12-2022

GUIDE FOR THE COLLECTION OF INSTRUSION DATA FOR MALWARE ANALYSIS AND DETECTION IN THE BUILD AND DEPLOYMENT PHASE

Musa Gassama

Guide for the Collection of Intrusion Data for Malware Analysis and Detection in the Build and Deployment Phase

by

Musa Gassama

A Thesis

Submitted to the Graduate Faculty of

St. Cloud State University

in Partial Fulfillment of the Requirements

for the degree of

Master of Science

in Information Assurance

December, 2022

Thesis Committee:
Lynn A. Collen, Chairperson
Jim Chen
Abu Hussein Abdullah
Akalanka B. Mailewa

Abstract

During the COVID-19 pandemic, when most businesses were not equipped for remote work and cloud computing, we saw a significant surge in ransomware attacks. This study aims to utilize machine learning and artificial intelligence to prevent known and unknown malware threats from being exploited by threat actors when developers build and deploy applications to the cloud. This study demonstrated an experimental quantitative research design using Aqua. The experiment's sample is a Docker image. Aqua checked the Docker image for malware, sensitive data, Critical/High vulnerabilities, misconfiguration, and OSS license. The data collection approach is experimental. Our analysis of the experiment demonstrated how unapproved images were prevented from running anywhere in our environment based on known vulnerabilities, embedded secrets, OSS licensing, dynamic threat analysis, and secure image configuration. In addition to the experiment, the forensic data collected in the build and deployment phase are exploitable vulnerability, Critical/High Vulnerability Score, Misconfiguration, Sensitive Data, and Root User (Super User). Since Aqua generates a detailed audit record for every event during risk assessment and runtime, we viewed two events on the Audit page for our experiment. One of the events caused an alert due to two failed controls (Vulnerability Score, Super User), and the other was a successful event meaning that the image is secure to deploy in the production environment. The primary finding for our study is the forensic data associated with the two events on the Audit page in Aqua. In addition, Aqua validated our security controls and runtime policies based on the forensic data with both events on the Audit page. Finally, the study’s conclusions will mitigate the likelihood that organizations will fall victim to ransomware by mitigating and preventing the total damage caused by a malware attack.

Keywords: Artificial Intelligence, Bot, Machine Learning, Phishing, Ransomware, Spyware, Trojans, Virus, Vulnerabilities, Worms

Acknowledgments

First and foremost, I thank Allah (S.W.T.) for showering his blessings throughout my master's thesis. Second, I want to express my deep gratitude to my thesis supervisor Professor Collen Lynn A., for her continuous support of my master's study and related research. Most importantly, her patience, motivation, and knowledge guided me throughout the research. To Professor Jim Chen, Abu Hussein Abdullah, and Akalanka, thank you for the positive feedback to ensure I complete my research quickly. Everyone significantly influenced my career achievement and encouraged me to study and grow.

Finally, I am incredibly thankful to my parents for their unconditional love, prayers, and sacrifice. My parents have always encouraged me to be a leader and think for myself. My wife deserves special gratitude for understanding and encouraging me to finish my master's thesis.

Table of Contents

Chapter Page

List of Tables

Table Page

List of Figures

Figure Page

Chapter I: Introduction

Malware are meant to exploit the vulnerability and exposure of various software product such as applications, Operating Systems (OS), drivers, etc. The popularity of OS and applications make them a hot target for malware attacks (Kumar & Subbiah, 2022). We need to minimize malware threats to protect sensitive information such as financial accounts, social security numbers, login I.D.s, and medical data. As malware detection techniques have evolved, attacks have increased as companies such as Google, Yahoo, and others have established strategies to safeguard their networks. However, as malicious software becomes more prevalent, there is a more significant requirement for low-cost host-based security techniques to prevent it from spreading. Because a few hundred million users are at risk daily, it is difficult to avoid all sorts of attacks. For example, malware is detected using various methods, so it must be attacked from multiple angles to ensure it is detected effectively and simultaneously. In addition, cybercriminals have taken advantage of malware to take over computers and steal confidential information for monetary gains. One of the most challenging tasks for cybersecurity specialists is responding to incidents rapidly while reducing risk and damage costs. Statistics from IBM X-Force Incident Response and Intelligent Services show that destructive malware attacks experienced by organizations are way too costly. Including the cost of equipment replacement, lost productivity and other damages make malware attacks a real disaster for companies (Ben Abdel Ouahab et al., 2020)

This paper uses Aqua to analyze and detect malware, exploitable vulnerabilities, and sensitive data. With Aqua, developers can securely build and deploy into the cloud because it prevents attacks and stops them as they happen. As a critical contribution, we find that Aqua allows us to define, configure and manage runtime policies (*Policies*, n.d.). And these runtime policies reduce attackers' ability to operate with policies that permit or block workload activities (*Cloud Native Detection and Response CNDR*, n.d.). Finally, we established evidence that our docker image doesn't contain severe vulnerabilities, malware, or sensitive data. Finally, our executive summary document shed some actionable advice to security executives on protecting against a new and growing breed of attacks. This study aims to utilize machine learning and artificial intelligence to prevent known and unknown malware threats from being exploited by threat actors when developers build and deploy applications to the cloud for an organization.

Problem Statement

Malware can quickly access critical corporate information by infiltrating the server system. It has been predicted that the total loss of organizations due to ransomware will be around $20 billion in 2021, and the new organization will be hit by those attacks every 11 seconds (Oz et al., 2021). This will harm the company's operations in the market. However, malware may potentially cause hardware failure in rare circumstances. Therefore, malware analysis and detection are imperative to safeguard sensitive corporate information because they will provide actionable information by identifying and categorizing malware. By recording and identifying the virus through malware analysis, you obtain a plethora of knowledge that may be used to assist and avoid future incidents. This will protect businesses from being victimized by groups of hackers seeking monetary benefit.

Nature and Significance of the Problem

Malware proliferation on the Internet has increased significantly in the global community. Today, due to the sophisticated malware techniques used by malware perpetrators, zero-day attacks and false positives have become the most challenging problem in malware detection. Cybersecurity Ventures reported that the projected total damage caused by malware attacks was $3 trillion in 2015 and is likely to reach $6 trillion by 2021 (Alo et al., 2021). According to statistical data from the Independent IT-Security Institute, approximately 1001.51 million malware collected worldwide were examined and classified in the year 2019. There was a 17% increase in new malware detected compared to 856 million in 2018 (Mutalib et al., 2021). Modern machine learning and AI techniques will detect new malware variants and prevent hackers from exploiting them.

Deep Instinct estimates that the total damage cost of ransomware in 2019 exceeded the predicted USD 11.5 billion, stating that ransomware developers specifically targeted large enterprises due to their profitability (McDonald et al., 2022). As a result, organizations should take extra precautions to reduce their likelihood of becoming ransomware victims. One method of reducing this possibility is creating a plan based on all available information. The gap must be filled when there is a general lack of information.

WannaCry and NotPetya are two Ransomware that caused significant concern in the business world in 2017. On 12 May 201, the WannaCry outbreak began. The ransomware gained substantial media attention within hours, as it crippled several significant institutions and critical infrastructure in Europe, such as the United Kingdom's National Health Service, Deutsche Bahn, Renault, FedEx, and several other high-profile organizations. WannaCry affected over 300,000 businesses across 150 countries in the first few days of its outbreak (McDonald et al., 2022). The WannaCry attack forced firms outside of the I.T. industry to assess their security procedures and decide whether to upgrade their I.T. infrastructure. Its mainstream popularity has consequences unrelated to cybersecurity practice. The finance sector took advantage of the incident and experienced excess positive returns in cybersecurity exchange-traded funds because of WannaCry.

NotPetya ransomware was released in June 2017, shortly after WannaCry. The infection began as cybercriminals had managed to infiltrate the Ukrainian accountancy software update server used by an estimated 80% of companies in Ukraine. Following this, the attackers developed a backdoor in the accountancy software and pushed this out to all users through the update server they had gained control over. From this vulnerability, the attackers would deploy the ransomware, which would then spread further to other machines on the network using the EternalBlue exploit. This resulted in large-scale corporate infections, which were only furthered by the lack of SMB version 1's security precautions that should have been implemented before and after WannaCry's devastation. NotPetya attack was so severe that Merck & Co, an American pharmaceutical company, estimated that by the end of 2017, it had cost them $870 million in losses, a figure that would eventually increase to $1.3 billion when insurance claims were filed (McDonald et al., 2022).

Objective of the Study

The objective of the study is to:

- Research modern techniques and tools to detect and analyze malware
- Observe how specific pieces of malware behave to build defense mechanisms to safeguard an organization's network.
- Analyze malware similarities to understand how they differ from previously recognized ones.
- Make recommendations on swiftly detecting malware and preventing it from causing severe damage.

Study Questions/Hypotheses

1. What forensic data should be collected when performing malware analysis and detection?

Definition of Terms

To better understand this research, the following terms are defined in the context of this study.

Malware terminologies (*Glossary,"* n.d.):

Artificial Intelligence (AI): AI is a system's or an application's ability to correctly interpret and learn from data to achieve specific goals and tasks.

Bot: The word "bot" is a derivative of "robot." It usually pertains to one or more compromised machines controlled by a botmaster or herder to spam or launch DDoS attacks.

Machine Learning: Machine learning is a form or subset of artificial intelligence (AI) where computers use large data sets and statistical techniques to improve specific tasks without being manually reprogrammed.

Phishing: Phishing scams attempt to obtain your information by presenting themselves as legitimate websites, then asking for your password, credit card details, or other sensitive information.

Ransomware: Ransomware is a form of malware that locks you out of your device, encrypts your files, and then forces you to pay a ransom to get them back.

Spyware: Spyware is a type of malware that gathers information on a device and sends it to a third-party actor or organization that wouldn't usually have access.

Trojans: Trojans are programs that claim to perform one function but actually do another, typically malicious. Trojans can take the form of attachments, downloads, and fake videos/programs and, once active on a system, may do a number of things, including stealing sensitive data or taking control of the device.

Virus: A virus is a malware attached to another program (such as a document) that can replicate and spread after an initial execution on a target system where human interaction is required.

Vulnerabilities: A software vulnerability is a bug or error found in a cybersecurity system and is a point of weakness that cybercriminals can exploit.

Worms: Worms are malware-like viruses that do not need to be attached to another program to spread. (*Malwarebytes Glossary, n.d.*)

Summary

This chapter covers some background on malware analysis utilizing current approaches such as machine learning and AI and how businesses may utilize it to defend themselves against ransomware. The terms are well defined in this chapter, and the scope of the topic is being researched. In addition, some literature addresses malware's nature and relevance. The purpose of our research and the hypotheses are clearly explained in this chapter. In addition, the study aims to utilize machine learning and artificial intelligence to prevent known and unknown malware threats from being exploited by threat actors when developers build and deploy applications to the cloud for an organization.

Chapter II: Background and Review of Literature

Introduction

Malware is a program inserted into a system, usually covertly, to compromise the confidentiality, integrity, or availability of the victim's data, applications, or operating system or to otherwise annoy or disrupt the victim (Souppaya & Scarfone, 2013). Malware is further classified into non-exclusive groups such as trojans, viruses, adware, worms, and others. Thousands of malware are produced each year and utilized to target various businesses. Nuance Technologies revealed in March 2018 that the Notpetya infection had cost the company 92 million dollars. A group of hackers conducted a series of cyber-attacks against several institutions around the same time. The attackers took 31 gigabytes of data, with a total financial loss of almost three billion dollars. To steal login credentials and exfiltrate sensitive information, spear-phishing emails were used. Security researchers had to collect malware samples from the field to counteract malware-based attacks. These samples are then ‘detonated’ in a controlled environment, and their behavior ‘is logged.’ Using this behavioral data, security analysts map various malware to known indicators and means of attack. This will enable analysts and forensic experts to determine where to look for malware, the means and indicators of the attack, and, finally, the attack's outcomes. Recent years have seen rapid advancements in theory and deployment in machine learning (ML).

Machine learning technologies have achieved remarkable success in various applications, such as object detection and natural language processing. ML is also widely used in the security domain, including network intrusion detection and malware detection. Unlike traditional signature-based detection methods, ML-based detection systems exhibit high accuracy and can detect unseen and zero-day attacks (Hu et al., 2022, p. 1).

Background Related to the Problem

The scope of cybersecurity issues extends to the security of IT systems deployed in enterprises and broader digital networks, including critical national infrastructures. Unfortunately, preliminary security surveys by governments such as Australia Cyber Security Centre (ACSC) in 2020 show an increasing number of cyber threats targeting enterprises, but with a lack of information about the characteristics of the attacks and their possible impacts. Therefore, it is essential to analyze existing cybersecurity threats, vulnerabilities, and their solutions with a comprehensive view of cybersecurity, to gain a complete picture of the cybersecurity practices of medium-sized enterprises (Nagahawatta et al., 2021, p. 3). Malware can be categorized into various types, such as viruses, worms, Trojans, rootkits, ransomware, etc. Malware variants can steal confidential data, initialize distributed denial of service (DDoS) attacks, and perform disruptive damage to computer systems (Aslan & Yilmaz, 2021). New malware variants use concealing techniques such as encryption and packing to remain invisible in the victim's system. Those new variants spread by exploiting human trust as an infection vector. For instance, opening email attachments, downloading fake applications, and visiting and downloading files from phony websites are well-known methods of malware-spreading vectors (Aslan & Yilmaz, 2021). The Internet has been used in many fields like e-commerce, online education, banking, financial services, social media, and communication. According to Sahin & Bahtiyar (2020), 3 billion people use the internet daily. Because of economic reasons and benefits, people with cyber-criminals try to take advantage of themselves. To protect the computer systems, we must detect malware as soon as it infects them; otherwise, our passwords and files will be stolen and computers inoperable. Malware detection is analyzing a suspicious file and identifying whether it is malware or benign. According to Aslan & Yilmaz (2021), detecting malware requires three steps of operations:

1. Malware files are analyzed with appropriate tools.
2. Static and dynamic features are extracted from the analyzed files.
3. Features are grouped in specific ways to separate malicious software from benign.

To increase the detection rate, different sciences and techniques, including data science, machine learning, and heuristic, as well as technologies such as cloud computing, big data, and blockchain, are used in these processes. Different malware detection approaches use the above techniques and technologies. These approaches are mainly signature, behavior, model checking, and heuristic-based detection.

Literature Related to the Problem

Most companies have adopted cloud computing. According to the 2020 Flexera survey on cloud computing trends, 93% of enterprises have a multi-cloud strategy, depending on their regulatory requirements and availability, costs, and data sovereignty needs. These enterprises have, on average, 2.2 public and 2.2 private clouds. Although cloud providers use various security mechanisms and tools, they are targeted by attackers that use sophisticated malware to perform cyber-attacks. The 2019 Netskope cloud cybersecurity report states that the top three cloud security challenges are data privacy, data loss, and data leakage. The most common cloud vulnerability exploited by attackers is associated with insecure interfaces and APIs. According to the Check Point 2019 cloud security report, 15% of the surveyed organizations confirmed a cloud security incident, and 25% do not know whether they have been breached. Furthermore, Symantec reported that nearly two-thirds of the security incidents investigated in 2019 occurred at the cloud level (Panker & Nissim, 2021).

Federated learning for malware detection in IoT devices article states that by 2025, forecasts estimate there will be about 64 billion IoT devices online (Rey et al., 2022). One strategy that has gained relevance when detecting devices corrupted by malware is monitoring device activities to generate behavioral fingerprints or profiles.

