IoT network monitoring. After presenting the related work in Section II, we describe the network traffic generation process in Section III. We also define the set of features used to describe network traffic data. Then in Section IV, a machine learning based device type recognition method is presented along with the experimental results. Next, in Section V an autoencoder based anomaly detection model is proposed.

II. Related Work

A few works exist focusing on IoT device type identification. Y. Meidan et al. [4] [5] present a machine learning based network traffic analysis approach to identify IoT devices. The purpose is to create whitelists of authorized devices. T. D. Nguyen et al. [6] propose a method to detect compromised IoT devices taking advantage of the temporal periodicity of traffic generated by IoT devices . First, legitimate communication profiles are created for individual devices. A recurrent neural network is then used to detect any deviation from the legitimate behavior. M. Miettinen et al. [7] present a method to identify the type of an IoT device being connected to the network by analyzing network traffic generated during the device setup. They use features extracted from link, network, transport and application layers of the packets sent by a device during its setup phase. B. Bezawada et al. [8] also describe a method to perform device behavioral fingerprinting. Our work on IoT device recognition through network traffic classification differs from existing ones in that we use a very different set of features that are easily extractable even from encrypted network traffic.

Some works focus on leveraging machine learning for intrusion detection in IoT. R. Doshi et al. [9] use supervised machine learning to perform DDoS attack detection in IoT network. Other works take advantage of unsupervised learning [10] [6]. In [10], Y. Meidan et al. describe network traffic using statistics aggregated by source IP, source IP-MAC, channel and socket. That is, the developed system assume that the IP and MAC addresses of each device are known beforehand. Which is not necessarily the case if the devices communicate through a NAT proxy and the detection system is located outside the local network. Moreover, the work primarily focuses on detecting infected devices and does not aim to separate malicious communications from legitimate ones. In [6] data are extracted from the incoming packet flows and are fed to a Gated Recurrent Unit (GRU) that estimates the occurrence probability of each new packet. A device is considered as being compromised if the probability of occurrence of a stream of packets fall below a determined threshold. They train a different model for each device. Hence, the device type needs to be known beforehand to apply the appropriate model limiting the capabilities of the approach. The presented method is also limited because it requires the

Fig. 1: Experimental smart home network

developed NIDS to be deployed in the same local network as the monitored device because network flows are defined based on the MAC address of the monitored device.

III. Network Traffic Data

Two types of network traffic is needed for our study: legitimate IoT network traffic and malicious network traffic (generated by malware targeting IoT devices). To generate legitimate IoT network traffic, a small smart home network is built as shown in Figure 1. The experimental smart home network consists of 4 smart devices: TP-Link Connected Bulb, Nest Security Camera, Mini, D-Link Motion Detector and Wemo Switch Smart Plug. The communication of the devices is collected thanks to a Raspberry Pi placed between the wireless access point and the Internet as shown in Figure 1. The network traffic is collected for 7 days. The collected data is split into a training set and a test set. To get malicious network traffic we have two solutions. Either we generate the malicious traffic in an offline network by purposely infecting a device, or we can get malicious traffic collected by an IoT honeypot such as IoTPOT [11]. The raw network traffic is preprocessed to extract useful features to feed the learning algorithms with. Our work focuses only on TCP connections. Hence, bidirectional flows identified by the tuple (source IP, destination IP, source port, destination port) are extracted from the raw network traffic. The bidirectional flows are described by the following features:

- The size of the first N packets sent
- The size of the first N packets received
- The N - 1 packet inter-arrival times between the first N packets sent
- The N - 1 packet inter-arrival times between the first N packets received

IV. IoT Device Recognition

In this section we test six different machine learning algorithms to classify the bidirectional flows according to the IoT device they belong to. Note that for this study only the legitimate network traffic is needed. The tested classification algorithms are Random Forest, Decision Tree, SVM (with rbf kernel), k-Nearest Neighbors, Artificial Neural Network (ANN) and Gaussian Naïve Bayes. The ANN is a fully connected feedforward neural network consisting of two hidden layers with 10 neurons each and using a dropout rate of 0.5 for regularization. The variable N, defined in Section III, is equal to 10. The metrics used to assess the performance are the accuracy, the micro-average of precision and recall, and the F1 score. The accuracy of the classifier is the proportion of flows that are correctly classified. Let us consider our 4-class classification problem. The four classes are $device_1$, $device_2$, $device_3$ and $device_4$. Let TP_i, TN_i, FP_i and FN_i be the number of true positive, true negative, false positive, and false negative respectively for $device_i$. The micro-average of precision, recall and F1 score are given by:

$$microAvPrecision = \frac{\sum TP_i}{\sum TP_i + \sum FP_i}$$

$$microAvRecall = \frac{\sum TP_i}{\sum TP_i + \sum FN_i}$$

$$microAvF1Score = 2\frac{microAvPrecision.microAvRecall}{microAvPrecision + microAvRecall}$$

