Scientific and Practical Cyber Security Journal (SPCSJ) 3(3): 1 - 11 ISSN 2587-4667 Scientific Cyber Security Association (SCSA)

Proposed Framework for Effective Detection and Prediction of Advanced Persistent Threats Based on the Cyber Kill Chain

Faisal A. Garba
Department of Computer Science Education, Sa'adatu Rimi College of Education, Kano, Nigeria.

[1] Sahalu B. Junaidu, [2]Barroon I. Ahmad, [3]Abdoulie M. S. Tekanyi
[1,2]Department of Computer Science, Ahmadu Bello University, Zaria, Nigeria
[3]Department of Electrical & Computer Engineering, Ahmadu Bello University, Zaria, Nigeria

ABSTRACT

The cost of data breach resulting from cyber attacks is estimated to be $3.62 million dollars worldwide according to a report. Advanced Persistent Threat (APT) is a targeted cyber attack that is tailored, proceeds at a stealth and has a high objective. The state of the art security monitoring tools have failed in their attempts to detect APT. Therefore, there is a need for a solution that is fool-proof in the detection of an APT. This paper proposed the use of cyber kill chain to detect the various attack methodologies used in an APT campaign and to correlate and predict the existence of an APT attack. APT attack deploys various attack techniques which are mapped to the stages of the cyber kill chain. For each of those techniques, an attack detection methodology has been proposed in this paper. The detection result of each of these methodologies, will now be correlated in the correlation module to ascertain whether there is an ongoing APT attack and raise an alert. The result from this research work will be evaluated against a current related work. This research will therefore advance the state of the art in APT attack detection.

KEYWORDS: Advanced Persistent Threat (APT), cyber kill chain (CKC), data breach, cyber attack, APT detection.

I. Introduction

Complex, long-term set of actions aimed against specific persons, organizations or companies is referred to as Advanced Persistent Threat (APT). Adversaries often study their targets for months before launching the attack. Adversaries maintain stealth and can exfiltrate data for a long period of time (Rot and Olszewski, 2017). The targets are mostly companies, government agencies and even individuals. The APT attacker can be an individual, organized crime group or nation state actors. It could take days or years before an APT attack is detected. When the attacker discovers that he has been detected, he might become more violent, change the method of attack or resort to an alternate course of action (Baksi and Upadhyaya, 2017). The state of the art security monitoring tools have failed in their task to detect APT (NIS Platform, 2014; Oprea *et al*., 2015; ENISA, 2018). There is therefore the need for an effective APT detection framework.

II. The Cyber Kill Chain (CKC)

The CKC also called the Intrusion Kill Chain (IKC) is a model that describes the phases of intrusions proposed by Hutchins *et al.* (2011). The CKC is a seven phase model that describes the stages APT actors follow to achieve their objectives.

a. Reconnaisance

This is the planning stage of the cyber attack. During this stage the attackers conduct a research on their target. Attackers harvests email addresses, identify employees on social media, gather press releases, contract awards, conference attendee lists and search for corporate Internet facing servers (Martin, 2015). The target can be an individual or an organization. Reconnaisance can be broken down into target identification, selection and profiling. Reconnaisance can be passive or active. In passive reconnaisance, the target is unaware of the process. Active reconnaisance on the other hand involves a deeper probing of the victim's information technology infrastructure which may trigger alert of the victim's security monitoring tools.

Table 1 gives examples of reconnaissance techniques and techniques used for both passive and active attacks.