For example, fingerprints can detect deviations caused by cyberattacks or malicious software modifications. In IoT devices, heterogeneous behavior sources such as network communications, resource consumption, software actions, events, or user interactions can be monitored (Rey et al., 2022).

The increasing use of smartphones and tablets has caused cybercriminals to change their attack tactics to mobile devices. The growth of Android has attracted cybercriminals to create malicious applications that steal sensitive information that affects mobile systems. Some elements deployed by criminals, such as social engineering, find vulnerabilities in the mobile operating system, thereby planning attacks (Mohamad Arif et al., 2021). According to the Kaspersky Lab report, malware attacks doubled in 2018, totaling 116.5 million, while 66.4 million in 2017. Of the total malware detected, more than 99.6% targeted Android (Mohamad Arif et al., 2021). The top 20 Android malware detections in 2018 are listed in Table 1. Android.Adware.AdultSwine is the most common, accounting for 17.29 percent of all detections, yet it is still in the moderate range.

Table 1

Android malware detected in 2018 – Top 20

Android malware detected in 2018 – Top 20.

Android Malware	Percentage	Level	Rank
Android.Adware.AdultSwine	17.29	Moderate	New
Android.Adware.Uapush.A	13.98	Moderate	1
Android.Trojan.Leech.d	4.69	High	20
Android.Trojan.AndrClicker.D	4.41	High	7
Android.Spyware.mSpy	4.11	High	12
Android.MobileSpyware.FlexiSpy	3.62	High	22
Android.Trojan.Xgen.FH	3.12	High	15
Android.InfoStealer.Adups	3.03	High	13
Android.Trojan.Rootnik.i	3.01	High	10
Android.Trojan.Triada	2.76	High	New
Android.Trojan.Gmobi.a	2.61	High	New
Android.BankingTrojan.Marcher.A	2.39	High	4
Android.BankingTrojan.Acecard.m	2.15	High	18
Android.Trojan.HiddenApp	2.08	High	28
Android.Trojan.Sivu.C	2.06	High	5
Android.Trojan.HiddnAp.AE	1.88	High	New
Android.Worm.ADB.miner	1.48	High	New
Android.BankingTrojan.FakeCarrierMMS	1.46	High	New
Android.Trojan.Xiny.19.origin	1.46	High	11
Android.Test.FakeMalwareTomTom	1.19	High	5

It is predicted by 2021, there will be an increase in ransomware five times compared to the current attack rate. Furthermore, cybercrime damages are anticipated to cost $6 trillion annually by 2021 (Humayun et al., 2021). With that amount of cost predicted to be caused by ransomware, this will shake organizations, individuals, and the growth of technology (Humayun et al., 2021). Also, it was discussed by the authors in “Internet of things and ransomware: Evolution, mitigation, and prevention “that a lot of ransomware attacks emerged in 2015 that targeted individuals and organizations, and criminals earned more than 4.5 million dollars through Ransomware attacks (Humayun et al., 2021). According to an AV-TEST report from 2019 to 2020, more than 114 million new malware are developed yearly, and over 78% of them have been applied to

Windows systems (C. Li et al., 2022). Due to the novel coronavirus (COVID-19) pandemic, cybercrime is up by 600,% and cybercrime by the end of 2021 and 2025 is expected to cost the world approximately $6 and $10 trillion, respectively (Nawaz et al., 2022).

The article “Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges” demonstrated how filtration email is one of the most essential and prominent approaches to detecting and preventing spam (Ahmed et al., 2022). They listed several machine learning and deep learning techniques used in email and IoT platforms by classifying them into suitable categories (Naïve Bayes, decision trees, neural networks, and random forest. They alluded that in the future, experiments and models should be trained on real-world data rather than manually created datasets because models trained on artificial datasets perform very severely on real-world data, according to several articles by (Ahmed et al., 2022). Another gap (Ahmed et al., 2022) mentioned is that blockchain ideas and concepts, in addition to machine learning, might be employed for email spam detection in the future (Ahmed et al., 2022).

The authors of the study Detecting Malware in Cyberphysical Systems Using Machine Learning: a Survey discussed the extent to which current automotive systems are vulnerable to attack and the severity of what malware can accomplish in smart automobiles. One of the examples they use is a controlled attack on a Jeep driving at 70 mph on a highway in St. Louis (U.S.A.), where the attackers remotely hijacked the car to demonstrate various electronic control units. Windshield wipers to braking and engine systems can be remotely manipulated via the vehicle's built-in cellular connection. Even though this attack was carried out in a controlled environment, it is stated that a remote attack on a vehicle is a genuine threat that might have severe ramifications for the lives of the vehicle's occupants (Montes et al., 2021).

It is estimated that millions of people worldwide will live in smart houses soon, so home security and comfort should be enhanced by utilizing this technology. The rapid increase of IoT devices utilized in smart home environments has increased security vulnerabilities, and the dangers associated with the smart house have risen. According to Sapalo Sicato et al. (2019), ensuring privacy in smart home devices is one of the biggest challenges. In the "VPNFilter Malware Analysis on Cyber Threat in Smart Home Network" article, the authors described how malware might leak confidential information because of illegal modification of software and hardware in smart home products. In the case of VPN filter malware, for example, the intruder will reprogram the router to deliver data in the form of packets to the servers and the attacker. This presents serious societal implications as well as privacy and data storage difficulties. Finally, it becomes a target for attackers who perceive it as a means of obtaining sensitive information about individuals, making them easy targets for attacks such as identity theft, phishing, or fraud (Sapalo Sicato et al., 2019).

In a 2018 study, researchers traced an estimated USD 16 million in ransom payments through two years from a potential 19,750 victims, with a further estimated total of over USD 25 million in payments between 2016 and 2017(McDonald et al., 2022). The ransomware SamSam alone had netted its developers USD 6.5 million over under two years, with its highest single ransom payment recorded at USD 64,000. Although ransomware profits seem exorbitant, the cost of damages is even more astounding (McDonald et al., 2022).

Literature Related to the Methodology

Table 2

Literature Review Summary

Sn	Title of Articles	Research Problem	Major Findings	Further Studies	Source Of Material	Goal
1	Android mobile malware detection using fuzzy AHP [2021]	Android mobile is very challenging because it is an open-source operating system that is also vulnerable to attacks. Previous studies have shown various mobile malware detection methods to overcome this problem, but still, there is room for improvement.	Risk analysis is used to raise the mobile user's knowledge of any permission request that has a high-risk level. The study employed 10,000 samples from Drebin and AndroZoo. The results demonstrate a high accuracy rate of 90.54 percent values, allowing the Android application to classify into four danger levels appropriately.	Comparison research between fuzzy AHP and other MCDM approaches can be undertaken in the future to validate effective strategies for improving mobile malware detection systems. Furthermore, as one of the approaches to raise awareness among Android users, this study should be	(Mohamad Arif et al., 2021)	In this work, the fuzzy AHP technique is used to assess risk. This technique utilizes a pair-wise comparison of criteria performed via a matrix table to analyze the criterion weight and consistency of the judgment.

		Mobile users mostly ignore long lists of permissions because these are difficult to understand. Therefore, it is necessary to evaluate Android mobile applications to distinguish benign or malware applications and ensure the probability of each permission request is understood.		considered to broaden the security vulnerabilities of Android applications that expose consumers to malware assaults. Furthermore, it is strongly advised to utilize updated real-world data and App Store applications to evaluate the performance of the generated model, and it will be critical to review the Android mobile malware detection system in the future.		

				The limitation of this study is that it only focused on permission-based features. However, other static elements, such as Java code and the intent filter, can be selected to broaden the investigation. This study's findings will help future researchers improve the Android mobile malware detection system.		
2	Detection of Exceptional Malware Variants Using Deep Boosted Feature Spaces and	Software designed with a malicious purpose to harm users or systems falls under the category of malware.	The proposed DBFS-MC improved performance for these difficult-to-discriminate malware classes using the idea of feature boosting generated through customized CNNs. The proposed classification framework DBFS-MC showed	The suggested frameworks (DBFS-MC and DFS-MC) may be extended to additional malware attacks in the future	(Asam et al., 2021)	This study suggests two new malware classification frameworks: Deep Feature Space-based Malware

	Machine Learning [2022]	Malware may harm the system without user knowledge of any level of damage; it may range from gaining system access, deleting files, ransom demands, or even sabotage. A substantial increase in credential harvesting using malware and well-established tactics has been noted in the recent past. During the COVID-19 pandemic, Microsoft reported 16 state-level actors who targeted	promising results in terms of accuracy: 98.61%, F-score: 0.96, precision: 0.96, and recall: 0.96 on stringent test data, using 40% unseen data.	utilizing the standard benchmark dataset, such as Android and IoT malware. Furthermore, this research might be improved by creating an anti-malware program for Microsoft Windows OS that can examine FTP traffic in real-time scenarios for malware detection.		classification (DFS-MC) and Deep Boosted Feature Space-based Malware classification (DFS-MC) (DBFS-MC). Deep features are generated from customized CNN architectures and fed to a support vector machine (SVM) algorithm for malware classification in the proposed DFS-MC framework. In contrast, the discrimination power is enhanced in the DBFS-MC framework by

		commercial and academic institutions for stealing vaccine-related research knowledge. These threat actors have rapidly become more sophisticated over the past years. They are skilled, persistent, and can launch attacks that are harder to spot				combining deep feature spaces of two customized CNN architectures to achieve boosted feature spaces.
3	Federated learning for malware detection in IoT devices [2022]	Billions of IoT devices lacking proper security mechanisms have been manufactured and deployed for the last few years, and more	A framework for identifying malware on IoT devices using federated learning is provided. The suggested framework was tested using N-BaIoT, a dataset that models the network behavior of multiple IoT devices infected with malware. Supervised and unsupervised	In the future, we intend to assess the impact of adversarial attacks in the unsupervised scenario to ensure that they affect the	(Rey et al., 2022)	This work investigates the opportunities federated learning provides for detecting IoT malware and the security

		will come with the development of Beyond 5G technologies. Their vulnerability to malware has motivated the need for efficient techniques to detect infected IoT devices inside networks. With data privacy and integrity becoming a major concern in recent years, increasing with the arrival of 5G and Beyond networks, new technologies such as federated learning and blockchain	federated models (multi-layer perceptron and autoencoder) capable of identifying malware influencing both observable and unseen N-BaIoT IoT devices have been trained and tested. Their performance has also been compared in two recognized methods. The first allows each participant to train a model locally using its data. In contrast, the second requires participants to share their data with a central entity in charge of training a global model. This comparison has revealed that using more diverse and significant data in federated and centralized techniques significantly influences model performance. Furthermore, while maintaining participant privacy, federated models produce similar outcomes as centralized ones. For example, an adversarial setup with numerous malicious	findings in the same manner they do in the supervised counterpart. Furthermore, assessing the model's resilience against evasion attacks, using fabricated adversarial samples to escape detection during the assessment, might be intriguing for future research. Additionally, this effort intends to research current defenses against adversarial assaults, such as Krum, Bulyan, and AUROR.		challenges of this new learning paradigm.

		emerged. In addition, they allow training machine learning models with decentralized data while preserving its privacy by design.	participants poisoning the federated model was examined as an extra contribution to assess the federated method's resilience. Even with a single adversary, the baseline model aggregation averaging step used in most federated learning algorithms looks particularly susceptible to various assaults. Under the same assault scenarios, the performance of various model aggregation algorithms serving as countermeasures is therefore tested. These functions significantly improve against malicious participants, but more effort is needed to make federated approaches robust.	Scalability in real B5G situations is another issue that could not be studied with any of the current datasets, generating a much bigger and more diversified one.		
4	Internet of things and ransomware: Evolution, mitigation, and prevention	Internet of things architecture integrates real-world objects and places with the internet. This	The literature indicates a greater trajectory toward ransomware assaults, which is predicted to be five times higher by 2020, with more than $6 trillion in ransom against ransomware attacks. Furthermore, this analysis suggests that a ransomware	Ransomware is a crucial concern of emerging technological development. However, this development requires a safe	(Humayun et al., 2021)	This paper provides a comprehensive survey on the evolution, prevention, and mitigation of Ransomware in

	[2021]	technological boom is bringing ease to our lifestyle and making formerly impossible things possible. The Internet of things is vital in bridging this gap easily and rapidly. For example, IoT is changing our lifestyle and the way of working with technologies by bringing them together on one page in several application areas of daily life. However, IoT has to face several challenges in the form of	attack occurs every 11 seconds worldwide. Furthermore, this research focuses on existing IoT-linked ransomware attacks, mitigation procedures, and recommended ransomware preventive methods. After a ransomware attack, prevention becomes easier than finding a cure. User behavior and training protect businesses, organizations, and individuals against infection. In addition, the FBI recommends limited privileged, timely backup, disabled macro and java scripts, software restriction policies, and employee training regarding Ransomware awareness.	and secure path to further its boom. This increase in ransomware attacks is an open research issue and a challenge for further growth. In the future, the authors want to look at more efficient ransomware mitigation approaches.		the IoT context.

		cyber scams; one of the significant challenges IoT has to face is the likelihood of Ransomware attacks. Ransomware is a malicious kind of software that restricts access to vital information in some way and demands payment for getting access to this information. The ransomware attack is becoming widespread daily, bringing disastrous consequences, including loss of sensitive data,				

		loss of productivity, data destruction, reputation loss, and business downtime. Which further leads to millions of dollars in daily losses due to downtime.				
5	Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges [2022]	Email spam, also called junk emails or unwanted emails, is a type of email that can be used to harm any user by wasting their time, computing resources, and stealing valuable information. The ratio of spam emails is increasing	According to the findings of this study, the majority of the suggested email and IoT spam detection systems are based on supervised machine learning approaches. Therefore, a labeled dataset is required for supervised model training, which is time-consuming. In spam identification, the supervised learning algorithms SVM and Nave Bayes outperform other models.	In the future, experiments and models should be trained on real-life data rather than manually created datasets because, in the various article, the models trained on artificial datasets perform very poorly on real-life data. Currently,	(Ahmed et al., 2022)	This study categorizes the machine learning algorithms used for spam filtering strategies in email and IoT systems. These methodologies are also thoroughly compared in terms of accuracy, precision, recall,

		rapidly day by day. Spam detection and filtration are significant and enormous problems for email and IoT service providers nowadays. Among all the techniques developed for detecting and preventing spam, filtering email is one of the most essential and prominent approaches. Several machine learning and deep learning techniques have been used for this purpose, i.e., Na¨ive		supervised, unsupervised, and reinforcement learning algorithms are used for spam detection, but we can get higher accuracy and efficiency by using hybrid algorithms in the future. Feature extraction can be improved in the future by using deep learning for feature extraction Along with machine learning, blockchain models and concepts can		etc.

