Proposed Models for Advanced Persistent Threat Detection: A Review

Santiago Quintero-Bonilla(✉) and Angel Martín del Rey

Institute of Fundamental Physics and Mathematics,
Department of Applied Mathematics, University of Salamanca, Salamanca, Spain
{santiago.quintero,delrey}@usal.es

Abstract. Advanced Persistent Threat is a sophisticated, targeted attack. This threat represents a risk to all organisations, specifically if they manage sensitive data or critical infrastructures. Recently, the analysis of these threats has caught the attention of the scientific community. Researchers have studied the behaviour of this threat to create models and tools that allow early detection of these attacks. The use of Artificial Intelligence can help to detect, alert and automatically predict these types of threats and reduce the time the attacker can stay on a network organisation. The objective of this work is a review of the proposed models to identify the tools and methods that they have used.

Keywords: Artificial intelligence · Machine learning · Advanced persistent threats · Malware · Detection · Cybersecurity

1 Introduction

Advanced Persistent Threats (APT) has been defined in several ways. In this work, we have been adopted the definition given by the US National Institute of Standards and Technology (NIST): "An adversary that possesses sophisticated levels of expertise and significant resources which allow it to create opportunities to achieve its objectives by using multiple attack vectors (e.g., cyber, physical, and deception)" [15].

Attackers study the behaviour of their victim to use them as the initial intrusion. They will look for ways to move laterally within the network, in search of users with high privileges, and access services with sensitive information [14].

The attack vectors can differ considering the particular and specific characteristics of the victim profiles. In this sense, Social Engineering plays an important role [11]. Attackers perform an intelligence gathering phase, and this generates attacks on mobile devices, exploit vulnerabilities, and even use the distribution of infected USB, e.g. the Stuxnet case [5], Duqu 2.0 [10] and Industroyer [3].

The characteristics of APT are the following: they have specific targets, sophisticated and highly organised attacks, endowed with a large number of resources. This type of attack is designed to remain within a computer network undetected for an extended period, usually in order to extract confidential

E. Herrera-Viedma et al. (Eds.): DCAI 2019, AISC 1004, pp. 141–148, 2020.
https://doi.org/10.1007/978-3-030-23946-6_16

information [17]. These features have made detection inefficient using traditional detection methods. Several companies have submitted reports [2,4,13], demonstrating the techniques used by APTs against government institutions, companies, industry sectors and individuals.

Each APT campaign is customised for each victim or organisation, and it has a unique life cycle. The report of the company FireEye about APT1 [12], "Anatomy of Persistent Advanced Threats", proposes a life cycle consisting of the following eigth phases: (1) Initial Recon, (2) Initial Compromise, (3) Establish Foothold (4) Escalate Privileges, (5) Internal Recon, (6) Move Laterally, (7) Maintain Presence (8) Complete Mission; In this life cycle, from the phase 4 to phase 7, the attackers maintain an extended persistence access in the attacked environment. In addition, the authors in [21] has analyzed 22 APT campaigns and they have identified three main phases of the life cycle of an APT: (1) initial compromise, (2) lateral movement and (3) command and control. This work allows visualising the relevant attributes with the techniques and methods used in an APT attack to propose prevention and detection approaches.

This article presents a review of the frameworks and models proposed for detection of APT attack.

The rest of paper is organised as follows: in Sect. 2 the Machine Learning methods are detailed, in Sect. 3 the related work is introduced and finally in Sect. 4 the conclusion is presented.

2 Machine Learning Methods

This section describes Machine Learning methods and the algorithms used for the different APT detection approaches that will help to understand the later sections.

Machine Learning (ML) is a tool that has begun to show usefulness in many scientific areas for problem-solving, due to the significant advantages of adaptability and scalability to the unknown [9]. Some examples of machine learning applications in cybersecurity are phishing detection, network intrusion detection, checking protocol security properties, cryptography and spam detection in social networks [6]. ML uses two types of techniques: supervised learning (training the model with both known input and output data to predict future results), and unsupervised learning (extracting the description of hidden structures from unlabelled data) [22]. The ML algorithms have been used for APT attack detection, and prediction approaches are the following:

- Desicion tree (DT) is a predictive model, it divides a complex desicion process into groups of less complicated decision [17]. Commonly used DT models are classification and regression trees (CART), ID3 and C4.5 [22].
- Support vector machine (SVM) analyses data and identifies patterns. It is a classifier based on finding a separating hyperplane in the feature space between two classes in such a way that the distance between the hyperplane and the closest data points of each class is maximised [1].