Table 1: Reconnaissance Techniques (Yadav and Mallari, 2016)

	Reconnaissance Techniques	Type of Reconnaissance	Techniques Used
1.	Target identification and selection	Passive	Domain names, WHOIS records from APNIC, RIPE and ARIN
2.	Target profiling		
	Target social profiling	Passive	Social Networks, Public Documents, Reports and Corporate Websites
	Target system profiling	Active	Pingsweeps, Fingerprinting, Port scanning and services
3.	Target validation	Active	SPAM messages, Phishing mails and social engineering.

b. Weaponization

This is an operation preparation stage. Automated tools are used in generation of malware. A weaponizer is developed by coupling malware and exploit into a deliverable payload. A decoy document is chosen to be delivered to the victim for file based exploits (Martin, 2015). It is specifically the binding of software/application exploits with a Remote Access Trojan (RAT). Weaponization involves the use of two components; RAT and exploits. RAT is the payload of the cyber weapon. RAT is a sofware that is installed on the victim's machine to give access to the attacker. RAT is made up of two parts; a client and a server. Exploits serve as a carrier for RAT and facilitates the execution of the RAT. The main reason behind the use of RAT is to avoid victim's attention while establishing a stealth backdoor access using RAT (Yadav and Mallari, 2016).

c. Delivery

At this stage the operation is launched. The malware is conveyed to the target at this stage. Some user actions may be required like downloading and executing malicious files or visiting malicious web pages on the Internet. Some attacks are performed without user interactions by exploiting network devices e.g. CVE-2014-3306, CVE-2014-9583 (Mitre, 2014). Multiple delivery methods are usually employed since no single method can guarantee 100% success (Yadav and Mallari, 2016).

d. Exploitation

This stage is where the exploits is triggered (Yadav and Mallari, 2016). The attackers exploits a vulnerability to gain access. Exploits may be triggered by an adversary for server based exploits or by a victim through opening an attachment of malicious email or clicking a malicious link (Martin, 2015). Exploits might not usually be successful unless the following conditions are matched.

1. Victim is using the operating system or software for which the exploit has been created.
2. The software/operating system not updated or upgraded to the newest version
3. End host protection mechanism should not be able to detect the exploit or payload

Table 2 gives examples of delivery mechanisms and their peculiar characteristics.

Table 2: Delivery Mechanism (Yadav and Mallari, 2016).

	Delivery Mechanism	Characteristics
1	Email attachments	Enticing email content is composed to appeal to the user.
2.	Phishing attacks	Fake websites is used to harvest user credentials
3.	Drive by downloads	Intentionally or unintentionally victim is lured into

		downloading malicious content.
4.	USB/Removable media	Malware is kept in a USB device to attack victims
5.	DNS cache poisoning	Vulnerabilities in DNS are utilized to divert internet traffic from legitimate

The payload then connects to its Command & Control (C & C) counterpart to inform about successful execution and commands to execute. Exploits are made from the Common Vulnerabilities and Exposures (CVEs) publicly made available (Yadav and Mallari, 2016). Exploits are also made available from vulnerabilities discovered through fuzzing methodology (Yadav and Mallari, 2016).

e Installation

Installation stage is when the attackers install a persistent backdoor or implant in the victim environment to maintain access for an extended length of time (Martin, 2015). Malware utilizes droppers and downloaders to maintain stronghold on the victim's machine. Dropper installs and executes the malware on the target machine. Dropper first disables the endpoint protection on the device and hides the installed malware (Yadav and Mallari, 2016). Downloaders performs a similar function as droppers with the exception that they do not contain the malicious payload. The malicious payload is downloaded later when the downloader connects to a remote repository. Malware authors now employ the following techniques to stealthy stronghold and hidden installations (Yadav and Mallari, 2016).

1. Anti debugger and anti emulation
2. Anti antivirus
3. Rootkit and bootkit installation
4. Targetted delivery
5. Host based encrypted data exfiltration

f. Command & Control

In this stage a command channel is opened by a malware to enable the attackers control the victim remotely. A two way communication channel to the Command & Control (C2) infrastructure is opened. The C & C channels are mostly over web, DNS and email protocols. The C2 might be owned by the attacker or might be another victim's network (Martin, 2015). The aim of the C & C channels is to provide a secret channel for issuing commands to infected machines. The three types of the C & C communication structures are:

1. Centralized structure
2. Decentralized structure
3. Social network based structure

Attackers employ the following techniques to achieve stealth and unidentified communication channel (Yadav and Mallari, 2016).