		Bayes, decision trees, neural networks, and random forests.		also be used for email spam detection in the future Experts in linguistics and psycholinguistics can collaborate in the future for the manual annotation of datasets, which will result in the development of effective and standard spam datasets with high dimensionality In the future, spam filters can be designed with faster processing and classification accuracy using Graphics		

				Processing Units (GPUs) and Field Programmable Gate Arrays (FPGAs), which offer low energy consumption, flexibility, and real-time processing capabilities. Future research should concentrate on the availability of standard labeled datasets for researchers to train classifiers and the addition of more attributes to the dataset to improve the accuracy and reliability of spam detection		

				models, such as the spammer's IP address and the location		
6	Ransomware: Analysing the Impact on Windows Active Directory Domain Services [2022]	Ransomware has become an increasingly popular type of malware across the past decade and continues to rise in popularity due to its high profitability. As a result, organizations and enterprises have become prime targets for ransomware as they are more likely to succumb to ransom demands as part of operating expenses to	Three ransomware variants (WannaCry, TeslaCrypt, and Jigsaw) were subjected to dynamic analysis to determine how crypto-ransomware impacts Windows Server-specific services and processes. According to the findings, none of the three variations stopped the processes and left all domain services undisturbed. However, while the services remained functioning, they were notably dysfunctional because ransomware encrypted the associated data. The hypothesis suggested that ransomware would not disable the tested services but would disrupt their functioning by encrypting relevant data. The authors created a virtual environment with a domain	In a future study, testing third-party applications from computer-oriented software to the software responsible for physical entities could produce vastly different results, as third-party software does not typically use system-critical file paths.	(McDonald et al., 2022)	This paper describes the practical study conducted while WannaCry, TeslaCrypt, and Jigsaw were obtained and tested against various domain services.

		counter the cost incurred from downtime. Despite the prevalence of ransomware as a threat to organizations, there is little information outlining how ransomware affects Windows Server environments, mainly its proprietary domain services such as Active Directory.	controller running Windows Server 2016 and a client PC running Windows 10. Several Windows Server services were then built to enable prolonged testing to produce qualitative and quantitative data. All tested services remained working despite the three ransomware variants. Services that used files that were not part of the service's normal settings and file paths experienced delays in functioning, whereas system-critical pathways remained unaffected. This validated the previously stated hypothesis.			
7	VPNFilter Malware Analysis on Cyber Threat in Smart Home Network	Recently, new malware was implemented in many different routers on the network. Known as VPNfilter	The authors developed a taxonomy focusing on cyber threat attacks that may influence a smart home system. They identified several critical issues about VPNFilter malware, a large-scale Internet of Things (IoT)-based botnet malware	Privacy in smart home devices is one of the biggest challenges. In the case of unauthorized manipulation of	(Sapalo Sicato et al., 2019)	This article aims to study the many components of cyber-physical threats on the smart home from a security

	[2019]	malware, it is considered a sophisticated malware variant mainly targets networking devices from a wide range of manufacturers named Vpnfilter. This malware can collect confidential information that passes through an infected router, allowing attackers to gain control of Wi-Fi routers directly to obtain unexpected sensitive personal data. In addition, the malware infects routers to manipulate sites	infection. The first taxonomy offers a series of four levels in the smart home system, each of which may be attacked and is required for the safety of the entire network, not just the specific technology. Using this taxonomy, the authors methodically examined the privacy problems and security risks and all layers of the smart home system. The second taxonomy refers to attacks based on a smart home central hub, and the last taxonomy describes attacks based on the physical security of the smart house.	software and hardware in smart home appliances, confidential information may leak. In the case of VPN filter malware, for example, the intruder will reprogram the router to provide data in the form of packets not only to the servers but also to the attacker. This presents vast societal implications as well as privacy and data storage difficulties. In addition, it attracts attackers who perceive it as a method to obtain sensitive		standpoint, describe the types of attacks, including sophisticated cyber-attacks and cyber-physical system attacks, and assess the impact on a smart home system in everyday life.

		visited by users on the same network because the threat acts as the source of internet signal; it need not directly affect the victim's smartphone and computer.		information about individuals, making them easy targets for assaults such as identity theft, phishing, or fraud. Vulnerability: Various vulnerabilities as a weakness in the system allow an attacker to access unauthorized data and execute the command VPN filter. This was described as DOS attacks. The smart home system, made up of two major components, software, and hardware,		

				frequently has design problems. Malware-based software vulnerabilities can be detected in the device's application software and operating system. For example, it is difficult to discover and repair hardware vulnerabilities in routers. However, several technical flaws have been discovered to result from human flaws. Software exploitation: Based on the smart home system and the		

				devices therein, the authors advised that we consider the possibility of infection by malicious software such as VPNfilter malware, DDoS, DOS, and others. Smart home gadgets operate independently, prompting operational enemies to look for software flaws to attack and access the system's sensitive information. It is now the focus of several attacks, the resulting traffic in the devices serving to operate		

				VPNfilter, and DOS attacks. DDoS attacks, for example, were launched using IoT devices against DNS servers to disrupt internet access. The cost of a smart home is one of the biggest challenges to consider in a smart home environment under a cyber-security attack. The attack raises users' costs by affecting their well-being and compromising their gadgets—the		

				psychological impact on the user's health and the expense of replacing contaminated gadgets both rise. The manufacturers suffer a cost impact in terms of providing increased security to assure their customers that their products are safe and secure. In addition, they are required to invest in developing devices that offer robust security measures.		
8	A Survey on Botnets, Issues, Threats,	Botnets have become increasingly common and	From the evidence gathered, botnets' primary causes of penetrating network systems are phishing attacks and brute-	Open ports in a system must be closed.	(Owen et al., 2022)	The study examines how threat actors use botnet code to

	Methods, Detection, and Prevention [2022]	progressively dangerous to business and domestic networks. Due to the Covid-19 pandemic, many people have been performing corporate activities from their homes. This leads to speculation that most computer users and employees working remotely do not have proper defenses against botnets, resulting in botnet infection propagating to other devices connected to the target network. Consequently,	forcing sessions within packet transition. This means that to reduce the risk for botnets, it is vital for the network administrators to equip firewalls against botmaster's variations on the malware code and update firewalls constantly. It is also essential for IDS and IPS to be implemented if the botnet can penetrate the firewalls. Password strength will also need to be considered for SSH sessions. In addition, using long passwords with encryption can ensure that sessions are not cracked and allow the bot master to conduct an insertion throughout the host communication. ACLs are valid as they can enable botnets to have limited propagation on hosts and isolate the infected hosts to ensure that machines with more sensitive information and data would be more secure from any other	Providing staff with training on how to respond to different social engineering-based threats. Ensuring proper measurements are taken if emails have been sent to staff members for training Updating firewalls implemented into networks and configuring firewalls allow for analyzing suspicious packets—firewall updates to ensure that traffic that contains botnet code is denied		infect target devices. Machine learning algorithms are investigated to identify how they may be utilized to support AI-based detection and what benefits and limitations they have to compare the most suited algorithm that organizations can adopt. Finally, current botnet prevention and countermeasures are explored to identify how botnets may be stopped from entering corporate and

		not only did botnet infection occur within the target user's machine but also neighboring devices.	related attacks. Phishing attacks can be prevented via staff training on appropriately responding to emails and links from unknown sources. For further confirmation on emails being sent to them, staff can ask the sender personally if they have sent an email. Other measures that can be taken are our email and website filtering. AI has also been pivotal in detecting infections using fuzzy logic. For example, it could consider missing binary values within data packets during traffic during flow time to detect the presence of malicious code by using its decision-making capabilities. IDS and IPS can ensure whether the bot master can penetrate the network. For example, the IDS can alert administrators to the bot master accessing the network and allow the IPS to ensure that the bot master is removed from	access.		domestic networks and to guarantee that future assaults can be avoided.

			the network. Using aspects such as fuzzy logic and ML-based IDS/IPS, AI can contribute to the network and IoT security by protecting them from botnets or malware threats.			
9	A Three-Level Ransomware Detection and Prevention Mechanism [2018]	Ransomware encrypts victims' files or locks users out of the system. Victims will have to pay the attacker a ransom to decrypt and regain access to the user files. Petya targets individuals and companies through email attachments and download links. NotPetya has worm-like capabilities and	Three Level Security (3LS) is a solution to ransomware that utilizes virtual machines and browser extensions to scan any files the user wishes to download from the Internet. For example, a browser extension would send the downloaded files over a cloud server relay to a virtual machine. Any changes to the virtual machine after downloading the file would be observed, and if there were a malfunction in the virtual machine, the file would not be retrieved to the user's system.	In the future, the authors hope to increase the number of virtual machines one computer can handle with technological advancement. In their research, they firmly believe that virtual machines can be a valuable protection mechanism against malware, which is a step in the right direction to combating	(Ren et al., 2018)	In the author's solution, we proposed a method to deal with ransomware or malware by using virtual machines. The aim is to isolate potential malicious files in the virtual machine and quarantine them instead of letting malware the host system.

		exploits EternalBlue and EternalRomance vulnerabilities. Protection methods include vaccination, applying patches, et cetera. Challenges to combat ransomware include social engineering, outdated infrastructures, technological advancements, backup issues, and standards conflicts.		malware.		
10	Malware Detection and Prevention using Artificial Intelligence	With rapid technological advancement, security has become a significant issue	Study shows that adopting futuristic approaches for developing malware detection applications shall provide significant advantages. This synthesis's comprehension shall		(Hossain Faruk et al., 2021)	In this study, the authors emphasize Artificial Intelligence (AI) based

	Techniques [2021]	due to the increase in malware activity that seriously threatens the security and safety of computer systems and stakeholders. Protecting the data from fraudulent efforts is one of the most pressing concerns to maintaining stakeholders, particularly end users' security. For example, malware is malicious programming code, scripts, active content, or intrusive	help researchers further research malware detection and prevention using AI. The findings indicate that AI can be utilized as a promising domain for the development of anti-malware systems for detecting and preventing malware attacks or security risks of software applications towards a technological wonderland			techniques for detecting and preventing malware activity. In addition, they present a detailed review of current malware detection technologies, their shortcomings, and ways to improve efficiency.

		software designed to destroy intended computer systems and programs or mobile and web applications. According to a study, naive users cannot distinguish between malicious and benign applications. Thus, computer systems and mobile applications should be designed to detect malicious activities to protect stakeholders. In addition, several algorithms can detect malware				

		by utilizing novel concepts, including Artificial Intelligence, Machine Learning, and Deep Learning.				
11	A Survey on Mobile Malware Detection Techniques [2020]	Modern mobile devices are equipped with various tools and services and handle increasing amounts of sensitive information. In the same trend, the number of vulnerabilities exploiting mobile devices is also augmented daily. Undoubtedly, popular mobile platforms, such	The author's work provides a state-of-the-art survey on the timely topic of mobile malware detection techniques. They categorized and briefly analyzed the various detection schemes proposed in the literature during the last eight years, i.e., from 2011 to 2018, based on their detection method. They also highlight the benefits and limitations per category of techniques and the examined scheme, where applicable, to offer a comprehensive overview of this challenging and fast-evolving topic.	Most of the techniques surveyed in Sect. Three still lack in detecting zero-day malware, but this is somewhat expected. Furthermore, with the current sophistication of malware, it is difficult to detect it through traditional rule matching using existing technologies. This may be the main reason	(Kouliaridis et al., 2020)	This survey aims to provide state-of-the-art information on current mobile malware trends. Furthermore, it offers a comprehensive overview of the different approaches to mobile malware detection to understand their detection method, discuss their evaluation results, and possibly categorize each

		as Android and iOS, represent a tempting target for malware writers. While researchers strive to find alternative detection approaches to fight against mobile malware, recent reports exhibit an alarming increase in mobile malware exploiting victims to create revenues, climbing towards a billion-dollar industry. Unfortunately, current mobile malware analysis and detection		behind a large number of malicious apps still on the loose in official app stores. Therefore, future research efforts should clarify how to efficiently join detection techniques into hybrid solutions to increase the subset of malware that can be detected, as proposed in previous work, and offer actual detection improvement.		contribution under a novel classification scheme.

		approaches cannot always keep up with future malware sophistication.				
12	Effective classification of android malware families through dynamic features and neural networks [2021]	Due to their open nature and popularity, Android-based devices have attracted several end-users around the World and are one of the main targets for attackers. Because of the reasons given above, it is necessary to build tools that can reliably detect zero-day malware on these devices. At the moment, many of the	In this paper, the realization of a new Android malware dataset called Unisa Malware Dataset (UMD) has been presented. UMD is available at http://antlab.di.unisa.it/malware/. The proposed dataset has been realized by analyzing 30,113 malware applications through CuckooDroid Sandbox. The UMD contains 20,426 apps organized into 66 families for AMD and 4849 applications organized into 143 families for Drebin. Besides, for each analyzed application, static and dynamic features are available, such as hash fingerprints, permissions, dynamic API calls, and so on. Then, an experiment with Artificial Neural Networks (ANNs) was presented to show the extracted API calls' potential	The authors would like to propose two possible future works. First, to improve the number of malware applications and the number of the considered families, they will update the proposed dataset by considering other malware datasets, such as Android Adware and General Malware Dataset (AAGM Dataset) (Habibi Lashkari et al.,	(D'Angelo et al., 2021)	This paper's main aim is to propose a new Unisa Malware Dataset (UMD) dataset based on extracting static and dynamic features characterizing the malware program activity. They also showed some experiments based on standard ML end DL techniques to demonstrate how it is possible to build

		frameworks that have been proposed to detect malware applications leverage Machine Learning (ML) techniques. However, an essential requirement to build these frameworks is using very large and sophisticated datasets for model construction and training purposes. Their success, indeed, strongly depends on the choice of the right features used for building a	by considering five malware families (Airpush, Dowgin, FakeInst, DroidKungFu, and Opfake). However, UMD is an unbalanced dataset consisting of many malware families with few applications. At the same time, many malware families should be included in our dataset. Consequently, only a limited subset of families can be considered to propose new AI-based solutions. Furthermore, 500 samples have been selected for each family, and dynamic API calls have been extracted from them as an API image. Finally, a Convolutional Neural Network (CNN) and a Recurrent Neural Network (RNN) have been used and validated using statistical metrics. The results show that these neural networks are an effective solution to recognize malware families when the right features are used to describe the behavioral properties of individual malware flavors.	2017, august), AndroZoo (Allix et al., 2016), Genome (Zhou & Jiang, 2012) and so on. They did not include these datasets yet, because the analysis' process is costly and time-consuming. Second, they will propose new AI models based on the extracted static and dynamic features to improve the results. For example, a Recurrent Neural Network (RNN) based on Long Short-Term Memory (LSTM) layers could be suitable		efficient malware classification solutions using the proposed dataset to train several kinds of AI-based models properly.

		classification model providing adequate generalization capability. Furthermore, creating a training dataset that represents the malware properties and behavior is one of the most critical challenges in malware analysis.		for using temporal features, such as timestamps. Moreover, using CNN autoencoders could be investigated to obtain important features by API-image based on the extracted static and dynamic information. Additionally, we will explore new DL approaches that can classify dynamic features as a film. Finally, several combinations among LSTM layers, CNNs, and Stacked Autoencoders (SAEs) could be		

				investigated to consider a single API-Image as a stream of sub-API images by assuming many sets of images obtained at fixed multiple temporal windows.		
13	A malware Detection Approach Using Autoencoder in Deep Learning [2022]	Today, in the field of malware detection, the expanding limitations of traditional detection methods and the increasing accuracy of detection methods designed based on artificial intelligence algorithms are driving research	A novel malware detection model combines a grey-scale image representation of malware with an autoencoder network in a deep learning model, analyses the feasibility of the grey-scale image approach of malware based on the reconstruction error of the autoencoder, and uses the dimensionality reduction features of the autoencoder to achieve the classification of malware from benign software. The proposed detection model achieved an accuracy of 96% and a stable F-score of about 96% by using the Android-side	In future work, the authors will continue to explore more effective methods for representing malware feature images and focus our research on the pre-processing data stage to explore newer malware detection methods.	(Xing et al., 2022)	The authors propose a novel malware detection model in this paper. This model combines a grey-scale image representation of malware with an autoencoder network in a deep learning model, analyses the feasibility of the grey-scale

		findings in favor of the latter.	dataset we collected, which outperformed some traditional machine learning detection algorithms. Experimental results show the feasibility of their proposed approach of converting the bytecode of all methods in software into a greyscale image to represent the features in a software sample. Their method is more accurate than malware detection methods designed based on traditional machine learning algorithms. The author's method requires less training and detection time than other malware detection systems designed based on deep learning models.			image approach of malware based on the reconstruction error of the autoencoder, and uses the dimensionality reduction features of the autoencoder to achieve the classification of malware from benign software.
14	Dynamic Analysis for IoT Malware Detection With Convolution Neural	In the IoT environment, devices are connected and exchange information. Because of this	First, a cloud-based nested virtual environment was designed and implemented to analyze and detect IoT malware in a safe environment. Then, the DAIMD model was created by performing training, validation,	A study on implementing a model that can detect IoT malware using the hybrid analysis	(Jeon et al., 2020)	This paper proposes a dynamic analysis for IoT malware detection (DAIMD) to

