Malware threat analysis techniques and approaches for IoT applications: a review

Chimeleze Collins Uchenna[1], Norziana Jamil[2], Roslan Ismail[3], Lam Kwok Yan[4], Mohamad Afendee Mohamed[5]
[1,2,3]College of Computing and Informatics, Universiti Tenaga Nasional, Selangor, Malaysia
[4]School of Computer Science and Engineering, Nanyang Technological University, Singapore
[5]Faculty of Informatics & Computing, Universiti Sultan Zainal Abidin, Malaysia

Article Info

Article history:

Received Mar 26, 2020
Revised Jul 12, 2020
Accepted Apr 27, 2021

Keywords:

Data analytics
Internet of things
Malware threat
Mobile threat

ABSTRACT

Internet of things (IoT) is a concept that has been widely used to improve business efficiency and customer's experience. It involves resource constrained devices connecting to each other with a capability of sending data, and some with receiving data at the same time. The IoT environment enhances user experience by giving room to a large number of smart devices to connect and share information. However, with the sophistication of technology has resulted in IoT applications facing with malware threat. Therefore, it becomes highly imperative to give an understanding of existing state-of-the-art techniques developed to address malware threat in IoT applications. In this paper, we studied extensively the adoption of static, dynamic and hybrid malware analyses in proffering solution to the security problems plaguing different IoT applications. The success of the reviewed analysis techniques were observed through case studies from smart homes, smart factories, smart gadgets and IoT application protocols. This study gives a better understanding of the holistic approaches to malware threats in IoT applications and the way forward for strengthening the protection defense in IoT applications.

Corresponding Author:

Norziana Jamil
College of Computing and Informatics
Universiti Tenaga Nasional
Selangor, Malaysia
Email: norziana@uniten.edu.my

1. INTRODUCTION

Over the years, the internet of things (IoT) concept has exhibited great potential for actuating various domains (personal and enterprise environments); with examples and likely applications but not restricted, smart health for cashless and easy admission into major hospitals, smart cites for energy cost and pollution reduction, smart transportation for developing alternative means to solve road traffic issues as well as smart homes whereby energy industries are developing systems for increasing energy preservation and security among others [1], [2].

With the advent of IoT, computing platforms for general purpose that runs on conventional desktops are now been substituted by platforms like tablets and smartphones. High functionality applications that were once restrained for usage on highly efficient desktops and laptops are currently accessible on the existing mobile platforms as their computational power rises. The ripple effects of the growing trend in usage and popularity of the smartphones have been evidenced in several internet applications where products, accessibility and applications have been migrated to the platform for productivity and interoperability enhancement [3], [4]. IoT technologies are being utilized as foundational technologies for the cooperative-intelligent transport system (C-ITS), that is regarded as next generation intelligent transportation system [5]. Hence, IoT has become a bridge that unites the physical and digital world through the inclusion of smart objects that relate with physical ambience with no direct human interference [6]. Highly essential in IoT applications are confidentiality, authentication, access control and integrity through the implementation of accurate security and privacy protocols [7]. Whilst the novel functionality derived from the IoT can be utilized in improving the lives of humans, the possibility of conventional cyber-attacks on IoT system is rife [8]. More so, it has been well established that a lot of IoT gadgets are susceptible to simple intrusion trials such as utilizing weak or at times default passwords. For instance, in spite of the high sensitivity exhibited by mobile platforms and the tendencies for abuse, various security threats not so different from those that are currently affecting their desktop counterparts have begun to emerge. Similarly with the capacity of smartphones to have several sensors and connection with a lot of IoT gadgets, it becomes a main target for malwares [9].

Furthermore, their nature to host other integrated components such as microphones, inbuilt cameras, accelerometer etc. makes them prone to hijack by malware as well as eavesdrop on their enclosure [10]. Recently, significant numbers of malware including worms, viruses, Trojan horses and rookits have begun to emerge which are targets at exfiltrating confidential data in mobile platform or leverages on the compromised asset so as to access confidential networks through which the gadget has genuine access [11], [12]. Atamli and Martin [13] has identified the three primary groups of malicious entities threatening IoT as: (1) external attackers, (2) malicious users and (3) bad manufacturers [13]. For the purpose of assisting academics and developers to easily comprehend the insight of several kinds of IoT security attacks as well as to create relevant security measures in their IoT developments, Nawir *et al.* [14] designed a well-organized taxonomy as presented in Figure 1 that outlines eight different categories under which attackers can attack the IoT systems.