1. Internet Relay Chat (IRC)
2. TCP/HTTP/FTP
3. Steganography
4. The Onion Router (TOR)

Attackers also deploy the following techniques to remain stealth (Yadav and Mallari, 2016).

1. DNS Fast Flux
2. DNS as a medium
3. Domain Generation Algorithm

g. Act on Objectives

The attackers at this stage have a hands-on keyboard access to their victim and can now accomplish their mission. The objectives ranges from harvesting user credentials, escalation of privileges, information gathering, lateral movement, data exfiltration (Martin, 2015) and espionage. Sometimes attack might be physical like in the case of Stuxnet (Angle *et al*., 2017). Attack can lead to the destruction of system hard drive or device drivers. Attacker may lead to the Central Processing Unit (CPU) using its highest capability for a very long duration which leads to the damage of the CPU hardware (Yadav and Mallari, 2016).

Although there are other models that seek to describes the stages of an APT attack, for example the Mandiant (Aldridge, 2016), the Dell Secureworks models (Dell SecureWorks, 2012) and the APT attack lifecycle (Ghafir and Prenosil, 2016) the CKC model is the most widely known and more frequently cited (Herlow, 2015).

III. Comparison of Classification Algorithms Prediction Accuracy

Kotsiantis (2007) reviewed supervised machine learning classification algorithms. In the review the study ranked Support Vector Machine (SVM) as the algorithm with the highest accuracy followed by Neural Networks, Decision Trees, kNN and Rule Learners on the same position followed by Naive Bayes in the last position.

Wei-Chih and Yu (2009) performed email spam filtering using Naive Bayesian, SVM with RBF kernel, Linear kernel, SVM using Taguchi Method and SVM using grid search. SVM using grid search has the highest accuracy followed by SVM using Taguchi Method which is proposed work of Wei-Chih and Yu (2009).

Mezghani *et al*. (2010) compared the prediction accuracy of SVM kernels with three other popular learning algorithms: Naive Bayes (NB), Decision Tree C4.5 and Multi Layer Perceptron (MLP) for speaker identification. SVM trained using polynomial kernel emerged the best for speaker identification tasks and SVM was the best compared with other algorithms.

Amami *et al*. (2012) performed an empirical comparison of SVM, K-Nearest Neighbour, Naive Bayes, Quadratic Bayes Normal and Nearest Mean on TIMIT vowel data for a multi-class recognition problem. SVM using RBF kernel achieved the best performance amongst the different classifiers evaluated.

Yasin and Abuhasan (2016) utilized five classification algorithms using Random Forest, J48, Naive Bayes, SVM and Multi-Layer Perceptron (MLP) for phishing email detection. Random Forest gives the best result followed by J48.

Agarwal and Kumar (2016) performs spam filtering with SVM using different kernel functions (linear, polynomial, RBF, sigmoid) and different parameters (C-SVC, NU-SVC). The best result is achieved with linear kernels on C-SVC.

Hong *et al*.(2017) compare SVM kernel functions for landslide susceptibility mapping. The results of the study revealed that SVM-RBF is the most suitable for landslide susceptibility assessment.

According to Kotsiantis (2007) there is no a single learning algorithm that can evenly do better than other algorithms over all datasets. Whenever we are confronted with the decision of selecting the precise algorithm for a classification problem the easiest way is to approximate the preciseness of the candidate algorithms on the problem and choose the one that is more precise (Kotsiantis, 2007).

From the works reviewed, it could be clearly seen that SVM is leading in terms of accuracy followed by Random Forest. This study will therefore test the prediction accuracy of the SVM kernels (linear, polynomial, RBF, sigmoid) using C-SVC and NU-SVC parameters, SVM using Taguchi Method and SVM using grid search in the prediction module to predict the APT attack.