	Network Model [2020]	characteristic, the number of attacks such as distributed denial of service (DDoS), cryptocurrency malicious mining, and botnet activities is expanding at a fast pace. In addition, to cope with the rapidly increasing demand for IoT devices, some manufacturers are mass-producing IoT devices vulnerable to security breaches and providing them to users. If vulnerable IoT devices are	and testing according to the following phases: debugging, feature extraction, feature pre-processing, feature selection, and classification in the cloud environment Since the feature data of the behaviors extracted through the detection process were numerous, they were converted to images to prevent a complex computation problem for training and classification of the feature data in the classification phase, reducing the number of dimensions of the data. In addition, the features of IoT malware and benign files were comprehensively represented through the DAIMD visualization technique. The infection of IoT devices or the propagation of IoT malware to other IoT devices connected through the Internet can be prevented using DAIMD. Furthermore, because the	technique, which analyzes malware by utilizing static and dynamic techniques, will be conducted in the future.		reduce damage to IoT devices by detecting well-known, new, and variant IoT malware evolved intelligently.

		distributed in the market, they will be the main target for malware makers.	DAIMD selects and classifies behavior features using the CNN model without human subjective intervention, new and variant IoT malware with various intelligent attack techniques can be accurately detected The DAIMD proposed in this paper analyzed behavior features by executing IoT malware using a dynamic analysis technique. Because some IoT malware can easily recognize that they are executed in a limited environment such as a VM, they may avoid malware analysis and detection systems that use the dynamic analysis technique			
15	Improving the Robustness of AI-Based Malware Detection Using Adversarial Machine Learning	Cyber security protects computers and networks from ill-intended digital threats and attacks. However, it is getting more	The authors have implemented a malware classification system with machine learning, deep learning, and a pre-trained model, achieving an accuracy of 93% for the random forest, 92.3% for CNN, 93.7% for the efficient net, and 92% for VGG-16. Then, the authors performed	The future scope of research would be using other forms of attacks available and subsequently training the model against those attacks,	(Patil et al., 2021)	This paper proposes a framework for generating adversarial malware images and retraining the classification

	[2021]	difficult in the information age due to the explosion of data and technology. There is a drastic rise in the new types of attacks where the conventional signature-based systems cannot keep up with these attacks. Machine learning seems to be a solution to solve many problems, including problems in cyber security. It is a handy tool in the evolution of malware detection systems.	an FGSM attack on the EfficientNet model with images with 0.01, 0.1, and 0.15 epsilon values. The model successfully misclassified the results. When trained against these adversarial samples, this model will not misclassify the results and make the system robust against the FGSM adversarial attack. The adversarial training will assist the system in becoming robust while executing the detection, and the machine learning model will aid in identifying harmful files. The proposed system was able to demonstrate that the model is vulnerable to adversaries via adversarial attacks	making it even more robust.		models to improve malware detection robustness. Different classification models were implemented for malware detection, and attacks were established using adversarial images to analyze the model's behavior.

		However, the security of AI-based malware detection models is fragile. With advancements in machine learning, attackers have found a way to work around such detection systems using an adversarial attack technique. Such attacks are targeted at the data level, at classifier models, and during the testing phase. These attacks tend to cause the classifier to misclassify the given input,				

		which can be very harmful in real-time AI-based malware detection.				
16	Detection of Ransomware Using Process Behavior Analysis [2020]	Ransomware attacks are one of the biggest and most attractive threats in cyber security today. Anti-virus software's often inefficient against zero-day malware and ransomware attacks, and important network infections could result in a large amount of data loss. Such attacks are also becoming more dynamic and able to change	The authors were able to study a lot of different ransomware and extract values like the DLLs used and the system usage. As a result, they were able to increase my dataset of malware and ransomware. Even if the machine learning technique needs more training, they were able to implement a solution against ransomware attacks which allowed them to detect the process and determine if it is ransomware or not with the API calls of each function used by DLLs, with the extensions, the disk usage, and the number of threads. The author's system can detect zero-day ransomware attacks and warn users about a potential threat. The benefit of their solution is that it does not need a signature database but a	The author's next step is to try and get detection done within the first 5 seconds of malicious activity, then pass the information into an agent that will communicate the information securely to the ecosystem to form an early warning system for self-defense and create a more reactive preventative solution rather than a reactive defense – hence	(Arabo et al., 2020)	This study investigates the relationship between a process's behavior and its nature to determine whether it is ransomware. The paper aims to see if using this method will help them evade malicious software and use it as a self-defense mechanism using machine learning that emulates the human immune

		their signatures – hence creating an arms race situation	dataset of ransomware and none-ransomware data. As a result, the more the dataset is enhanced, the more the system is successful in its discrimination.	provide a zero-trust security solution		system
17	An Empirical Analysis of Image-Based Learning Techniques for Malware Classification [2021]	Traditionally, malware detection and classification have relied on pattern matching against signatures extracted from specific malware samples. While simple and efficient, signature scanning is easily defeated by several well-known evasive strategies. This fact has given rise to statistical and machine	For the author's deep learning techniques, they focused on multilayer perceptrons (MLP), convolutional neural networks (CNN), and recurrent neural networks (RNN), including long short-term memory (LSTM) and gated recurrent units (GRU). They also experimented with the image-based transfer learning techniques ResNet152 and VGG-19. Among these techniques, the image-based transfer learning models performed the best, with the best classification accuracy exceeding 92%.	Additional transfer learning experiments would be worthwhile for future work, as many more parameters could be tested. More extensive and diverse datasets could be considered. In addition, it would be interesting to consider both image-based and opcode features as part of a combined classification technique.	(Prajapati & Stamp, 2021)	This paper considers malware classification using deep learning techniques and image-based features. In addition, we employ a wide variety of deep learning techniques, including multilayer perceptrons (MLP), convolutional neural networks (CNN), long short-term memory (LSTM), and

		learning-based techniques, which are more robust to code modification. In response, malware writers have developed advanced forms of malware that alter their code's statistical and structural properties, which can cause statistical models to fail.				gated recurrent units (GRU).
18	Robust Android Malware Detection System against Adversarial Attacks using Q-Learning [2021]	Since the inception of Android OS, smartphone sales have been growing exponentially, and today it enjoys a monopoly in the smartphone	The authors also proposed a novel single policy attack for the white-box setting where an adversary has complete knowledge about the detection system. They design a reinforcement agent which performs an adversarial attack using a policy obtained from a single Q-table. The attack achieves an average fooling rate	In the future, the authors will explore fooling Android malware detection models based on other features. We also plan to design an adversarial attack based on	(Rathore et al., 2021)	In this paper, the authors developed eight Android malware detection models based on machine learning and deep neural network and

		marketplace. The widespread adoption of Android smartphones has drawn the attention of malware designers, which threatens the Android ecosystem. The current state-of-the-art Android malware detection systems are based on machine learning and deep learning models. Despite having superior performance, these models are susceptible to adversarial attacks.	of 44.28% across all eight detection models with a maximum of five modifications. The attack also achieves the highest fooling rate against the DT model (54.92%), whereas the lowest fooling rate is obtained for GB (37.77%) with a similar setting. Overall, the experimental result signifies that a single policy attack can successfully evade malware detection models and accomplish a high fooling rate even with limited modifications They also develop a state-of-the-art adversarial attack, namely a multi-policy attack for the grey-box setting where the attacker does not know the model architecture and classification algorithm. The multi-policy attack achieves the highest fooling rate for the DT model (86.09%), followed by the ET model (75.23%) with a maximum of five modifications. The average fooling rate is	reinforcement learning techniques like deep q-learning, actor-critic algorithm, proximal policy optimization, etc.		investigated their robustness against adversarial attacks.

			increased to 53.20%, which is higher than the single policy attack even with limited information. Finally, they propose a defense against adversarial attacks based on a single policy and multi-policy attack strategies. With adversarial retraining, the average fooling rate against the single policy attack is reduced by threefold to 15.22% and twofold for the multi-policy attack to 29.44%, i.e., it can now effectively detect variants (metamorphic) of malware. The experimental analysis shows our proposed Android malware detection system using reinforcement learning is more robust against adversarial attacks.			
19	Toward an Ensemble Behavioral-based Early Evasive Malware	Recently malware threats are evolved to be the most cyber security threats. Because	The developed framework involves three main phases, evasion behaviors collection, correlation-based features extraction, selection, and constructing the model phase. A	As the future direction of the author's work, the proposed model framework is	(Aboaoja et al., 2021)	This paper proposes a framework for building an effective early malware

	Detection Framework [2021]	of obfuscation and evasion techniques, malware has become more sophisticated in terms of multiple variants representing the same malware function and rapidly evades existing detection approaches. The current solutions extracted the entire data without considering the unrepresentative data that belongs to evasive malware when they recognize that they are	predetermined evasion techniques list was employed to identify the representative data of evasive behaviors during the data collection stage. The required features were extracted and selected using n-gram, TF-IDF, and PCC techniques based on the representative data. To enhance the detection accuracy, this paper developed an ensemble behavioral-based early evasive malware detection framework that can effectively recognize the sophisticated malware behaviors using an ensemble learning approach and getting the final decision according to the outcome of the majority voting strategy.	designed and developed to be concerned with the evasion techniques achieved by malware using API and system calls, so the proposed model is limited to coping with only those evasion techniques. Therefore, the proposed model needs improvement to cover the evasive malware that implements their evasion techniques directly without passing API calls.		detection model to protect systems and data from evasive malware attacks.

		executed in controlled environments. In addition, obfuscation techniques such as dead code insertion and reordering instructions aim to produce irrelevant data and make the previous approaches based on names, frequencies, and sequences of the extracted data suffer a low detection rate.				
20	Detection, Traceability, and Propagation of Mobile Malware Threats	With the popularity of smartphones and the rapid development of mobile applications	The authors studied the traceability, propagation, and detection of the threats, by performing research on all aspects of the end-to-end environment. By controlling the spread of the malware network,	The spread of malicious mobile programs in a new generation of the mobile Internet environment	(Chen et al., 2021)	The goal of the authors is to study the traceability, propagation, and detection of the application of

	[2021]	worldwide, mobile programs have become the main entrance to the Internet, becoming an essential part of massive data storage and end-to-end transmission. Android-based mobile terminals have quickly occupied the mainstream market because of their openness, completeness, creativity, and hardware compatibility. According to the Operating System Market Share	sample collection, research, and plugging sentences, the malware prevention system and key scientific issues are controlled to protect the safety of the mobile Internet. The network side technology monitors the download source, download channel, and terminal running environment of the mobile application, while network traffic analysis identifies the malware transmitted in the network. With machine learning based on mobile malware detection algorithms that integrate the dynamic and static research of the identification algorithm, application software samples are collected to study sentences. The authors perform detection on a large network in China Unicom mobile environment regarding 178,155 real malicious program data by using the data flow probe of Android mobile	dominated by high-speed 5G networks poses a significant risk. The deployment of future 6G, content center networks, SDN networks, and new networks of popular core applications may all become the main targets of new mobile malicious programs. Unknown types of malicious mobile programs spread and harm terminals in a distributed manner without the operator's network prevention and control		smartphones, Android operating systems, and mobile application

		Worldwide, the Android operating system has occupied first place in the mobile operating system share in recent years.	program data files to obtain detailed Guiyang (China ‘‘Data Center Capital’’) mobile malware-infected program data. Static, dynamic analysis of the malicious mobile program is carried out, and the social network social diagram is constructed to model the propagation of the malicious mobile program. The authors extended the approach of deriving common malware behavior through graph clustering to the Android mobile malicious program detection field in the Linux kernel. On this basis, Android behavior analysis is performed through our virtual machine execution engine to evaluate the maliciousness of the program through the heuristic analysis algorithm. The authors extended the family characteristics to the concept of DNA race genes by studying SMS/MMS, Bluetooth, 5G base station networks, metropolitan	mechanism. Many types of malicious mobile programs use the proximity of devices to replicate themselves in a distributed manner, making them difficult to detect. Due to the lack of a suitable network provider and highly dynamic prevention and control topology (hindering possible defense lines). In addition, mobile terminals, including their limited processing power, storage space, and battery power,		

			area networks, social networks, homogeneous communities, telecommunication networks, and application market ecosystem propagation scenarios and discovered the law of propagation.	all these attributes constitute an obstacle to the timely distribution of mobile malicious program signature files between mobile devices		
21	Study on Systematic Ransomware Detection Techniques [2022]	Cyberattacks have recently progressed in the Internet of Things and artificial intelligence technologies using the advanced persistent threat (APT) method. The damage caused by ransomware is rapidly spreading among APT	This study tested whether each EDR tool can detect file changes caused by ransomware. Moreover, GRR, osquery and OSSEC were chosen as representative EDR tools. In addition, detection results were displayed as notifications or logs when all files in a specific directory were encrypted, indicating that all three EDR tools in the Linux environment could detect ransomware. Through this open source-based threat detection, it is possible to determine the point at which the ransomware was executed and to analyze the attack method.	Although this study has not been tested under various environmental change conditions, in future work, the efficiency of the EDR tool will be compared by measuring the accuracy and speed with which EDR detects ransomware when EDR is	(S.-J. Lee et al., 2022)	This study identifies in real time whether clients are infected with open-source cryptographic ransomware, RAASNet, through Google Rapid Response (GRR), Facebook's osquery, and Open Source has SECurity (OSSEC) among

		attacks, and the range of damages to individuals, corporations, public institutions, and even governments is increasing. However, the seriousness of the problem has increased because ransomware has been evolving into an intelligent ransomware attack that spreads over the network to infect multiple users simultaneously.		deployed to large clients.		systematic ransomware detection techniques
22	Binary Black-Box Attacks	Recent machine learning- and	To emulate adversarial malware attacks, the authors propose	A promising future direction	(Ebrahimi et al., 2021)	The main contributions of