- k-nearest neighbours (k-NN) classifier is based on a distances function that measures the differences or similarity between two instances. The variation of k in k-NN represents the number of neighbour instances that need be compared, the results of the votes of the majority class element is considered the class [19,22].
- Ensemble learning combines multiple hypotheses, hoping to form a better prediction alone [1].
- Genetic programming (GP) is a subclass of genetic algorithms, which use replication and mutation to develop structures [16].
- Dynamic bayesian game model (DBG-model) is based on the diverse requirements of the application, and the game models are typically classified on two viewpoints: (1) dynamic and static game and (2) complete and incomplete information game [16].
- Correlation fractal dimension (FD) algorithm requires a reference dataset of the labelled features. Each new data point is classified as anomalous or benign by comparing the correlaction FD of the corresponding dataset [19].

3 Background

The detection of APT has been a challenge for current defence systems in cybersecurity. Different research approaches are being carried out to discuss this type of attacks. Recently, frameworks or models have been developed to help understand the behaviour of APT attacks. In what follows we present a brief analysis of existing models (see Table 1).

The authors in [7], present a novel machine learning-based system called MLAPT. This model consists of generating early warnings to detect an APT attack. The MLAPT is based on the analysis of a six-phase APT life cycle: (1) Intelligence Gathering, (2) Point of entry, (3) C&C Communication, (4) Lateral Movement, (5) Asset/Data Discovery, (6) Data Exfiltration. An individual detection model is inefficient at detecting APT attacks. For this reason, the authors seek to create a correlation framework between the different detection modules. These modules create alerts, and then they are analysed through ML algorithms. MLAPT works in three phases:

- Threat Detection: in this phase, network traffic is scanned through eight detection modules to detect techniques used by APTs. The detection modules are: Disguised exe File Detection (DeFD), Malicious File Hash Detection (MFHD), Malicious Domain Name Detection (MDND), Malicious IP Address Detection (MIPD), Malicious SSL Certificate Detection (MSSLD), Domain Flux Detection (DFD), Scan Detection (SD) and Tor Connection Detection (TorCD). The output of this phase are alerts known as events.
- Alert Correlation: in this phase, the alerts generated by the detection modules are correlated to find an APT attack. This phase is carried out in three steps: (1) alert filter, (2) alert clustering and (3) correlation indexing. At the end of the analysis can be generated two types of alerts: *apt_full_scenario_alert* and *apt_sub_scenario_alert*.

Table 1. Characteristics of frameworks proposed.

Paper	Framework	Design of the framework	APT life cycle
[7]	MLAPT	(1) Threat detection phase	6 phases
		(2) Alert correlation phase	
		(3) Attack prediction phase	
[16]	DFA-AD	(1) Initial phase: network traffic Intermediate phase: Classificator	Non-specified
		(2) Second phase: event correlation module	
		(3) Third phase: Voting services	
[19]	Fractal based anomaly	Anomaly classification algorithms:	Non-specified
		(1) k-NN	
		(2) Correlation fractal dimension based	
[20]	Detection based on dynamic analysis	Components:	3 phases
		(1) Network traffic redirection module	
		(2) User agent	
		(3) Reconstruction module	
		(4) Dynamic analysis module	
		(5) Decision module	
[18]	Detection based on big data	k-NN algorithm based on Mahout Phases:	Non-specified
		(1) Retrieve	
		(2) Reuse	
		(3) Revise	
		(4) Retain	
[8]	Context-based detection	Components:	6 phases
		(1) Pyramid attack levels	
		(2) Planes and events	
		(3) Correlation rules	
		(4) Detection rules	

– Attack prediction: this phase uses a machine learning-based prediction module. This process is completed in three steps (1) dataset preparation (2) training the prediction model and (3) using the model for prediction. In this module, the ML algorithms have been used are decision tree learning, support vector machine, k-nearest neighbours and ensemble learning.
As a result of this work, the authors indicate that the average of true positives is 81.8% and an average of 4.5% of false positives.

The authors in [16] propose a novel distributed framework architecture for APT detection called DFA-AD. This framework is based on multiple parallel classifiers, which classify events in a distributed environment and the correlation

of events between them. Each of these classifiers focuses on detecting the techniques used by the APT to carry out the attack.