IV. Related Work

Sharma *et al*.(2016) proposed a distributed framework architecture for the detection of APT. The work focuses on providing intrusion detection framework especially for APT attack detection. The aim of the work is to offer a new intrusion detection system that processes the network traffic and that is intelligent enough to identify an APT attack. The recognition of APT attack depends on the relationship between the events that are generated by different

classifier methods. The study designed a new framework architecture for intrusion detection system of network traffic for APT attacks in a distributed environment. The intrusion detection process was performed in a distributed environment in the trusted platform module where it stays hidden from the attackers. Initially network traffic packets to identify all possible strategies that could be utilized as a part of an APT attack cycle are collected, processed and analyzed using four different recognition methods which are independent of each other. The outputs of these classifier methods are then submitted to the next stage which is the event correlation phase. The event correlation modules takes all events provided by the outputs of all detection classifier methods as an input and correlates all of them individually as indicated by the principles specified by the system admin to raise alert on APT attack discovery. The outputs is then submitted to the next stage which is the voting stage. In the voting stage voting service analyzes and determines final result based on the information provided by event correlation for the different methods. The rationale behind the voting techniques is to lessen the rate of false positives and enhances the accuracy of the detection. Four classification methods used for the detection are genetic programming, classification and regression trees, support vector machines and dynamic bayesian game model. The proposed methodology was evaluated with results from the individual classifiers. The study did not validate the proposed approach by comparing it against other APT detection works.

Moya *et al*.(2017) proposed the use of expert knowledge and data analysis to detect APT. The accuracy of the proposed model was measured with several samples using bayesian techniques, decision trees and artificial neural networks. Decision trees shows better fitness. Validation tests was performed over all the samples and then selected some variables to be assessed: accuracy of the model created with decision trees, improvement over the trivial model, sensitivity to harmful behaviour, resistance accuracy of the model, resistance improvement over trivial model and resistance sensitivity to harmful behaviour. To choose the best possible proportion of activity logs the study developed descriptive analysis over each sample with the values of the variable described in the study (boxplots and arithmetic mean). The sample with the highest mean points to the most adequate model. After the analysis, the final system is run with the best sample and is able to alert of log registers that might be related with APTs. The results of the analysis revealed that ID3-C4.5 decision tree provides better accuracies and errors than Naive Bayes and probabilistic neural network. This led to the selection of the decision tree to detect anomalous behaviors in the network activity (Moya *et al*., 2017). There was no evaluation of the proposed methodology to show how effective it is against other proposals.

Ghafir *et al*.(2018) proposed MLAPT for the detection of APT using machine learning correlation analysis. The proposed methodology comprises of three parts: threat detection, alert correlation and attack prediction. The threat detection uses eight methods to detect various steps used in a multi-step APT attack. This methods are disguised executable 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 outputs of this phase generates events from the detection methods used. The events correlation phase correlates the events produced in the first phase with one APT attack scenario. The event correlation phase consists of three steps: alert filter (AF), alerts clustering (AC) and correlation indexing (CI). The aim of the event correlation phase is to reduce the false positive rate of the detection system. The attack prediction phase implements a machine learning based prediction module based on a historical record of the monitored network. The prediction module employs four classification algorithms thus: decision tree learning, support vector machine, k-nearest neighbours and ensemble learning. No reason was specified for choosing those classification algorithms. SVM has the highest degree of prediction accuracy and recommended to be used by the network security team to predict APT. The attack prediction module is aimed to help the network security team to predict APT attack. Other limitations of the proposal is that the attack detections modules did not adequately captured all the attack techniques used during APT. So there is a room for improvement with regards to that.