	Against Static Malware Detectors with Reinforcement Learning in Discrete Action Spaces [2021]	deep learning-based static malware detectors have shown breakthrough performance in identifying unseen malware variants. As a result, they are increasingly being adopted to lower the cost of dynamic malware analysis and manual signature identification. Despite their success, studies have shown they can be vulnerable to adversarial malware attacks. For example, an	AMG-VAC, a novel RL method designed specifically to support discrete modifications of malware executables in AMG tasks. They show that AMGVAC outperforms extant RL-based and non-RL-based AMG methods through rigorous evaluation. In addition, AMG-VAC contributes to the deep learning research community by offering a novel approach to extending the state-of-the-art RL framework to AMG. Furthermore, AMGVAC is an effective and explainable AMG technique contributing to the malware analysis research community.	could be a rigorous procedure for using the adversarial malware variants generated by AMG- VAC to enhance the robustness of DL-based malware detectors against adversarial attacks.		this paper are twofold. First, AMGVAC offers an automated vulnerability discovery method for advanced ML-based and DL-based static malware detectors without requiring prior knowledge about their architecture or parameters. Furthermore, AMG-VAC extends VAC to operate in non-continuous action spaces where discrete sequential modifications on a malware

		adversary modifies a known malware executable to fool the malware detector into recognizing it as a benign file.				executable can lead to evasive malware variants.
23	Deep Learning for Network Traffic Monitoring and Analysis (NTMA): A Survey [2021]	The growth of the communication systems and networks in terms of the number of users and the amount of generated traffic poses different daily challenges to NTMA, including storing and analyzing traffic data, using traffic data for business goals	Deep learning has efficiently facilitated analytics and knowledge discovery in large data systems to recognize hidden and complex patterns. Motivated by these successes, researchers in the field of networking apply deep learning models for Network Traffic Monitoring and Analysis (NTMA) applications, e.g., traffic classification and prediction.	Lack of labeled data: Difficulties in using DL for structured data: Lack of successful or full exploitation of DL in some NTMA applications: Resource-constrained networks: Retraining challenge: Theory of network:	(Abbasi et al., 2021)	This paper provides a comprehensive review of applications of deep learning in NTMA.

		through gaining insight, traffic data integration, traffic data validation, traffic data security, and traffic data acquisition. The unprecedented increase in the number of connected nodes and the volume of data amplifies the network complexity, calling for continuing studies to analyze and monitor the networking performance. Furthermore, the availability of the massive and				

		heterogeneous amount of traffic data necessitates adopting new approaches for monitoring and analyzing network management data. Due to these challenges, most works focus specifically on one aspect of NTMA, e.g., anomaly detection, traffic classification, or QoS				
24	Intelligent malware detection based on graph convolutional [2022]	Malware has seriously threatened the safety of computer systems for a long time. Due to the rapid	Due to the difference in malware, feature extraction is difficult, which is not conducive to applying traditional neural networks. To solve the problem, the authors use the flexibility of GCN input to design a malware detector based on GCN to adapt	In future work, the authors will focus on the research of an adaptive detection model based on GCN so that the	(S. Li et al., 2022)	Aiming to solve traditional static and dynamic detection problems, this paper proposes a novel approach to malware

		development of anti-detection technology, traditional detection methods based on static and dynamic analysis have limited effects. With its better predictive performance, AI-based malware detection has been increasingly used to deal with malware in recent years. However, due to the diversity of malware, extracting features from malware is difficult, making	to the differences of malware. The specific method is to extract the API call sequence from the malicious code and generate the directed cyclic graph, use the Markov chain to extract the characteristics of the graph, and then use GCN to realize classification. They have also done an evaluation compared with other machine learning algorithms. The results show that the method performs better in most detection, and the highest accuracy is 98.32%. The research found that the technology has potential adaptability, but it has not been realized yet.	malware detection system has a stronger adaptive ability to reduce the cost of personnel of malware detection		detection based on an application programming interface (API) call sequence and deep learning algorithm. Firstly, the API call relation is extracted, and the ordered cycle graph is constructed based on the Markov chain. Then, the graph convolution neural network (GCN) detects malware. Then, the performance analysis and comparison are carried out.

		malware detection not conducive to the application of AI technology.				
25	Ensemble dynamic behavior detection method for adversarial malware [2022]	Behavior-based malware detection approaches combined with deep learning techniques effectively against unknown malware and malware variants. However, such approaches are vulnerable to adversarial attacks. Adversarial malware is carefully optimized to evade detection	The authors conducted extensive experiments over large benign and malicious instances and demonstrated a generic, query-efficient gray-box adversarial attack to evaluate our model. The experimental results indicate that, compared with the individual classifiers, the detection accuracy is improved by up to 2.55%~, 11.34% (without anti-attack), 8.64%~, 21.33% (random perturbation), and 10.07%~21.34% (benign perturbation) respectively. To sum up, our method provides better effectiveness, generality, and resiliency in the absence of a constant re-training of the detector needed to cope with the evolution of adversarial malware.	In our future work, the authors hope to develop our proposed theory further to fundamentally alleviate or even solve the problems and challenges of adversarial malware. We will continue to try different policies of behavior feature extraction, find various adversarial attack characteristics, and further	(Jing et al., 2022)	In this paper, the authors propose an Ensemble adversarial dynamic behavior detection method aiming at Immediacy, Locality, and Adversary, called Ensila, which overcomes the limitations above. Ensila only requires a more straightforward but critical feature type, i.e., API call sequence, which

		by embedding numerous anti-detection techniques, e.g., inserting irrelevant API calls or using API calls in loops during the program execution to mask the malicious intentions.		construct an adaptive Ensila, which allows the ensemble schemes to be periodically updated as (adversarial) malware evolve.		is the most promising approach to characterize the real malware behavior as each API call acts as an interface that the programs use to request a service from the operating system's kernel.
26	Robust deep learning early alarm prediction model based on the behavioral smell for android malware [2022]	Due to the widespread expansion of the Android malware industry, malicious Android process mining became necessary to understand their behavior. Nevertheless, mining	The authors overcame the problem of massive feature size and complex associations by encapsulating related features in a few cluster classes. Accordingly, the cluster classes are exchangeably used to represent the features in the original calling sequences. Regarding substantially long sequences, experimental results showed that their model could predict whether a process is behaving maliciously or not	In future work, the authors aim to incorporate other behavioral-driven heuristics to keep our models adaptive against new malware threats.	(Amer & El-Sappagh, 2022)	The authors introduced a model that analyses malicious Android processes in this paper. Our model relies on various static and dynamic features.

		malicious Android processes have become a prominent obstacle due to the complexities of size, length, and associations of some essential and distinguishing Android applications' features, such as API calls and system calls. The malicious process mining obstacle is coupled with the increasing rate of zero-day attacks, with no prior knowledge about those behaviors. Hence, malware detection alone	based on rapid-sequence-snapshot analysis. Their proposed model counted on the LSTM model to classify the reformed API and system call sequence snapshots. Moreover, the authors used ensemble machine learning classifiers to classify Android permissions. They trained the LSTM model using random snapshots of the newly formed API and system call cluster sequences. We tested our model against common ransomware attacks. Their trained LSTM model showed stable performance at a particular snapshot size. The model showed competitive accuracy in predicting new sequences. Accordingly, they proposed an early alarm solution for blocking malicious payloads instead of identifying them after their fulfillment. Hence, they can avoid the cost of future damage.			

		is no longer enough; we need new methodologies to predict malicious behaviors early.				
27	Robust Intelligent Malware Detection Using Deep Learning [2019]	Security breaches due to attacks by malicious software (malware) continue to escalate, posing a major security concern in this digital age. With many computer users, corporations, and governments affected due to an exponential growth in malware attacks,	The authors evaluate the classical MLAs and deep learning architectures for malware detection, classification, and categorization using public and private datasets. Second, they remove all the dataset bias in the experimental analysis by splitting different public and private datasets to train and test the model in a disjoint way using different timescales. Third, their significant contribution is proposing a novel image-processing technique with optimal parameters for MLAs and deep learning architectures to arrive at an effective zero-day malware detection model. Finally, a comprehensive	Dimensionality reduction techniques to get a better classification rate can be thoroughly discussed to enhance the proposed method's performance in this study as future work. In future work, the spatial pyramid pooling (SPP) layer can allow images of any size to be	(Vinayakumar et al., 2019)	This paper evaluated classical machine learning algorithms (MLAs) and deep learning architectures based on Static analysis, Dynamic analysis, and image processing techniques for malware detection and designed a highly scalable framework

		malware detection continues to be a hot research topic. Current malware detection solutions adopting static and dynamic analysis of malware signatures and behavior patterns are time-consuming and have proven ineffective in identifying unknown malware in real time. Recent malware uses polymorphic, metamorphic, and evasive techniques to change the malware	comparative study of their model demonstrates that their proposed deep learning architectures outperform classical MLAs. Their novelty in combining visualization and deep learning architectures for static, dynamic, and image processing-based hybrid approach applied in a big data environment is the first of its kind toward achieving robust intelligent zero-day malware detection.	used as input. This learns features at variable scales and can be put between the sub-sampling layer and the fully connected layer to improve our model's flexibility. The robustness of the deep learning architectures is not discussed in the proposed work. This is one of the significant directions toward future work since malware defection is an essential application in safety-critical		called ScaleMalNet to detect, classify and categorize zero-day malware.

		behaviors and generate much new malware. Such new malware is predominantly variants of existing malware, and machine learning algorithms (MLAs) have been employed recently to conduct effective malware analysis. However, such approaches are time-consuming as they require extensive feature engineering, learning, and representation.		environments. However, a single misclassification can cause several damages to the organization.		

28	Machine Learning-Based File Entropy Analysis for Ransomware Detection in Backup Systems [2019]	With the advent of big data and cloud services, user data has become an important issue. Although various detection and prevention technologies are used to protect user data, ransomware that demands money in exchange for one's data has emerged. File- and behavior-based detection methods have been investigated to detect and prevent ransomware. Nevertheless, we still face ransomware	The authors developed a method for detecting ransomware-infected files using machine learning models that measure file entropy for the backup system. Even if the user system is attacked with ransomware, the suggested approach can recover the original file from the backup system by recognizing ransomware-affected files synced to the backup system. Compared to existing detection methods, the study's findings show that the suggested approach has a high detection rate with low false positive and false negative rates.	In the future, the authors will obtain results for various file formats and study a method to artificially detect ransomware by deriving the optimized values and parameters for each user based on the backup files of each user.	(K. Lee et al., 2019)	This paper proposes a method to detect files infected with ransomware based on the entropy of the files. The proposed method uses a feature that appears in encrypted files based on the behavior of the ransomware encrypting the files. One of the features of the cipher text is uniformity. In this paper, entropy is used as one of the methods to measure uniformity. Entropy can be

		threats, as it is difficult to detect and prevent ransomware containing unknown malicious codes. In particular, these methods are limited because they cannot detect ransomware for backup systems such as cloud services. For instance, if files infected with ransomware are synchronized with the backup systems, the infected files will not be able to be restored through the backed-up files.				measured using various methods, with NIST 800-90b representative among them.
29	Artificial	Nowadays,	The ISMS is used in production	The authors will	(K. Lee et al.,	In this paper, the

	intelligence and big data driven IS security management solutions with applications in higher education organizations [2021]	securing information systems has become a challenge like never. Failing in this endeavor may lead to severe consequences. For example, many security breaches have gone viral lately, like the SolarWinds attack and Microsoft Exchange security flaws. Such attacks may also affect public authorities, even the police. Usually, these consequences result from not paying attention	at Riga Technical University and can be adapted for use in other organizations. The proposed platform is based primarily on free and open-source tools and allows to prevent or minimize the consequences of malware's activity with little impact on the employee's privacy. The presented NFAI detection module detects malware activity by extracting features from NetFlow data within a 10-minute interval and feeding it into several trained classifiers. ISMS does not rely solely on the NFAI module alone; it uses an ensemble of modules and algorithms to increase malware detection accuracy. In addition, the presented IS security management system can be employed in a real-time environment. Its NFAI detection module allows identifying of an infected device as soon as it starts to communicate with the botnet (a logical collection of Internet-connected devices such	continue to expand the ISMS platform further by adding different modules based on the current threat level in the IS security landscape. Further, they plan to introduce different automated actions based on the identified risk level. Low-risk alerts could be only informative, for example, if the user has unwanted software installed (e.g., click gatherers, redirectors). In contrast, high-	2019)	authors focus on methods to detect botnets using supervised machine learning algorithms widely used in previous studies. Their article focuses on the artificial intelligence (AI) driven NetFlow data analysis (NFAI) module. Module extracts significant NetFlow features and uses machine learning algorithms to detect malware.

		to patches released by vendors, but in the case of SolarWinds, there is another possible reason – a built-in password. The problem with security nowadays, a part of visible security breaches, is invisible attacks and data exfiltration, usually done by botnet members.	as computers, smartphones, or IoT devices whose security has been breached and control ceded to a third party) command and control center to obtain new commands. The presented NFAI module has been validated in the production environment and identified infected devices which were not detected by antivirus software nor by firewall or Intrusion Detection System.	risk alerts could be acted upon immediately.		
30	MERLIN -- Malware Evasion with Reinforcement Learning [2022]	In addition to signature-based and heuristics-based detection techniques, machine learning (ML) is widely used to	The DQN model achieves outstanding results with Malconv and Grayscale, with a respective evasion rate of 100% and 98%. On Ember, its evasion rate reached 67%, which motivated us to develop a better technique using the	The prototype can also generate new datasets of undetectable malware to re-train ML detection models. The	(Quertier et al., 2022)	In this paper, the authors propose a method using reinforcement learning with DQN and REINFORCE algorithms to

		generalize new, never-before-seen malicious software (malware). However, it has been demonstrated that ML models can be fooled by tricking the classifier into returning the incorrect label. These studies, for instance, usually rely on a prediction score that is fragile to gradient-based attacks. In the context of a more realistic situation where an attacker has very little information about the outputs of a	REINFORCE algorithm. To our knowledge, it is the first time such an algorithm has been used for malware evasion. We train to REINFORCE against Ember, and our results show a slight improvement over DQN with an increase in the evasion rate from 67% to 74.2% without any impact on training time. We then challenge a well-known commercial AV. Once again, REINFORCE shows that it performs better than DQN, with a significant increase in the evasion rate from 30% to 70%. A key element of our work is our ability to compile a vulnerability report listing the most efficient actions to transform a malicious PE file and make it undetectable by the model under attack. In other words, we can identify the detection model weaknesses and the most effective actions to defeat a given AV. Security experts can then leverage these insights to understand why a	authors believe that their work will improve malware detection tools in the future and strengthen antivirus software by providing analysts with vulnerability reports.		challenge two state-of-the-art ML-based detection engines (MalConv & EMBER) and a commercial antivirus (AV) classified by Gartner as a leader AV [2]. Our method combines several actions, modifying a Windows portable execution (PE) file without breaking its functionalities. Our approach also identifies which actions perform better and compiles a detailed vulnerability

		malware detection engine, modest evasion rates are achieved	detection engine failed and react accordingly. Finally, our RL framework makes it possible to generate new malware variants and thus create a database of never-before-seen malicious files. This database could be a preventive asset to manage potential malware variants proactively.			report to help mitigate the evasion. Finally, we demonstrate that REINFORCE achieves excellent evasion rates even on a commercial AV with limited information.

Summary

In this chapter, we defined malware and discussed specific security incidents that resulted in some businesses' data loss and financial loss—provided background information on some of the hackers that conducted cyber-attacks against various organizations. For example, the history of how the malware was invented and how cybercriminals have used it to organize crime was shared, indicating that it is a multi-million-dollar enterprise—provided many reviews of malware-related literature.

Chapter III: Methodology

Introduction

AquaSec was used in this study to scan artifacts for vulnerabilities, malware, sensitive data, and other risks during development and staging (*Cloud Native Security Platform (CNAPP)*, n.d.). It compares cloud services, infrastructure as code templates, and Kubernetes configuration to best practices. Aqua establishes zero-trust networking and detects suspicious activity, including zero-day attacks. As a result, the current study uses a quantitative approach and relies on Statistical tools for data analysis.