TABLE I: Overall performance on the test set of the different classifiers

	accuracy	micro-av. precision	micro-av. recall	micro-av. F1 score
RF	.999	.999	.999	.999
DT	.995	.995	.995	.995
SVM	.993	.993	.993	.993
KNN	.989	.989	.989	.989
ANN	.986	.986	.986	.986
GNB	.919	.919	.919	.919

The results are shown in Table I.

With an overall accuracy of 91.9%, Gaussian Naive Bayes is the algorithm that performs the worst. All other algorithms achieve a high performance with an overall accuracy on the test set ranging between 98.6% and 99.9%. The best performance is achieved by the Random Forest classifier. Further details about the experiments are available in [12].

We also evaluate the performance achieved by the Random Forest classifier for different values of N (N being the number of packets sent and received that are taken into consideration, as defined in Section III). The obtained results are shown in Figure 2. The classifier achieves high accuracy even with a small value of N. Hence, for N equal to 2, the overall accuracy is as high as 98.9%. The maximum accuracy of 99.9% is achieved for N equal to 6 and higher.

V. Anomaly Detection for IoT Networks

Anomaly detection, also referred as unsupervised learning, is well-suited for IoT network because contrary to a laptop an IoT device performs a specific task. Indeed, the main difficulty encountered in general purpose network consisting of desktop computers, laptops or smartphone is the great variability and randomness of the generated network traffic. We propose to develop intrusion detection system in the case of a smart home environment. To this purpose we will explore unsupervised deep learning algorithms such as autoencoders. Anomaly detection models are trained on legitimate network

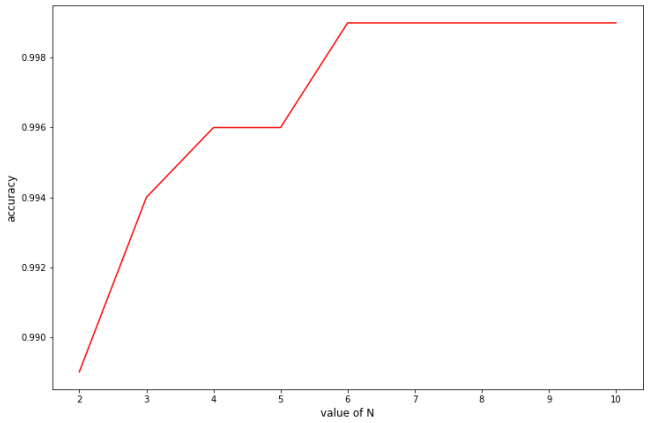

Fig. 2: Overall accuracy achieved by the Random Forest classifier for different values of N

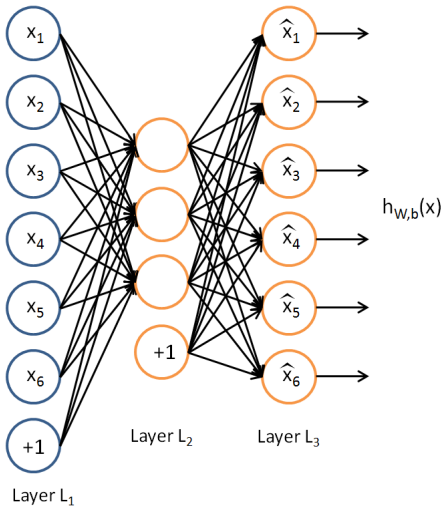

Fig. 3: Autoencoder

traffic data. The model learns the legitimate networking behavior profile of the device. Any networking activity that deviates from the expected behavior is considered as being malicious. An advantage of unsupervised learning is that it is able to detect new previously unseen attacks. An autoencoder is a particular neural network architecture that copy its input to its output under some constraints as shown in Figure 3 [13]. The constraint forces the neural network to learn an efficient representation of the input data. The constraint can be to limit the number of neurons in the hidden layer (vanilla autoencoder). The difference between the output and the input is called the reconstruction error. An autoencoder is very bad in reconstructing outliers. Hence, the reconstruction error can be used to detect anomaly. For each IoT device type, a different autoencoder is trained. Th eautoencoder will learn the expected legitimate behavior of the device. If the reconstruction error is too high, it indicates a possible attack.