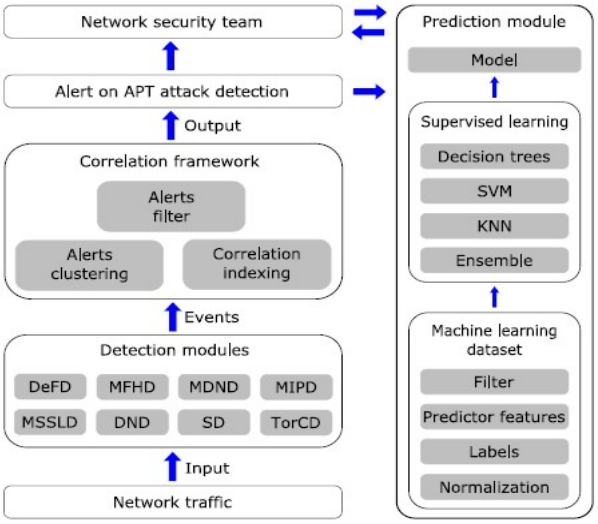

Figure 1: The Architecture of MLAPT (Ghafir *et al*., 2018)

This research work seeks to improve the work of Ghafir *et al*., (2018) by increasing the attack detection methodologies. The attack detection methodologies that will be added have been presented side by side with the proposed attack detection methodologies of Ghafir *et al*., (2018) in table 3. In the prediction module, Ghafir *et al*., (2018) utilizes four classification algorithms Decision trees, SVM, KNN and Ensemble. The classification algorithms that yields the highest detection accuracy is SVM using the linear kernel. This work seeks to improve upon the prediction accuracy by proposing to compare the prediction results of the different kernels of SVM (linear, polynomial, RBF, sigmoid), SVM using Taguchi Method and SVM using grid search. The different types of SVM have been depicted in figure 6 as SL(SVM linear), SP(SVM polynomial), SR(SVM RBF), SS(SVM sigmoid), STM(SVM using Taguchi Method) and SGS(SVM using grid search).

V. Methodology

1. Aim and Objectives

The aim of this research is to develop an effective framework for the detection and prediction of advanced persistent threat (APT) based on the cyber kill chain (CKC).

The specific objectives of this research are to:

a. design an effective APT detection and prediction framework
b. develop attacks detection modules for the attacks in the cyber kill chain stages
c. develop correlation module for the APT attacks detection
d. develop an APT prediction module
e. evaluate the effectiveness of the proposed APT detection framework with that of Ghafir *et al*., (2018)

2. System Architecture

The system architecture of the proposed APT detection framework based on the CKC is presented in figure 6.