Design of the Study

The study used a quantitative research method to provide insights into the vulnerability posture and prioritize remediation. The quantitative research method involves an experiment to scan a docker image to detect Critical/High Vulnerabilities, malware, and sensitive data before deploying it into the cloud. Furthermore, the study presented an experiment to understand better how to prevent attacks before they happen and be stopped. Finally, the experiment allowed us to define, configure and manage runtime policies in conjunction with security controls that determines which images will be allowed to run on a docker host and overall secure your application builds.

Data Collection

The data collected and analyzed in this study came from a controlled environment. Aqua comprehensively scans container images and serverless functions for known vulnerabilities, embedded secrets, OSS licensing issues, hidden malware, and configuration issues (*Automate DevSecOps*, n.d.). Aqua makes it easy to gather data because it provides real-time actionable information on vulnerability and configuration remediation, fed back to developers within their CI/CD tools, sent via Slack, or as a ticket in Jira. In addition, the data obtained from Aqua provides up-to-date statistics to evaluate the objective of the problem discussed in this study. Aqua provided us with some forensic data such as Host, Host IP, Image Name, Image ID, Image Hash, Container Name, Container ID, Action, Kubernetes Cluster, Aqua Response, Details, Group, Stamp, Entity, Image, Action taken, Policy, Failed control, Time Stamp. If any images fail to adhere to our security controls and runtime policies, we use the forensic data to understand why that occurred.

Tools and Techniques

The specific tool used in addressing the problem is Aqua. The main capabilities of Aqua are Cloud Security Posture Management (CSPM), Vulnerability Scanning, and Dynamic Threat Analysis (DTA). Aqua uses CSPM to scan, monitor, and remediate configuration issues in public cloud accounts according to best practices and compliance standards across Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP), and Oracle Cloud Infrastructure (OCI). As for Vulnerability scanning, Aqua scans the container images to detect known vulnerabilities (CVEs) and other security issues during the development cycle to gain insights into the vulnerability posture and prioritize remediation and mitigation according to contextual risk. According to Aqua, vulnerability scanning is delivered as part of Aqua image scanning as new vulnerabilities and exploits are discovered and published daily; scanning a container image once on-push is not enough. Aqua re-scans the images daily to ensure you are always aware of new risks in the container images as they become known. Aqua stated that Dynamic Threat Analysis (DTA) runs container images in an isolated sandbox environment that monitors and detects Indicators of Compromise (IOC) such as container escapes, malware, crypto miners, code injection backdoors, network anomalies, and more (*Aqua SaaS Overview*, n.d.). In addition, some security policies were enforced to get visibility in non-compliant docker images, such as sensitive data, malware, and vulnerability score.

Hardware and Software Environment

The study uses the SaaS Aqua model, cloud security & compliance solution designed to help developers and DevOps teams protect applications as they are built and the infrastructure they are deployed on. Aqua Platform is the complete Cloud-Native Security Platform that protects your entire stack, on any cloud, across VMs, containers, and serverless.

Summary

The approach to the study has been presented in this chapter. Our study followed the quantitative research method that involved an experiment. The experiment requires us to scan a docker image to detect Critical/High Vulnerabilities, malware, and sensitive data before deploying it into the cloud. The experiment presented by the study provides real-time actionable information on vulnerability and configuration remediation, fed back to developers within their CI/CD tools, sent via Slack, or as a ticket in Jira.

Chapter IV: Data Presentation and Analysis

Aqua scans images for vulnerabilities, malware, embedded secrets, configuration issues, and OSS licensing, allowing you to create custom policies that determine which images will be allowed to run on your Docker hosts. Based on a constantly updated data stream, Aqua's vulnerabilities database is aggregated from multiple sources and consolidated to ensure that only the most up-to-date data is included, increasing accuracy and reducing false positives and negligible CVEs. The security controls in Figure 1 Prevent developers from deploying applications into the cloud if one of the security controls is not in compliance. We found Malware, MicroEnforcer, Sensitive Data, Superusers, Forensic, and Vulnerability Scores to be our study's most imperative security controls. You can always customize the security controls base on the need of your organization. Developers will not be able to configure docker images as root users. Any vulnerability from 7 to 10 will stop an application from being deployed into the cloud. Images must be free from sensitive data and malware if developers want to deploy an application into the cloud.

Figure 1 Aqua allowed us to define, configure, and manage Runtime Policies. Aqua secures your application builds, infrastructure, and workloads by your organization's security policies (including requirements for regulatory compliance). Many security-related activities are categorized as either assurance or enforcement. Assurance can scan applications and infrastructure for potential security issues.

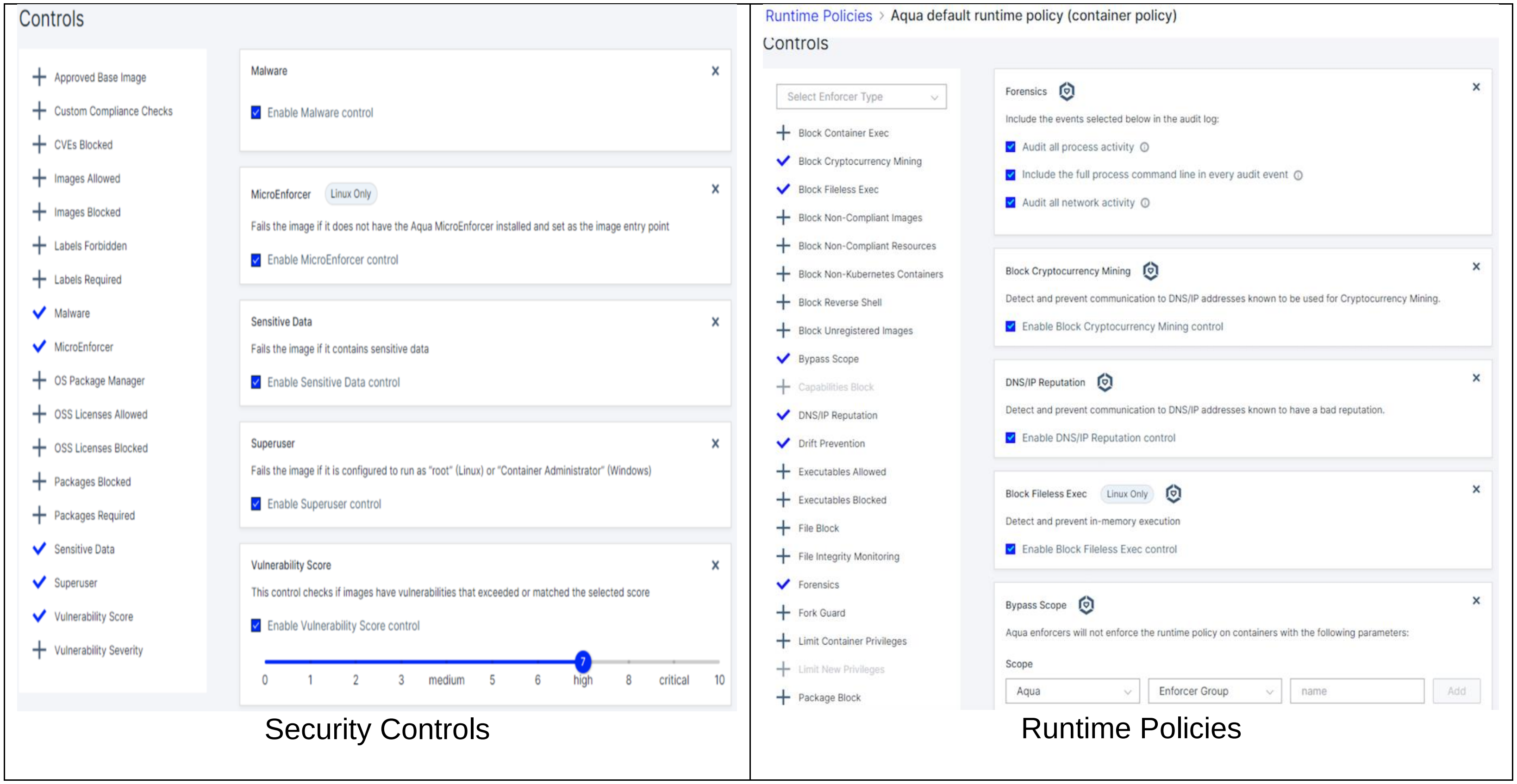
Enforcement can prevent, at runtime, workload, and infrastructure from performing potentially insecure operations.

A runtime policy has three parts (Bland, n.d.):

- Scope — You can create a blanket policy that can be applied to the entire environment. You can also use granular scoping mechanisms based on image attributes, container attributes, or Kubernetes constructs like pods, deployments, etc.
- Enforcement Mode — You can apply the policy in an Audit mode for the current state assessment of your environment, which allows you to discover and provides deeper insight into cloud-native workloads. Switch to the Enforcement mode for actively blocking or enforcing the specified policies.
- Controls — These are security-related tests that the Aqua Enforcer conducts while the workload run

Figure 1

Security controls and Aqua default runtime policy (container policy)



Security Controls

Runtime Policies

Aqua scans container images based on a constantly updated stream of aggregate vulnerability data sources (CVEs, vendor advisories, and proprietary research), ensuring up-to-date, broad coverage while minimizing false positives. Additionally, find malware, embedded secrets, OSS licenses, and configuration issues in your images to further reduce the attack surface (*The Leading Container Security Solution for Cloud Native Apps*, n.d.). In Figure 2, vulnerabilities are color-coded under the “security issues” tab. That is how Aqua demonstrates the Critical, High, Medium, and Low vulnerabilities to their customers. Aqua did not find malware, embedded secrets, OSS licenses, or configuration issues with our docker image.

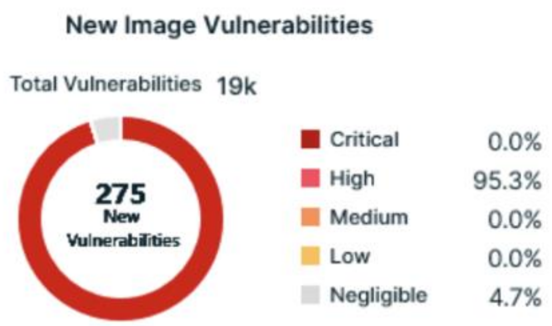
Figure 2

The docker image in Aqua

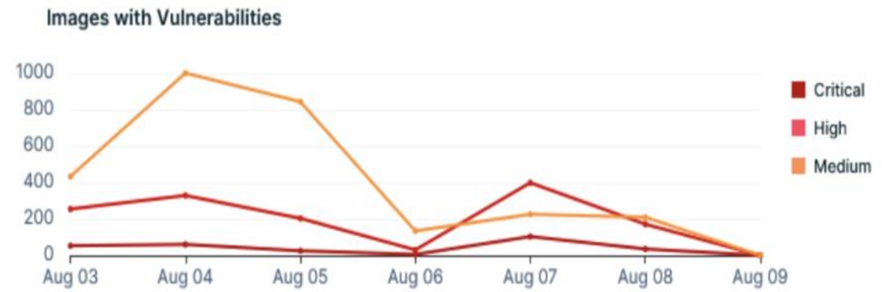
Once Aqua Scan was completed, we downloaded an “executive summary” document that provided an overview of the attacks and methods used and actionable advice to security executives on protecting against this new and growing breed of attacks.

Figure 3

The Executive Summary document or report

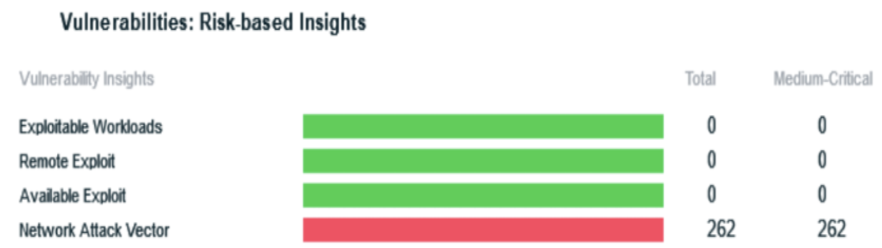


The pie chart shows the distribution of the vulnerabilities by severity, providing insight into the security of your build.



The individual lines show the number of images according to the highest severity of vulnerability found in the image. In addition, the graph indicates how your build security process is progressing over time.

Use the findings in this widget to understand which mitigation actions you can take. You can also see the number of vulnerabilities mitigated by Aqua vShields, and the number of vulnerabilities acknowledged by the user.

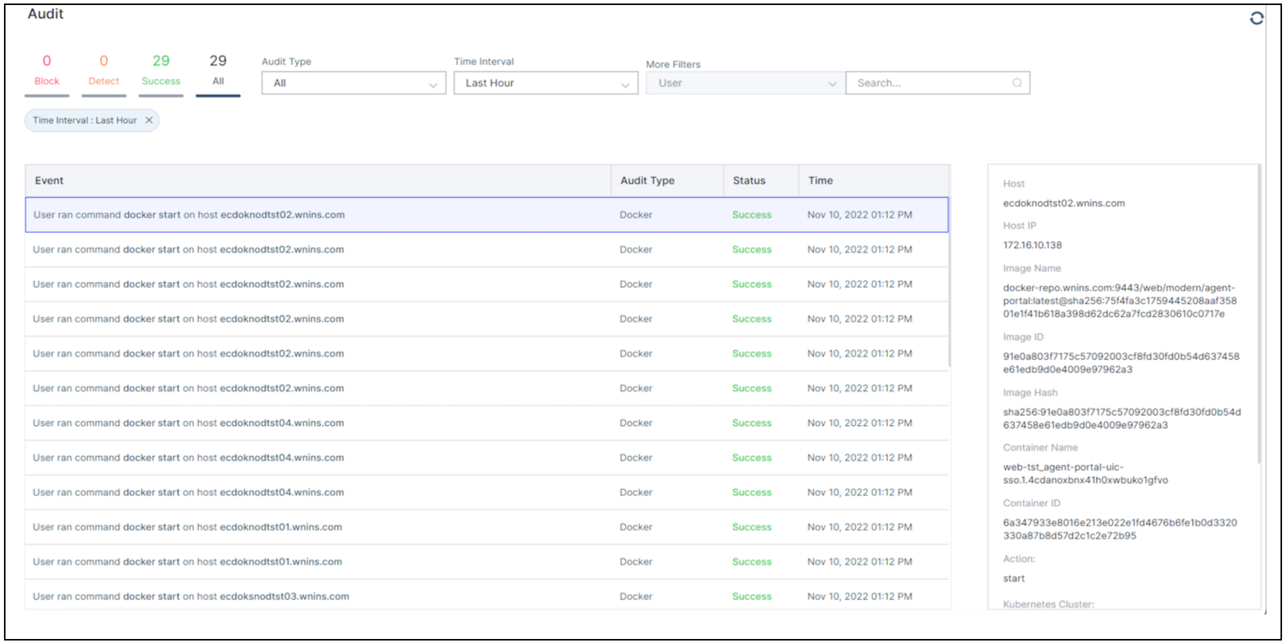


The Risked-based Insights widget shows the total number and severity of the detected vulnerabilities with different exploits. The number of the most severe vulnerabilities (medium, high, and critical) of each kind is also shown.

As revealed in Figure 4, Aqua provided us with an Audit trail and forensic data on each event associated with an image. In addition, the audit trail will provide visibility to security professionals to analyze how malware is similar and differs from previously identified malware.