Intrusion detection is realised in a distributed environment on the trusted platform module (TPM). A TPM can be integrated into the motherboard of a computer to implement secure communication within a network. DFA-AD is designed in three phases:

- Network traffic: in this phase, network traffic packets are collected, processed and analysed to identify all possible strategies that can be used in the life cycle of an APT attack. Also, the authors use four different classifiers as a method of recognition: genetic programming, classification and regression trees, support vector machines and dynamic bayesian game model. The output of these classification methods is transferred to the event correlation phase.
- Correlation event: in this phase, all the events generated by the recognition classifiers used in the previous phase are gathered to be evaluated using specific rules by an administrator who will warn about the detection of an APT attack.
- Voting service: after all the information generated in the previous modules, a voting service will analyse and determine if the system should generate an alert about an APT attack. The authors suggest that the module contributes to a decrease in the rate of false positives and improves the accuracy of the detection system.
  This framework has had a good detection rate of 98.5% with 0.024 false positives.

In the article [19], fractal-based anomaly classification mechanisms are presented to reduce false positives and false negatives.

The first step was to combine two datasets — one dataset with normal traffic packets and the another with APT attack network traffic packets. Then, an analysis of the TCP session data has been made to determine the characteristics of the vector. The classification of the compromise is specified in two metrics: (1) the total number of packets transferred during a single TCP session; (2) the duration of the complete session. Consequently, the dataset noise has been eliminated in two steps: (1) eliminate the packets that have zero length; (2) remove retransmitted packets. At the end of this pre-processing phase, the data is ready to be used in the algorithms.

Two anomaly classification algorithms have been used:

- k-nearest neighbours (k-NN) is a supervised learning algorithm, with a class of instant based learner.
- Correlation fractal dimension (FD) is based on exploiting the multifractal nature of the Internet data series.

The results of the fractal correlation dimension have improved by 12% accuracy and f-measure concerning the k-NN algorithm. The authors claimed the fractal dimension based algorithm is better than the Euclidean dimension based algorithm. The multifractal analysis extracts hidden information about a measurement that is not possible in a mono scale analysis.

The authors in [8], have designed a detection framework based on an attack pyramid model. This model works in the following way: correlation rules in contexts relate the events collected by each level of the pyramid. These contexts are exported to the alert system. This system applies the detection rules using a signature database. An evaluation of the detection rules is performed to update the levels of confidence and risk for each context. Threshold levels are checked based on confidence and risk results. If the threshold levels are in the alarm zone, an alarm is triggered and the APT incident response is initiated. Besides, the security analyst is notified to initiate the investigation of the event.

The authors in [20], propose a framework for dynamically reconstructing the network data stream to detect APT without affecting the typical workflow. This framework is built on five main components:

- Network traffic redirection module: this module copies the current network traffic and redirects it to the reconstruction module.
- User-agent: this module provides auxiliary information on the host for the reconstruction and decision module.
- Reconstruction module: restores data with malicious content and sends it to the dynamic analysis module.
- Dynamic analysis module: this module use the auxiliary information provided by the user-agent module and the data from the reconstruction module. It is possible to create a virtual environment similar to any host in the network.
- Decision module: this module integrates the above information and obtains a conclusion according to predefined criteria.

The authors in [18], propose an APT detection system, based on the Big Data architecture process. This system consists of four steps:

- Architecture of the APT system: in this phase, information related to the network data and the information in the system registers are collected and analysed.
- Processing technology: in this phase, to improve the analysis of an APT attack, is used the computational performance of the Hadoop[1] cluster.
- APT Analysis Technology: this phase includes the detection of malicious attacks through known vulnerabilities and suspicious connections with anomalous behaviour.
- APT detection algorithm based on Mahout: the authors indicate that the Mahout[2] library is suitable for APT detection because it can process large amounts of data, and the k-NN machine learning algorithm found within the Mahout project is used. This model has four phases: Retrieve, Reuse, Revise, Retain.

The results obtained during the test environment show that the capacity of the system to process the data can be up to 10 TB, together with the performance analysis of the data to classify an APT can be 2000 samples in parallel, to satisfy the requirements of APT detection.

[1] Apache Hadoop™ - https://hadoop.apache.org/.
[2] Apache Mahout™ - https://mahout.apache.org/.

4 Conclusion

In this article, we have reviewed the Machine Learning processes, techniques and algorithms that have been used in different approaches for the detection and prediction of APT attacks. We have been identified that the ML algorithms used in these approaches are supervised learning algorithms. This type of supervised learning allows data labelled from input data. Commonly algorithms used in these approaches to detect APT attacks are k-NN, SVM and Decision tree. The k-NN algorithm has been used in four of these approaches and has obtained the best results for APT attack prediction with a high rate of true positives.

In future work, we will propose a novel framework that implements algorithms to help process data generated and collected by security detection tools. Furthermore, the implementation of Machine Learning techniques in the event analysis would be advantageous, because a well-trained algorithm can learn to generate alerts to predict or not when an APT attack is executed. Also, the prediction step can be detected attack patterns, and anomalous connections with the data processed to make decisions. These decisions can be created rules to block traffic from a suspicious source and generate reports to the network administrators.