Meanwhile, traditional computers bring a lot of attacks in IoT environment and uses these computers to infect other connected devices in IoT environment. Having seen these trends, IoT applications are imminently a new area of security research. In this paper, we comprehensively explored the analysis techniques and approaches of current IoT applications regarding series of threats from malware. Several points of interest that an attacker can manipulate either by gaining access to unauthorized information or by causing a denial of service have been identified alongside with the appropriate security architecture to prevent the occurrence.

There are at least 2 research questions that need to be addressed: (1) What are the techniques that malware threats in IoT applications can be efficiently addressed or mitigated? (2) What are the attributes to be considered for malware analytic? This paper gives detailed information on recent challenges with malware threats in IoT applications and highlights existing security techniques for the evaluation of existing vulnerabilities in IoT applications. This paper is organized as follows: section 2 highlights challenges with malware threats in IoT applications, section 3 presents the security techniques in place for malware detection and evaluation and the conclusion and future work is given in section 4.

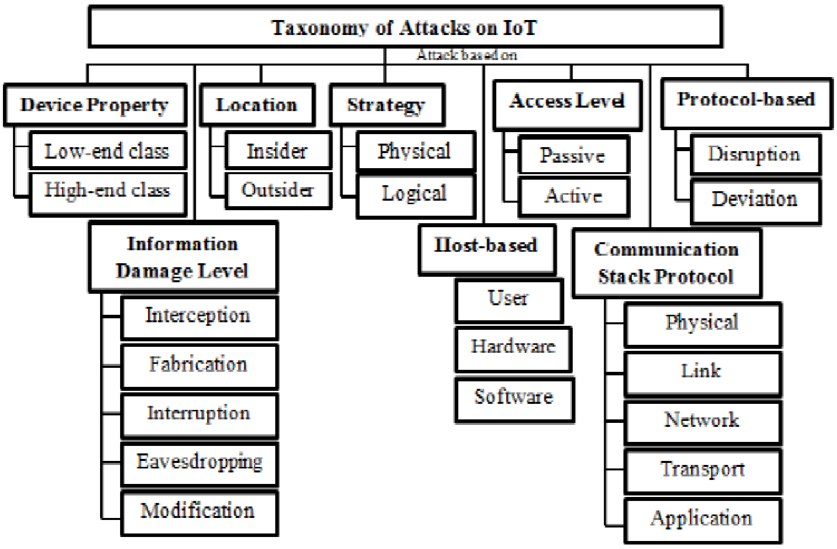

Figure 1. Taxonomy of security attacks on IoT [12]

2. CATEGORIES OF MALWARE THREATS IN IOT DEVICES AND APPLICATIONS

Security threats information technology (IT) have increased and advanced, and those developing these threats are becoming highly ingenious in concealing their efforts. Hence, it has become essential to explore the various means of protecting enabled services on the internet i.e. IoT. Due to this reason, it is important to give a summary of malware threats in IoT platforms and the possible vulnerabilities. Hence, malware threats in software, malware threats in hardware, malware threats in database and malware threats in network services are reflected in this section.

2.1. Malware threats in software

Software patching is an important process to update a particular software with remedy to vulnerabilities. Patching a software in IoT devices seems to be challenging because the device of different limited capabilities. Commonly cloud-based approach is used to patch software in IoT environment and has shown great potential. However, with a high possibility of the cloud imposing security threats such as data breaches, human error, malicious insiders, account hijacking, and DDoS attacks, it becomes essential for researchers to keep exploring software patches update for IoT devices.