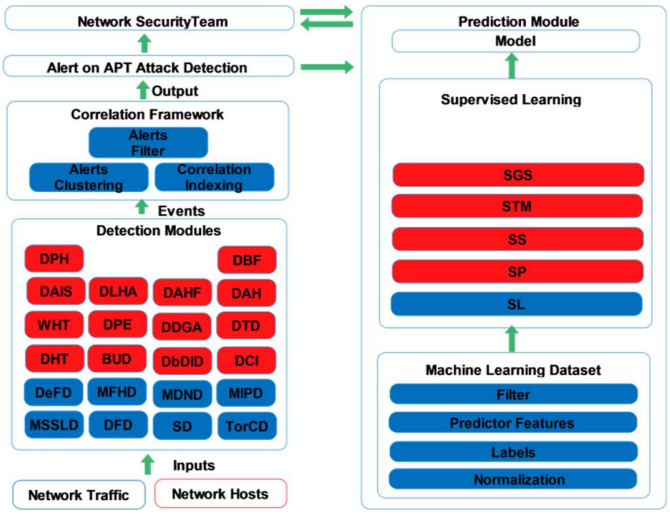

Figure 2: Proposed APT Detection System Architecture

The proposed framework is an extension of the work of Ghafir *et al*. (2018). Network traffic and network hosts will be monitored by the attack detection modules for attack. Fourteen detection methodologies have been added to the proposed work of Ghafir *et al*. (2018) to detect the several attacks employed during an APT campaign. Table 3 compares the attack detection methodologies proposed by Ghafir *et al.* (2018) which are based on the APT attack lifecycle against the attack detection methodologies proposed by this study to build on the work of Ghafir *et al*. (2018) and which are based on the CKC. These attack detection modules (methodologies) generates events which will then be fed to the correlation framework. The correlation framework correlates the events generated from the attack detection modules to one APT attack scenario. The rationale behind using the correlation framework is to lower the false positive rate of the system (Ghafir *et al*. 2018). The output from the correlation framework is an alert (event) on APT detection that will be channeled to the network security team who will utilize it to predict APT attack. The prediction module, predicts whether the event generated by the correlation framework will grow to a full APT attack scenario in the future based on the attribute of the event generated by the correlation framework. This will enable the network security team to perform more analysis on the corresponding two suspicious events (the event from the correlation framework and the full APT attack scenario) and stop the attack before it grows to a full APT (Ghafir *et al*. 2018).

Table 3: Comparison of the Proposed Cyber Kill Chain based APT Detection Methodology against Ghafir *et al*. (2018)

Cyber Kill Chain	APT Attack Lifecycle (Ghafir *et al*., 2018)	Methods of Detection Proposed by (Ghafir *et al*., 2018)	Methods of Detection Proposed by this study
Reconnaissance	Intelligence Gathering	None	Use of DNS Honey Tokens (DHT), Detection of Access to robots.txt Files, Detection of Access to Invisible Links, Detection of access to HTML Honey tokens (collectively referred to Detection of Web server Honey Tokens (WHT)) (Kollitris, 2015)

Weaponization		None	None
Delivery	Initial Compromise (Point of Entry)	Malicious Domain Name Detection (MDND), Disguised exe File Detection (DeFD), Malicious File Hash Detection (MFHD)	Malicious Domain Name Detection (MDND) (Ghafir *et al*., 2018), Disguised Exe File Detection (DeFD) (Ghafir *et al*., 2018), Malicious File Hash Detection (MFHD) (Ghafir *et al*., 2018), Bad USB Detection (BUD), Drive by Downloads/Install Detection (DbDID)
Exploitation	Initial Compromise (Point of Entry)		
Installation	Initial Compromise (Point of Entry)		Detection of Code Injection (DCI), Detection of API Hooking (DAH), Detection of Privilege Escalation (DPE)
Command & Control	Command & Control	Malicious SSL Detection (MSSLD), Malicious IP Address Detection (MIPD), Domain Flux Detection (DFD).	Detection of a Connection to a TOR (TorCD), Detection of a DGA (DDGA), Malicious SSL Certificate Detection (MSSLD) (Ghafir *et al*., 2018), Malicious IP Address Detection (MIPD) (Ghafir *et al*., 2018), Domain Flux Detection (DFD) (Ghafir *et al*., 2018), DNS Tunneling Detection (DTD).
Act on Objectives	Lateral Movement, Asset/Data Discovery and Data Exfiltration		Detection of Access to Internet Sink (DAIS) (Kollitris, 2015), Detection of Logging to Honey Account (DLHA) (Kollitris, 2015), Detection of Access to Honey Files (DAHF) (Kollitris, 2015), Tor Connection Detection (TorCD) (Ghafir *et al*., 2018), Detection of Pass the Hash (DPH), Detection of Brute Force Attack (DBF).

Figure 5 presents the system architecture for the proposed APT detection framework. The detection results of each of the method in the detection module will generate an event which will serve as an input to the correlation framework. The correlation framework aim is to find events that are related and belonging to one APT attack situation (Ghafir *et al*., 2018). To find out the probability of the early alerts leading to a complete APT attack, a machine learning based prediction module will be used in the final stage (Ghafir *et al*., 2018).