VI. Conclusion

In this work we presented different machine learning based approaches for IoT network monitoring. First, an experimental smart home network was built to generate network traffic data. Bidirectional TCP flows are then extracted from the generated network traffic. Features used to describe bidirectional flows include the size of the first N packets sent and received, along with the corresponding inter-arrival times. The collected data are used to train different classification algorithms to recognize the IoT device type. An overall accuraccy of 99.9% is achieved by the Random Forest classifier. Further details about the work are available in [12]. Finally, we propose to use unsupervised deep learning algorithms, such as autoencoder, to detect attacks in IoT networks. In the future, the developed device type recognition and anomaly detection models will be integrated to a software defined networking (SDN) environment.

References

[1] Louis Columbus, "Roundup Of Internet Of Things Forecasts And Market Estimates, 2016," https://www.forbes.com/sites/louiscolumbus/2016/11/27/roundup-of-internet-of-things-forecasts-and-market-estimates-2016/.

[2] M. A. Khan and K. Salah, "Iot security: Review, blockchain solutions, and open challenges," *Future Generation Computer Systems*, 2018.

[3] M. Antonakakis, T. April, M. Bailey, M. Bernhard, E. Bursztein, J. Cochran, Z. Durumeric, J. A. Halderman, L. Invernizzi, M. Kallitsis, D. Kumar, C. Lever, Z. Ma, J. Mason, D. Menscher, C. Seaman, N. Sullivan, K. Thomas, and Y. Zhou, "Understanding the mirai botnet," in *Proceedings of the 26th USENIX Conference on Security Symposium*, ser. SEC'17, 2017.

[4] Y. Meidan, M. Bohadana, A. Shabtai, M. Ochoa, N. O. Tippenhauer, J. D. Guarnizo, and Y. Elovici, "Detection of unauthorized iot devices using machine learning techniques," *CoRR*, 2017.

[5] Y. Meidan, M. Bohadana, A. Shabtai, J. D. Guarnizo, M. Ochoa, N. O. Tippenhauer, and Y. Elovici, "Profiliot: A machine learning approach for iot device identification based on network traffic analysis," in *Proceedings of the Symposium on Applied Computing*, 2017.

[6] T. D. Nguyen, S. Marchal, M. Miettinen, M. H. Dang, N. Asokan, and A.-R. Sadeghi, "Dïot: A crowdsourced self-learning approach for detecting compromised iot devices," *CoRR*, 2018.

[7] M. Miettinen, S. Marchal, I. Hafeez, N. Asokan, A. Sadeghi, and S. Tarkoma, "Iot sentinel: Automated device-type identification for security enforcement in iot," in *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, 2017.

[8] B. Bezawada, M. Bachani, J. Peterson, H. Shirazi, I. Ray, and I. Ray, "Iotsense: Behavioral fingerprinting of iot devices," *CoRR*, 2018.

[9] R. Doshi, N. Apthorpe, and N. Feamster, "Machine learning ddos detection for consumer internet of things devices," in *2018 IEEE Security and Privacy Workshops (SPW)*, 2018.

[10] Y. Meidan, M. Bohadana, Y. Mathov, Y. Mirsky, D. Breitenbacher, A. Shabtai, and Y. Elovici, "N-baiot: Network-based detection of iot botnet attacks using deep autoencoders," *CoRR*, 2018.

[11] Y. M. P. Pa, S. Suzuki, K. Yoshioka, T. Matsumoto, T. Kasama, and C. Rossow, "Iotpot: Analysing the rise of iot compromises," in *9th USENIX Workshop on Offensive Technologies (WOOT 15)*. Washington, D.C.: USENIX Association, 2015. [Online]. Available: https://www.usenix.org/conference/woot15/workshop-program/presentation/pa

[12] M. R. Shahid, G. Blanc, Z. Zhang, and H. Debar, "Iot devices recognition through network traffic analysis," in *2018 IEEE International Conference on Big Data (Big Data)*, 2018.

[13] A. Ng *et al.*, "Sparse autoencoder," *CS294A Lecture notes*, 2011.