Acknowledgements. This research has been partially supported by Ministerio de Ciencia, Innovación y Universidades (MCIU, Spain), Agenda Estatal de Investigación (AEI, Spain), and Fondo Europeo de Desarrollo Regional (FEDER, UE) under project with reference TIN2017-84844-C2-2-R (MAGERAN) and the project with reference SA054G18 supported by Consejería de Educación (Junta de Castilla y León, Spain).

S. Quintero-Bonilla has been supported by IFARHU-SENACYT scholarship program (Panama).

References

1. Buczak, A.L., Guven, E.: A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun. Surv. Tutorials **18**(2), 1153–1176 (2016)
2. Check Point Research: Global Cyber Attack Trends Report. Technical report (2017)
3. Cherepanov, A.: WIN32/INDUSTROYER: A new threat for industrial control systems. Technical report (2017)
4. Cisco Systems, Inc.: Midyear Cybersecurity Report. Technical report (2017)
5. Falliere, N., Murchu, L.O., Chien, E.: W32. stuxnet dossier. White Pap. Symantec Corp., Secur. Response **5**(6), 29 (2011)
6. Ford, V., Siraj, A.: Applications of machine learning in cyber security. In: Proceedings of the 27th International Conference on Computer Applications in Industry and Engineering (2014)
7. Ghafir, I., Hammoudeh, M., Prenosil, V., Han, L., Hegarty, R., Rabie, K., Aparicio-Navarro, F.J.: Detection of advanced persistent threat using machine-learning correlation analysis. Futur. Gener. Comput. Syst. **89**, 349–359 (2018)

8. Giura, P., Wang, W.: A context-based detection framework for advanced persistent threats. In: Proceedings of the 2012 ASE International Conference on Cyber Security, CyberSecurity 2012, SocialInformatics, pp. 69–74 (2012)
9. Jang-Jaccard, J., Nepal, S.: A survey of emerging threats in cybersecurity. J. Comput. Syst. Sci. **80**(5), 973–993 (2014)
10. Kasperky Lab: The Duqu 2.0 - Technical Details (V2.1). Technical report June (2015)
11. Krombholz, K., Hobel, H., Huber, M., Weippl, E.: Advanced social engineering attacks. J. Inf. Secur. Appl. **22**, 113–122 (2015)
12. Mandiant: APT1 Exposing One of China's Cyber Espionage Units. Technical report (2013)
13. Mandiant: M-Trends 2017: A view from the front lines. Technical report (2017)
14. Navarro, J., Deruyver, A., Parrend, P.: A systematic survey on multi-step attack detection. Comput. Secur. **76**, 214–249 (2018)
15. NIST: Managing information security risk: Organization, mission, and information system view. Special Publication 800-839 (2011)
16. Sharma, P.K., Moon, S.Y., Moon, D., Park, J.H.: DFA-AD: a distributed framework architecture for the detection of advanced persistent threats. Cluster Comput. **20**(1), 597–609 (2017)
17. Shenwen, L., Yingbo, L., Xiongjie, D.: Study and research of apt detection technology based on big data processing architecture. In: 5th International Conference on Electronics Information and Emergency Communication (ICEIEC), pp. 313–316. IEEE (2015)
18. Shenwen, L., Yingbo, L., Xiongjie, D.: Study and research of apt detection technology based on big data processing architecture. In: 2015 5th International Conference on Electronics Information and Emergency Communication (ICEIEC), pp. 313–316. IEEE (2015)
19. Siddiqui, S., Khan, M.S., Ferens, K., Kinsner, W.: Detecting advanced persistent threats using fractal dimension based machine learning classification. In: Proceedings of the 2016 ACM on International Workshop on Security And Privacy Analytics - IWSPA 2016, pp. 64–69. ACM Press (2016)
20. Su, Y., Li, M., Tang, C., Shen, R.: A framework of apt detection based on dynamic analysis. In: 2015 4th National Conference on Electrical, Electronics and Computer Engineering. Atlantis Press (2015)
21. Ussath, M., Jaeger, D., Cheng, F., Meinel, C.: Advanced persistent threats: behind the scenes. In: Annual Conference on Information Science and Systems (CISS), pp. 181–186. IEEE (2016)
22. Xin, Y., Kong, L., Liu, Z., Chen, Y., Li, Y., Zhu, H., Gao, M., Hou, H., Wang, C.: Machine learning and deep learning methods for cybersecurity. IEEE Access **6**, 35365–35381 (2018)