VI. Conclusion

The result of this research work will be a framework that will effectively detect APT. The attack detection modules proposed in the study will be developed and evaluated against recent study. The events generated from the attack detection modules will be fed to the correlation framework and subsequently the various SVM kernels will be used to develop a model to predict APT attack. The model that supersedes in accuracy will be recommended for use by the network defense team. The APT prediction accuracy of the proposed framework will be evaluated against the work of Ghafir *et al*. (2018).

An effective framework capable of effectively detecting APT based on cyber kill chain has been proposed. The proposed study builds upon the work of Ghafir *et al*. (2018). This will be achieved by increasing the number of attack detection modules in the proposed framework and the use of several SVM kernels have also been proposed to predict APT attack.

REFERENCES

Agarwal, D. K., & Kumar, R. (2016). Spam Filtering using SVM with different Kernel Functions. International Journal of Computer Applications, 136(5), 16-23. Retrieved from https://www.ijcaonline.org/research/volume136/number5/agarwal-2016-ijca-908395.pdf

Aldridge, J. (2016). Remediating Targeted-threat Intrusions. Fire Eye. Retrieved from https://www2.fireeye.com/rs/848-DID-242/images/WP-Remediating-Intrusions.pdf?mkt_tok=eyJpIjoiT1RNMk1HWmxNalF3WkRBNSIsInQiOiJ6dVwvVXR0cGFZS2UzaFF1UlBsdUZ3Sjl0b2NUbVJWTWpIK3dLS04yazUxcFowN0dJQU9rUlM4ZnF2cGRsMStDb2paU3o5RzFyXC9LdnZyQVpWS29EbUdNaE1ia0p2QXFmQn

Amami, R., Ayed, D. B., & Ellouze, N. (2012). An Empirical Comparison of SVM and Some Supervised Learning Algorithms for Vowel Recognition. International Journal of Intelligent Information Processing (IJIIP), 3(1.6), CoRR. doi:doi: 10.4156/IJIIP

Angle, M. G., Madnick, S., & Kirtley, J. (2017). Identifying and Mitigating Cyber Attacks that Could Cause Physical Damage to Industrial Control Systems . IEEE Power and Energy Technology Systems Journal, 1-10 .

Baksi, R. P., & Upadhyaya, S. J. (2017). Kidemonas: The Silent Guardian. SKM'17, (pp. 1-6). Tampa, FL, USA

Dell SecureWorks. (2012). Lifecycle of an Advanced Persistent Threat. Dell. Retrieved from http://www.redteamusa.com/PDF/Lifecycle%20of%20an%20Advanced%20Persistent%20Threat.pdf

ENISA. (2018). ENISA Threat Landscape Report 2017: 15 Top Cyber-Threats and Trends. Heraklion, Greece: ENISA. doi:DOI 10.2824/967192

Kotsiantis, S. B. (2007). Supervised Machine Learning: A Review of Classification. Informatica, 249-268.

Ghafir, I., & Prenosil, V. (2016). Proposed Approach for Targeted Attacks Detection. In H. Sulaiman, M. Othman, M. Othman, Y. Rahim, & N. Pee, Lecture Notes in Electrical Engineering, (Vol. 362, pp. 73-80). Springer, Cham.

Ghafir, I., Hammoudeh, M., Prenosil, V., Han, L., Hegarty, R., Rabie, K., & Aparicio-Navarro, F. J. (2018). Detection of Advanced Persistent Threat using Machine-Learning Correlation Analysis. Future Generation Computer Systems, 89, 349-359. doi:https://doi.org/10.1016/j.future.2018.06.055

Herløw, L. (2015). Detection and Prevention of Advanced Persistent Threats: Evaluating and Testing APT Lifecycle Models Using Real World Examples and Preventing Attacks through the Use of Mitigation Strategies and Current Best Practices. Denmark: DTU Compute: Department of Applied Mathematics and Computer Science.