Figure 4

Audit Trail



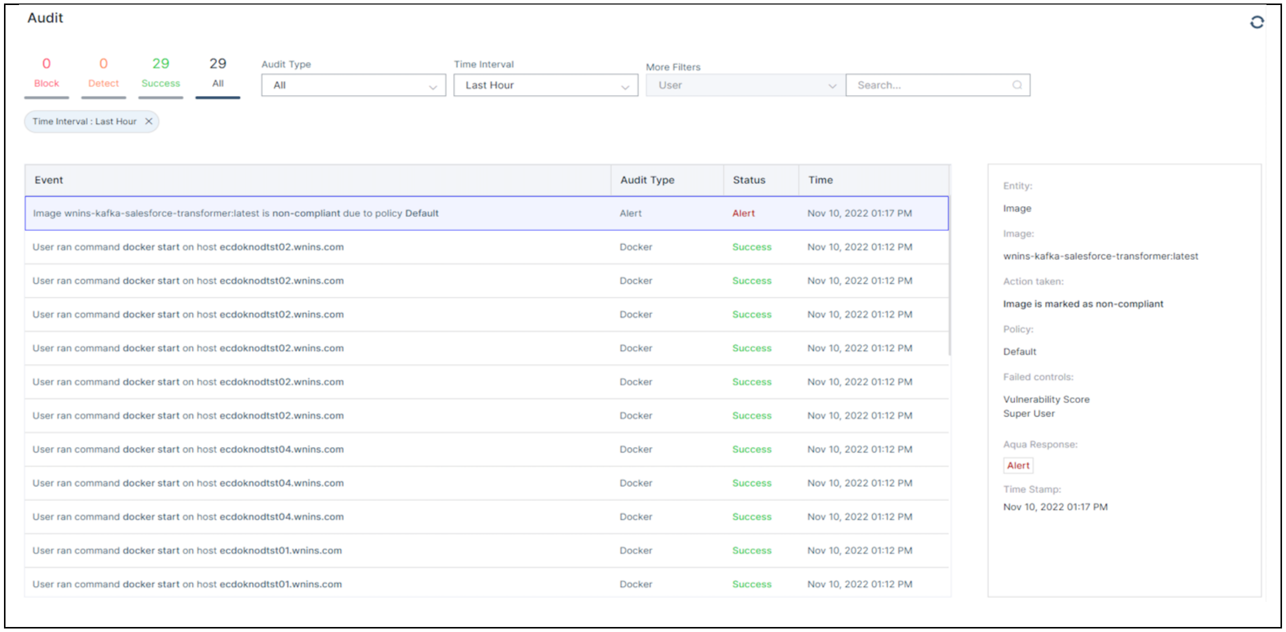


Figure 5 depicts the forensic data obtained from Aqua's malware analysis and detection on our image. As noted in Figure 5, the image "wnins-kafta-salesforce-transformer:latest" have two failed controls, vulnerability score, and super User, which triggered an alert in Aqua. By addressing known exploits, this forensic data will assist security professionals and DevOps in ensuring that a Docker image is secure to deploy in the production environment. In addition, the forensic data will mitigate the possibility of cyber criminals exploiting an image in a production environment and allow security experts to enhance their defense mechanisms.

Figure 5

Forensic Data from Aqua

Host
ecdoknodtst02.wnins.com

Host IP
172.16.10.138

Image Name
docker-repo.wnins.com:9443/web/modern/agent-portal:latest@sha256:75f4fa3c1759445208aaf35801e1f41b618a398d62dc62a7fcd2830610c0717e

Image ID
91e0a803f7175c57092003cf8fd30fd0b54d637458e61edb9d0e4009e97962a3

Image Hash
sha256:91e0a803f7175c57092003cf8fd30fd0b54d637458e61edb9d0e4009e97962a3

Container Name
web-tst_agent-portal-uic-sso.1.4cdanoxbnx41h0xwbuko1gfvo

Container ID
6a347933e8016e213e022e1fd4676b6fe1b0d3320330a87b8d57d2c1c2e72b95

Action:
start

Kubernetes Cluster:

Cluster-1

Aqua Response:
Success

Details
Unauthorized image. Image was not blocked because it already had running containersEnforcer

Group:

Aqua_TestTime

Stamp:
Nov 10, 2022 01:12 PM

Entity:
Image

Image:
wnins-kafka-salesforce-transformer:latest

Action taken:
Image is marked as non-compliant

Policy:
Default

Failed controls:
- Vulnerability Score
- Super User

Aqua Response:
Alert

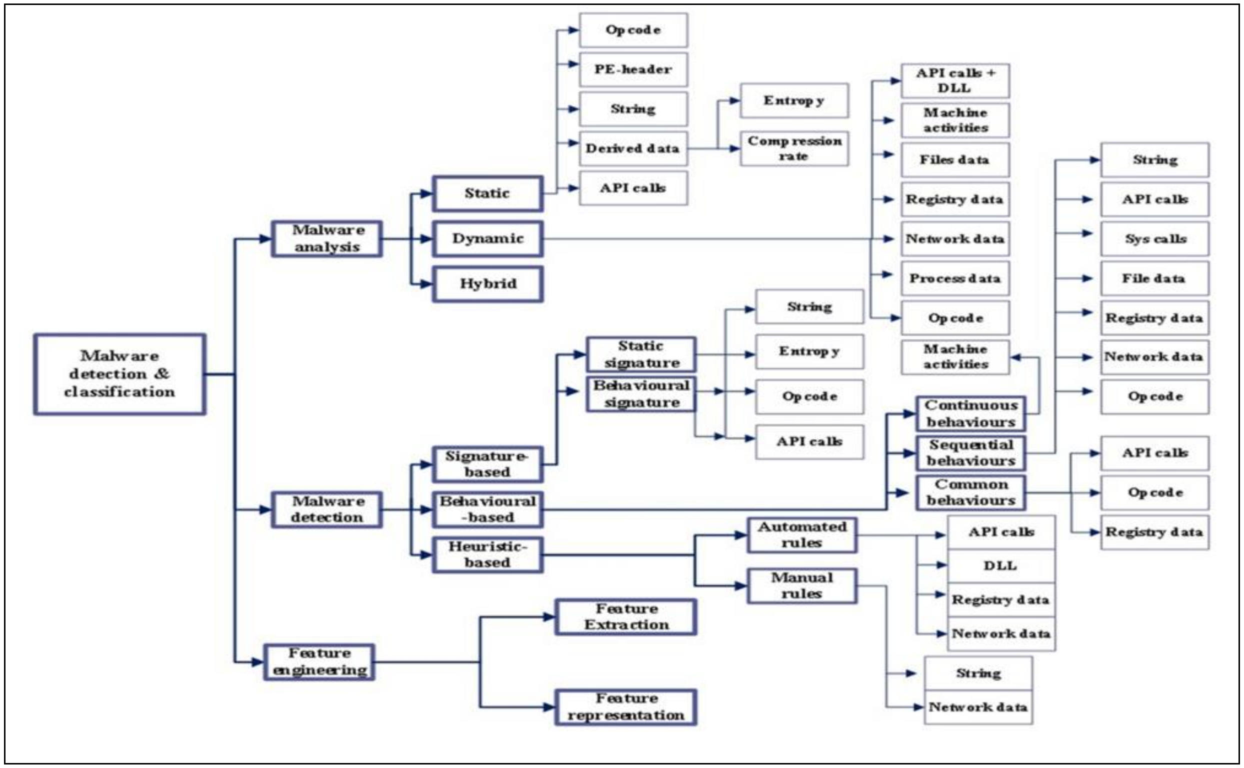
Time Stamp:

Nov 10, 2022 01:17 PM

Figure 6 demonstrates malware analysis and detection taxonomy, where the analysis approaches are presented as static, dynamic, and hybrid, showing the frequently used data types with each analysis approach. Regarding malware detection, sub-detection approaches which go deeper than the well-known approaches, signature-based, behavioral-based, and heuristic-based, have been presented. In addition, static and dynamic signatures, continuous, sequential, common behavioral, and automated and manual rules are displayed as categories of the major detection approaches and associating each sub-detection approach with the most used data types.

Figure 6

Malware Analysis and Detection Taxonomy (Aboaoja et al., 2022)



Malware Analysis Discussion

Static Analysis

The static analysis approach has been widely utilized by exploring the source code without running the executable files to extract a unique signature used to represent the file under investigation. Several types of static data can be collected via static analysis, including PE-header data and derived data such as string-based entropy and compression ratio. Additionally, static analysis tools, such as IDA pro disassembler and Python-developed modules, are also used to collect static opcode and API calls (Aboaoja et al., 2022). In addition, static analysis, although capable of tracking all potential execution pathways, is impacted by packing and encryption schemes.

Dynamic Analysis

Several researchers performed a dynamic analysis approach to collect various data types from differentiating between malware and benign files by running the executable files in isolated environments, virtual machines (VM), or emulators to monitor the executable file behavior during the run-time to collect the desired dynamic data. Various kinds of data have been collected utilizing a dynamic analysis approach. Malicious activities can be dynamically represented using both executable file behavior and by retaining memory images during run-time (Aboaoja et al., 2022). The behaviors of executable files are identified by collecting the initiated API calls, machine activities, file-related data, registry, and network data. In addition, an opcode-based memory image can be used to depict malicious activity dynamically.

Hybrid Analysis

Some previous studies combined data extracted through static and dynamic analysis to reduce the drawbacks of both analysis approaches and achieve a higher detection rate. Different tools, including Cuckoo sandbox, IDA pro disassembler, and OlleyDbg, are employed to collect dynamic and static data. Then hybrid feature sets are created based on several types of data, such as string, opcode, API calls, and others (Aboaoja et al., 2022). Although the hybrid analysis technique has advantages over static and dynamic analysis, it also has limitations.

Malware Detection Discussion

Signature-Based

Static string-based signatures have been generated to detect malicious VBasic software by representing the obtained strings using frequency vectors while generating static signatures based on n-grams and binary vectors. In addition, static and behavioral signature-based malware detection models suffer from low detection rates when classifying unknown signatures that may be linked to unknown malware or different variants of known malware (Aboaoja et al., 2022).

Behavioral Based

After monitoring the executable files in an isolated environment and collecting the exhibited behaviors, features extraction techniques have been developed to extract the sensitive features by which the developed model can classify the known malicious behaviors and any behavior that seems to be like them concerning false positive behaviors. The ability to identify novel malware behaviors and the known ones based on collecting behaviors during run-time has made this approach more valuable than the signature-based approach (Aboaoja et al., 2022).

Heuristic-Based

A heuristic-based approach has been used in various research by generating generic rules that investigate the extracted data, which are given through dynamic or static analysis to support the proposed model of detecting malicious intent. The generated rules can be developed automatically using machine learning techniques, the YARA tool, and other tools or manually based on the experience and knowledge of expert analysts (Aboaoja et al., 2022). In addition, several experiments have been conducted to establish malware detection models in which choices are made based on automated behavioral rules built utilizing machine learning techniques and the YARA tool.

Summary

This chapter discussed the experiment conducted by this study. The experiment addressed the security control and runtime policies enforced on the docker image. First, we selected Malware, MicroEnforcer, Sensitive Data, Superuser, Forensic, and Vulnerability Score security controls deemed imperative to our study. These security controls determine which images can run on a docker host. Additionally, it provides forensic data if an image doesn’t meet security controls and runtime policies. Finally, we analyzed the security issues associated with our docker image based on low, Medium, High, and critical criticality. Finally, we were presented with an executive summary document that advises security experts on protecting against a new and growing breed of attacks. Furthermore, we included malware Analysis and Detection Taxonomy by Aboaoja et al. (2022) and then discussed malware analysis and detection classification.

Chapter V: Results, Conclusion, and Recommendations

Introduction

This chapter concludes the study. It summarizes our findings, future research, and the study's conclusion.

Results

Figure 1 presents security controls such as Malware, MicroEnforcer, Sensitive Data, Superusers, Forensic, and Vulnerability Scores. The security controls minimize the security risk and enforce compliance associated with our docker. Additionally, in Figure 1, We added a layer of Aqua's defense in depth: the runtime policies (Aqua default runtime policy), such as Block Cryptocurrency Mining, Block Files Exec, Bypass Scope, DNS/IP Reputation, and Drift Prevention. All these runtime policies in Figure 1 are imperative to this study, but "Drift Prevention" caught my attention. Drift prevention is the cloud-native answer to malware, worms, and zero-day exploits. It's also one of the best things to happen to security since the firewall (Korren, n.d.). Drift Prevention prohibits running files that are not a part of the original image from running, ensuring no changes are made to the image after it is instantiated into a container. This prevents hackers from downloading new malicious code to the running container. We proceeded with our docker image in our finding because it did not violate the runtime policies. Even if one of the runtime policies were violated, our docker image would not be deployed into the Kubernetes container.

Figure 2 demonstrates how Aqua, by default, scanned our image and identified 25 vulnerabilities ranging from High, Medium, and Low. Moreover, Aqua did not find malware, embedded secrets, OSS licenses, and configuration issues. Instead, figure 3 presents us with an executive summary document. This document shows us 275 new vulnerabilities associated with our docker image and newly added images to Aqua. The total number of vulnerabilities is 19k, and out of that, 95.3% was High was 4.7 was Negligible. As a result of the Aqua scan, we found 262 Network Attack Vector that attackers could exploit, and these exploits range from Medium to Critical. Some other exploits, such as Exploitable workloads, Remote Exploit, and Available Exploits, were addressed using Aqua, but zero vulnerabilities were identified. This finding helps developers and security experts manage known and published exploits.

Figure 4 revealed a series of events about our images in the registry and forensic data associated with each event. The audit trail provided security professionals visibility to analyze how malware is similar and how it differs from previously identified malware. Figure 5 presents the Forensic data collected when Aqua analyzed our images in the registry based on the customized security controls and runtime policies we have in place. The forensic data collected are “Host, Host IP, Image Name, Image ID, Image Hash, Container Name, Container ID, Action, Kubernetes Cluster, Aqua Response, Details, Group, Stamp, Entity, Image, Action taken, Policy, Failed control, Time Stamp”. If any images fail to adhere to our security controls and runtime policies, we use the forensic data to understand why that occurred. This allows security analysts to investigate security incidents associated with our images and ensures that images are secure to deploy in the production environment.

As demonstrated in Figure 5, the image "wnins-kafta-salesforce-transformer:latest" had two failed controls vulnerability score and super User, triggering an Aqua alert. Finally, Figure 6 demonstrates malware analysis and detection taxonomy. The taxonomy provided us with an understanding of malware detection and analysis approaches. As for malware detection approaches, the taxonomy elaborated on signature-based, behavioral-based, and heuristic-based, and as for malware analysis, the taxonomy presented us with static, dynamic, and Hybrid.

Conclusion

Aqua replaces outdated signature-based approaches with modern controls that leverage the cloud-native principles of immutability, microservices, and portability. Using dynamic threat analysis, machine-learned behavioral whitelisting, integrity controls, and nano-segmentation, Aqua makes applications more secure than ever (*Modern Container Security For Cloud Native Apps*, n.d.). Using machine learning and AI to analyze and detect malware when building applications and deploying them into the cloud is an approach for enterprises to adapt. In this research, we used Aqua Dynamic Threat Analysis to scan our docker image from the registry, providing us with forensic data such as container escapes, malware, crypto miner, injection backdoors, and network anomalies. To prevent an array of attack vectors, the findings of our investigation were provided in-depth to understand which mitigation actions we can take, and Aqua vShields mitigated some vulnerabilities. The results gave us some visibility of exploitable vulnerabilities, notably 262 network attack vectors from Medium to Critical and 275 new image vulnerabilities, of which 95.3% were High. These identified vulnerabilities mitigate false positives and provide security experts with actual vulnerabilities that threat actors could exploit. In addition to the experiment, the forensic data collected in the build and deployment phase are exploitable vulnerability, Critical/High Vulnerability Score, Misconfiguration, Sensitive Data, and Root User (Super User).