2.2. Malware threats in hardware

IoT devices are exposed to the public space due to the clear structure, thus creating for the system free threat sources in several aspects including the exposure of the device ID or serial number of the IoT devices. Meanwhile, most of the IoT devices are connected through cloud infrastructure [2] that poses serious security concerns. The possibility of an attacker to compromise the cloud through the loading of a single malware to multiple IoT devices instantaneously is very high. Data transfer becomes a serious security issue if important or confidential data is shifted in an open wireless network in a random public Wi-Fi network. More so, multiple instance of hacks and data breach have exposed password of the user of popular websites and companies.

2.3. Malware threats in database

In IoT environment, data sent by IoT nodes are usually passed through an IoT gateway before they are sent to the database that resides at cloud. This database, without proper sanitizing, will open doors to hackers to exploit it through SQL injection attacks or other web application attacks, that yields impersonation or false command control.

2.4. Malware threats in network services

The network services is also one of the platforms through which IoT devices can be attacked. The failure of sophisticated encryption algorithms execution on IoT systems promotes their vulnerability with respect to information disclosure attacks. Failure to detect normal data traffic might leave a system exposed to malware infection through distributed denial of service (DDoS) attacks [2]. Owing to resource constraints including computational power and data storage capacity, IoT devices are least required to carry out payload verification as well as integrity check that promotes insecurity in the IoT device.

3. EXISTING SECURITY TECHNIQUES

3.1. Malware detection techniques in software

With the threat of malicious software becoming an essential factor in securing smartphones, Zhao *et al.* [15] designed and implemented an SVM active learning algorithm based Android malware detection architecture called AntiMalDroid with the capacity for the detection and restriction of several common and few novel malwares running on the Android platform. The framework of the AntiMalDroid as shown in Figure 2 consists of two major parts: (i) Learning component which includes the characteristic monitoring module, characteristic learning module, behavior characteristics signature module and signature database, and (ii) Malware detection which includes the run-time behavior monitoring module, behavior signature module, decision module ans the response module. From the results obtained, the performance evaluation i.e. time consumption and battery consumption overhead was a little more but bearable.

The development and execution of a web-based software for detecting and classifying malware was highlighted by Dogru and Kiraz [16]. The system developed depends on client-server structure, static evaluation and web-scraping techniques. Static analysis technique was employed to analyze Android applications. For the purpose of developing a benign application dataset, Android apps of formal institutes (in Turkey and other countries), and common apps on the Google Play market were downloaded through the utilization of the APKPure web page. Malicious software dataset was retrieved from the developed dataset in the Drebin that has been distributed as an open resource.

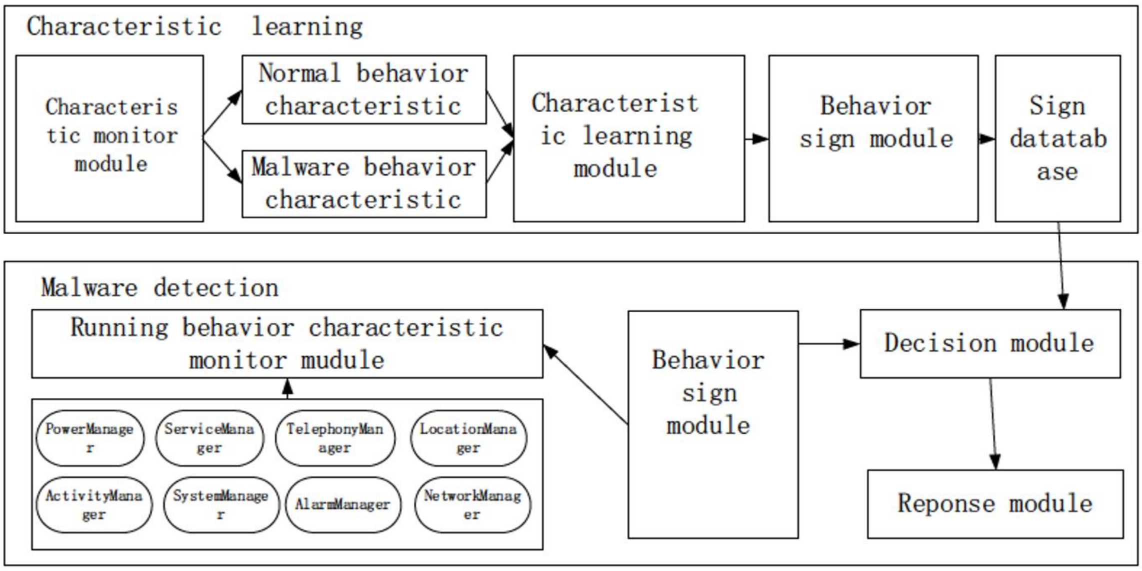

Figure 2. Overview of AntiMalDroid malware detection architecture [15]