Hong, H., Pradhan, B., Bui, D. T., Xu, C., Yousseff, A. M., & Chen, W. (2017). Comparison of Four Kernel Functions used in Support Vector Machines for Landslide Susceptibility Mapping: A Case Study at Suichuan Area (China). Geomatic, Natural Hazards and Risk, 8(2), 544–569. doi:http://dx.doi.org/10.1080/19475705.2016.1250112

Hutchins, E. M., Cloppert, M. J., & Amin, R. M. (2011). Intelligence-Driven Computer Network Defense Informed by Analysis of Adversary Campaigns and Intrusion Kill Chains. 6th Annual International Conference on Information Warfare and Security (pp. 1 - 14). Washington DC: Academic Conferences and Publishing International.

Kollitris, N. V. (2015). Detecting Advanced Persistent Threats through Deception Techniques. Greece: Information Security and Critical Infrastructure Protection (INFOSEC) Laboratory.

Mandiant. (2004). APT1: Exposing One of China's Cyber Espionage Units. Mandiant.

Martin, L. (2015). Gaining the Advantage: Applying Cyber Kill Chain Methodology to Network Defense. Lockheed Martin Corporation.

Mezghani, B. A., Boujelbene, Z., & Ellouze, N. (2010). Evaluation of SVM Kernels and Conventional Machine Learning. International Journal of Hybrid Information Technology, 3(3), 23-34. Retrieved from http://www.sersc.org/journals/IJHIT/vol3_no3_2010/3.pdf

Mitre. (2014). Search Results. Retrieved May 5, 2018, from Common Vulnerabilities and Exposures: https://cve.mitre.org/cgi-bin/cvekey.cgi?keyword=CVE+-2014-3306

Moya, J. R., García, N. D., Díaz, R. Á., & Tamargo, J. L. (2017). Expert Knowledge and Data Analysis for Detecting Advanced Persistent Threats. *Open Mathematics, 15*(1), 1108-1122. doi:https://doi.org/10.1515/math-2017-0094

NIS Platform. (2014). State of the Art of Secure ICT Landscape. NIS. Retrieved from https://resilience.enisa.europa.eu/nis-platform/shared-documents/wg3-documents/state-of-the-art-of-the-secure-ict-landscape/at_download/file

Oprea, A., Li, Z., Yen, T.-F., Chin, S., & Alrwais, S. (2015). Detection of Early-Stage Enterprise Infection by Mining Large-Scale Log Data. 2015 45th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (pp. 45-56). Rio de Janeiro, Brazil: IEEE. doi: 10.1109/DSN.2015.14

Rot, A., & Olszewski, B. (2017). Advanced Persistent Threats Attacks in Cyberspace Threats, Vulnerabilities, Methods of Protection. Federated Conference on Computer Science and Information Systems. 12, pp. 113–117. Prague, Czech Republic: ACSIS. doi:DOI: 10.15439/2017F488

Sharma, P. K., Moon, S. Y., Moon, D., & Park, J. H. (2017). DFA-AD: A Distributed Framework Architecture for the Detection of Advanced Persistent Threats. *Cluster Computing, 20*(1), 597–609. doi:https://doi.org/10.1007/s10586-016-0716-0

Wei-Chih, H., & Yu, T.-Y. (2009). E-mail Spam Filtering Using Support Vector Machines with Selection of Kernel Function Parameters. 2009 Fourth International Conference on Innovative Computing, Information and Control (pp. 764-767). Kaohsiung, Taiwan: IEEE. doi:DOI: 10.1109/ICICIC.2009.184

Yadav, T., & Mallari, R. A. (2016). Technical Aspects of Cyber Kill Chain. 1 - 7.

Yasin, A., & Abuhasan, A. (2016). An Intelligent Classification Model for Phishing Email Detection. International Journal of Network Security & Its Applications (IJNSA), 8(4), 55-72. doi:DOI: 10.5121/ijnsa.2016.8405