Furthermore, Aqua validated our security controls and runtime policies by providing us with forensic data associated with two events on the Audit page. The event that triggered an alert provided forensic data such as Entity, Image, Action taken, Policy, Failed control, and Time Stamp, whereas the event that passed our security controls and runtime policies provided forensic data such as Host, Host IP, Image Name, Image ID, Image Hash, Container Name, Container ID, Action, Kubernetes Cluster, Aqua Response, Details, Group, and Stamp. If any images fail to adhere to our security controls and runtime policies, we use the forensic data to understand why that occurred. This allows security analysts to investigate security incidents associated with our images and ensures that images are secure to deploy in the production environment. This is significant because it will enable security analysts to focus on what’s most important by identifying security vulnerabilities and their root causes.

In conclusion, the Aqua Dynamic Threat Analysis technique, in conjunction with customized security controls and Aqua runtime policies, shows to prevent unapproved images from running anywhere in our environment. This allows DevOps to develop and deploy applications to the cloud securely. Finally, the taxonomy presented in the study provided us with an understanding of malware detection approaches and analysis approaches so that security professionals can save time by allowing them to focus on the specific approach required to analyze and detect malware.

Future work will expand on this technology to detect additional attack vectors by expanding on the Dynamic Threat Analysis by integrating it with deep learning techniques to enhance the detection level of zero-day attacks. In future research, the docker image should be deployed into the production environment, and Aqua should be open-source and have some knowledge base articles. Aqua should expand on its dynamic threat analysis by including deep learning techniques in its tool, which will enhance the detection process of malware and effectively analyze malware

References

Abbasi, M., Shahraki, A., & Taherkordi, A. (2021). Deep Learning for Network Traffic Monitoring and Analysis (NTMA): A Survey. *Computer Communications*, *170*, 19–41. https://doi.org/10.1016/j.comcom.2021.01.021

Aboaoja, F. A., Zainal, A., Ghaleb, F. A., Al-rimy, B. A. S., Eisa, T. A. E., & Elnour, A. A. H. (2022). Malware Detection Issues, Challenges, and Future Directions: A Survey. *Applied Sciences*, *12*(17). https://doi.org/10.3390/app12178482

Aboaoja, F. A., Zainal, A., Ghaleb, F. A., & Saleh Al-rimy, B. A. (2021). Toward an Ensemble Behavioral-based Early Evasive Malware Detection Framework. *2021 International Conference on Data Science and Its Applications (ICoDSA)*, 181–186. https://doi.org/10.1109/ICoDSA53588.2021.9617489

Ahmed, N., Amin, R., Aldabbas, H., Koundal, D., Alouffi, B., & Shah, T. (2022). Machine Learning Techniques for Spam Detection in Email and IoT Platforms: Analysis and Research Challenges. *Security and Communication Networks*, *2022*, 1–19. https://doi.org/10.1155/2022/1862888

Alo, U. R., Nweke, H. F., & Ele, S. I. (2021). Machine Learning-Based Framework for Automatic Malware Detection using Android Traffic Data. *Journal of Theoretical and Applied Information Technology*, *99(15)*.

Amer, E., & El-Sappagh, S. (2022). Robust deep learning early alarm prediction model based on the behavioural smell for android malware. *Computers & Security*, *116*, 102670. https://doi.org/10.1016/j.cose.2022.102670

Aqua SaaS Overview. (n.d.). Aqua Support. Retrieved October 18, 2022, from https://support.aquasec.com/support/solutions/articles/16000111531-aqua-saas-overview

Arabo, A., Dijoux, R., Poulain, T., & Chevalier, G. (2020). Detecting Ransomware Using Process Behavior Analysis. *Procedia Computer Science*, *168*, 289–296. https://doi.org/10.1016/j.procs.2020.02.249

Asam, M., Hussain, S. J., Mohatram, M., Khan, S. H., Jamal, T., Zafar, A., Khan, A., Ali, M. U., & Zahoora, U. (2021). Detection of Exceptional Malware Variants Using Deep Boosted Feature Spaces and Machine Learning. *Applied Sciences (2076-3417)*, *11*(21), 10464. https://doi.org/10.3390/app112110464

Aslan, Ö., & Yilmaz, A. A. (2021). A New Malware Classification Framework Based on Deep Learning Algorithms. *IEEE Access*, *9*, 87936–87951. https://doi.org/10.1109/ACCESS.2021.3089586

Automate DevSecOps: Security and speed without compromise. (n.d.). Aqua. Retrieved October 18, 2022, from https://www.aquasec.com/use-cases/devops-security/

Ben Abdel Ouahab, I., Bouhorma, M., Lotfi, E., & Anouar Abdelhakim, B. (2020). Towards a New Cyberdefense Generation: Proposition of an Intelligent Cybersecurity Framework for Malware Attacks. *Recent Patents on Computer Science*, *15*. https://doi.org/10.2174/2666255813999201117093512

Bland, J. (n.d.). *EKSworkshop.com*. Retrieved October 18, 2022, from https://eksworkshop.com

Chen, L., Xia, C., Lei, S., & Wang, T. (2021). Detection, Traceability, and Propagation of Mobile Malware Threats. *IEEE Access*, *9*, 14576–14598. https://doi.org/10.1109/ACCESS.2021.3049819

Cloud Native Detection and Response CNDR. (n.d.). Aqua. Retrieved October 18, 2022, from https://www.aquasec.com/use-cases/cndr-cloud-native-detection-and-reponse/

Cloud Native Security Platform (CNAPP). (n.d.). Aqua. Retrieved October 18, 2022, from https://www.aquasec.com/aqua-cloud-native-security-platform/

D'Angelo, G., Palmieri, F., Robustelli, A., & Castiglione, A. (2021). Effective classification of android malware families through dynamic features and neural networks. *Connection Science*, *33*(3), 786–801. https://doi.org/10.1080/09540091.2021.1889977

Ebrahimi, M., Pacheco, J., Li, W., Hu, J. L., & Chen, H. (2021). Binary Black-Box Attacks Against Static Malware Detectors with Reinforcement Learning in Discrete Action Spaces. *2021 IEEE Security and Privacy Workshops (SPW)*, 85–91. https://doi.org/10.1109/SPW53761.2021.00021

Glossary. (n.d.). Malwarebytes. Retrieved October 13, 2022, from https://www.malwarebytes.com/glossary

Hossain Faruk, M. J., Shahriar, H., Valero, M., Barsha, F. L., Sobhan, S., Khan, M. A., Whitman, M., Cuzzocrea, A., Lo, D., Rahman, A., & Wu, F. (2021). Malware Detection and Prevention using Artificial Intelligence Techniques. *2021 IEEE*

International Conference on Big Data (Big Data), 5369–5377. https://doi.org/10.1109/BigData52589.2021.9671434

Hu, Yang., Wang, Ning., Chen, Yimin., Lou, Wenjing., & Hou, Y. Thomas. (2022). Transferability of Adversarial Examples in Machine Learning-based Malware Detection. *2022 IEEE Conference on Communications and Network Security (CNS)*, 28-36

Humayun, M., Jhanjhi, N., Alsayat, A., & Ponnusamy, V. (2021). Internet of things and ransomware: Evolution, mitigation and prevention. *Egyptian Informatics Journal*, *22*(1), 105–117. https://doi.org/10.1016/j.eij.2020.05.003

Jeon, J., Park, J. H., & Jeong, Y.-S. (2020). Dynamic Analysis for IoT Malware Detection With Convolution Neural Network Model. *IEEE Access*, *8*, 96899–96911. https://doi.org/10.1109/ACCESS.2020.2995887

Jing, C., Wu, Y., & Cui, C. (2022). Ensemble dynamic behavior detection method for adversarial malware. *Future Generation Computer Systems*, *130*, 193–206. https://doi.org/10.1016/j.future.2021.12.013

Korren, Tsvi. (2020, March 19). Blocking Attacks in Runtime with Drift Prevention. *Aqua Blog*. https://blog.aquasec.com/cloud-native-security-drift-prevention

Kouliaridis, V., Barmpatsalou, K., Kambourakis, G., & Chen, S. (2020). A Survey on Mobile Malware Detection Techniques. *IEICE Transactions on Information and Systems*, *E103.D*(2), 204–211. https://doi.org/10.1587/transinf.2019INI0003

Kumar, R., & Subbiah, G. (2022). Zero-Day Malware Detection and Effective Malware Analysis Using Shapley Ensemble Boosting and Bagging Approach. *Sensors*, *22*(7). https://doi.org/10.3390/s22072798

The Leading Container Security Solution for Cloud Native Apps. (n.d.). Aqua. Retrieved October 18, 2022, from https://www.aquasec.com/products/container-security/

Lee, K., Lee, S.-Y., & Yim, K. (2019). Machine Learning Based File Entropy Analysis for Ransomware Detection in Backup Systems. *IEEE Access*, *7*, 110205–110215. https://doi.org/10.1109/ACCESS.2019.2931136

Lee, S.-J., Shim, H.-Y., Lee, Y.-R., Park, T.-R., Park, S.-H., & Lee, I.-G. (2022). Study on Systematic Ransomware Detection Techniques. *2022 24th International Conference on Advanced Communication Technology (ICACT)*, 297–301. https://doi.org/10.23919/ICACT53585.2022.9728909

Li, C., Cheng, Z., Zhu, H., Wang, L., Lv, Q., Wang, Y., Li, N., & Sun, D. (2022). DMalNet: Dynamic malware analysis based on API feature engineering and graph learning. *Computers & Security*, *122*, 102872. https://doi.org/10.1016/j.cose.2022.102872

Li, S., Zhou, Q., Zhou, R., & Lv, Q. (2022). Intelligent malware detection based on graph convolutional network. *The Journal of Supercomputing*, *78*(3), 4182–4198. https://doi.org/10.1007/s11227-021-04020-y

McDonald, G., Papadopoulos, P., Pitropakis, N., Ahmad, J., & Buchanan, W. J. (2022). Ransomware: Analysing the Impact on Windows Active Directory Domain Services. *Sensors*, *22*(3), 953. https://doi.org/10.3390/s22030953

Modern Container Security For Cloud Native Apps. (n.d.). Aqua. Retrieved October 18, 2022, from https://www.aquasec.com/use-cases/cloud-workload-security/

Mohamad Arif, J., Ab Razak, M. F., Tuan Mat, S. R., Awang, S., Ismail, N. S. N., & Firdaus, A. (2021). Android mobile malware detection using fuzzy AHP. *Journal of Information Security and Applications*, *61*, 102929. https://doi.org/10.1016/j.jisa.2021.102929

Montes, F., Bermejo, J., Sanchez, L. E., Bermejo, J. R., & Sicilia, J. A. (2021). Detecting Malware in Cyberphysical Systems Using Machine Learning: A Survey. *KSII Transactions on Internet & Information Systems*, *15*(3), 1119–1139. https://doi.org/10.3837/tiis.2021.03.016

Mutalib, M. M. A., Zainol, Z., & Halip, M. H. M. (2021). Mitigating Malware Threats at Small Medium Enterprise (SME) Organisation: A Review and Framework. *2021 6th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE)*, 1–6. https://doi.org/10.1109/ICRAIE52900.2021.9703991

Nagahawatta, R., Lokuge, S., Warren, M., & Salzman, S. (2021). Cybersecurity Issues and Practices in a Cloud Context: A Comparison Amongst Micro, Small and Medium Enterprises, *Australasian Conference on Information Systems* p.3. http://arxiv.org/abs/2111.05993

Nawaz, M. S., Fournier-Viger, P., Nawaz, M. Z., Chen, G., & Wu, Y. (2022). MalSPM: Metamorphic malware behavior analysis and classification using sequential pattern mining. *Computers & Security*, *118*, 102741. https://doi.org/10.1016/j.cose.2022.102741

Owen, H., Zarrin, J., & Pour, S. M. (2022). A Survey on Botnets, Issues, Threats, Methods, Detection and Prevention. *Journal of Cybersecurity and Privacy*, *2*(1), 74-88. https://doi.org/10.3390/jcp2010006

Oz, H., Aris, A., Levi, A., & Uluagac, A. S. (2021). A Survey on Ransomware: Evolution, Taxonomy, and Defense Solutions, *ACM Computing Surveys*. http://arxiv.org/abs/2102.06249

Panker, T., & Nissim, N. (2021). Leveraging malicious behavior traces from volatile memory using machine learning methods for trusted unknown malware detection in Linux cloud environments. *Knowledge-Based Systems*, *226*, 107095. https://doi.org/10.1016/j.knosys.2021.107095

Patil, S., Varadarajan, V., Walimbe, D., Gulechha, S., Shenoy, S., Raina, A., & Kotecha, K. (2021). Improving the Robustness of AI-Based Malware Detection Using Adversarial Machine Learning. *Algorithms*, *14*(10). https://doi.org/10.3390/a14100297

Policies. (n.d.). Aqua Support. Retrieved October 18, 2022, from https://support.aquasec.com/support/solutions/articles/16000122298-policies

Prajapati, P., & Stamp, M. (2021). An Empirical Analysis of Image-Based Learning Techniques for Malware Classification, *ArXiv*, 1-11 http://arxiv.org/abs/2103.13827

Quertier, T., Marais, B., Morucci, S., & Fournel, B. (2022). MERLIN -- Malware Evasion with Reinforcement LearnINg. *AirXiv*, 1-19 http://arxiv.org/abs/2203.12980

Rathore, H., Sahay, S. K., Nikam, P., & Sewak, M. (2021). Robust Android Malware Detection System against Adversarial Attacks using Q-Learning. *Information Systems Frontiers*, *23*(4), 867–882. https://doi.org/10.1007/s10796-020-10083-8

Ren, A., Liang, C., Hyug, I., Broh, S., & Jhanjhi, N. (2018). A Three-Level Ransomware Detection and Prevention Mechanism. *EAI Endorsed Transactions on Energy Web*, 162691. https://doi.org/10.4108/eai.13-7-2018.162691

Rey, V., Sánchez Sánchez, P. M., Huertas Celdrán, A., & Bovet, G. (2022). Federated learning for malware detection in IoT devices. *Computer Networks*, *204*, 108693. https://doi.org/10.1016/j.comnet.2021.108693

Sahin, M., & Bahtiyar, S. (2020). A Survey on Malware Detection with Deep Learning. *13th International Conference on Security of Information and Networks*, 1–6. https://doi.org/10.1145/3433174.3433609

Sapalo Sicato, J. C., Sharma, P. K., Loia, V., & Park, J. H. (2019). VPNFilter Malware Analysis on Cyber Threat in Smart Home Network. *Applied Sciences*, *9*(13), https://doi.org/10.3390/app9132763

Souppaya, M., & Scarfone, K. (2013). *Guide to Malware Incident Prevention and Handling for Desktops and Laptops* (NIST SP 800-83r1; p. NIST SP 800-83r1). National Institute of Standards and Technology. https://doi.org/10.6028/NIST.SP.800-83r1

Vinayakumar, R., Alazab, M., Soman, K. P., Poornachandran, P., & Venkatraman, S. (2019). Robust Intelligent Malware Detection Using Deep Learning. *IEEE Access*, *7*, 46717–46738. https://doi.org/10.1109/ACCESS.2019.2906934

Xing, X., Jin, X., Elahi, H., Jiang, H., & Wang, G. (2022). A Malware Detection Approach Using Autoencoder in Deep Learning. *IEEE Access*, *10*, 25696–25706. https://doi.org/10.1109/ACCESS.2022.3155695