The robust nature of the system developed was analyzed by employing 5545 and 1173 Android apps that are malicious and benign respectively. Talha *et al.* [17] developed a permission-based malware detection system, APK auditor which utilizes static analysis for characterization and classification of Android applications as benign or malicious. A new approach to assess potential maliciousness of apps by estimating a statistical score through the requested permissions and uses a central server for the application analysis while the results are retrieved by a web service was proposed. Overall, the APK auditor is made up of three components: (1) a signature database to store extracted information about applications and analyse results, (2) an Android client which is used by end-users to grant application analysis requests, and (3) a central server with the capacity to communicate with both signature database and smartphone client as well as managing the overall analysis process.

A StaDynA andriod malware detection system was proposed by Zhauniarovich *et al*. [18] to address the problem of dynamic code updates in the security analysis of Android applications. The architecture of StaDynA presented comprises two logical components: a server and a client. The static analysis of an application is performed on the server. The client part of StaDynA is a modified Android operating system, hosted either on a real device or an emulator. The client runs the application whenever the dynamic analysis is required. D'Oraziob *et al.* [19] highlighted the potential for pairing mode in iOS devices (which gives room for establishment of a trusted relationship between iOS device and private computer) and exploitation for covert data exfiltration. A data exfiltration model with the capacity to scan iOS devices for vulnerabilities against data exfiltration was developed to exploit the trusted relationship between a private computer and iOS device to collect and transmit user data from victim device to an attacker.

Razak *et al.* [20] proposed bio-inspired Android malware detection system capable of examining new variants of known malware and also to detect the existence of dangerous permissions observed in mobile device applications. The Android malware detection system has three phases as shown in Figure 3 including data collection, machine learning and database. Data collection starts with gathering all permissions including benign and malware applications. The process includes decompiling .apk file, extracting and filtering the permission. The machine learning phase ensures mobile device users can optimize the permission features by employing features optimization approach. Navarro *et al*. [21] and Yang *et al.* [22] respectively combined leveraging ontologies and hybrid analysis with machine-learning techniques for Android malware detection and analysis. The former investigation employed the manifest XML files as the information source and a complex ecosystem of a commercial Android smartphone provided the benign applications downloaded from official apps stores and applications known as malware were obtained from security research repositories. Lastly, ontology queries were used to build the model and machine-learning algorithm (forest-based method) was used to process the original model. For the latter investigation, hybrid method was firstly used for extracting characteristics of software followed by design of a two-stage detection method based on machine learning to achieve the multi-label detection of malware. Random forest-based multi-classifier was employed to determine the family to which the malware belongs.

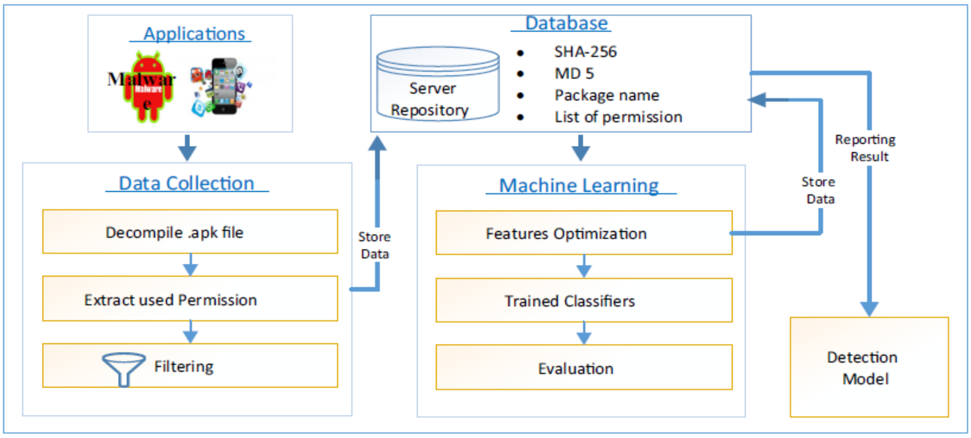

Figure 3. Bio-inspired malware detection architecture [20]

3.2. Malware detection techniques in hardware

Akatyev and James [23] developed a model of a near-future user-centric IoT (UCIoT) network data flow built on STRIDE and DREAD models capable of identifying high-level threats in smart home system and concentration areas for investigators. From the threat assessment made, it was observed that the threats to personal data posed great risks and the digital attacks in the developed have the potential to degenerate into physical consequences such as death. A scheme for minimizing security susceptibilities and threats in IoT devices as well as improving the security of the IoT service environment was proposed by Choi *et al.* [24]. With the implementation of the proposed scheme, the entire security of IoT apps and devices, especially in smart factory can be put into check through system hardening and security monitoring.

A technique that exploits virtual environments and agent-based simulation for evaluating cybersecurity solutions for the future of IoT applications in practical strategies was proposed by Furfaro *et al.* [25]. Most importantly, the integrated utilization of the newly designed virtual environments has the capacity for the exploitation of cutting-edge hardware virtualization technologies and cloud computing, simulation that is agent-based as well as actual gadgets that give room for development and evaluation, in a regulated manner, IoT technologies (applications, protocols, device prototypes) and relevant security threats before they are released in production. The efficiency of the technique was showcased through the consideration of a case study with regards to a regular smart home that involves the combination of real and virtual smart gadgets within a virtualized scenario that initially evaluates security challenges and are being handled thereafter.

In their investigation on malware threats targeted at gadgets deployed in industrial mobile-IoT networks and the complementing detection methods, Sharmeen *et al.* [26] systematically compared static, dynamic, and hybrid analyses by relying on data set, feature extraction and selection methods, detection techniques as well as the efficiency of these techniques. Suspicious API and system calls, as well as the permissions which were extracted and selected as features to detect mobile malware were identified during the investigation. The outcome of the investigation therefore offers great assistance to application developers in securing the use of APIs during the development of applications for industrial IoT network. A concept of dynamic permutation that handles both hardware Trojan and side-channel analysis attacks in emerging IoT applications was proposed by Dofe *et al.* [27]. The implementation of this technique creates enormous difficulty for attackers to launch a hardware attack successfully. More so, the dynamic nature of the permutation technique further hinders Trojan attack, replaces power profile with time and creates difficulty in retrieving the crypto key that relies on the power analysis.

Xiao *et al.* [28] investigated a cloud-based malware detection game where gadgets offload their application traces to security services through base stations/access locations in dynamic networks. The malware detection system was designed with Q-learning in order for a mobile gadget to obtain the optimum rate for offloading without identifying the trace generation and the ratio bandwidth model of the mobile gadgets. The study above was enhanced further in another study by the same set of authors [29] with the use of hotbooting-Q techniques in designing the mobile malware detection system which makes the quality values that rely on the malware detection experience. The deep Q-network method having a deep convolutional neural network was utilized to further enhance the detection speed, the detection accuracy and the utility. Patil *et al.* [30] developed in-VM-assisted lightweight agent-based malware detection (AMD) architecture for securing high-risk virtual machine (VM) from malware at the initial stage of VM life cycle.

The malware detection architecture has two parts, which are agent at VM and anomaly detection at hypervisor for detecting both known and unknown malware. Figure 4 presents the design of the AMD architecture.

Mishra *et al.* [31] proposed a dynamic evaluation-based introspection technique, named KVMInspector for malware detection in KVM-based cloud environment. Libraries of LibVMI and Nitro were utilized in extracting the reduced level information of a running virtual machine by checking its memory, trapping hardware events, as well as evaluating the vCPU registers from KVM. X. Jia *et al*. [32] proposed FindEvasion, a cloud based technique for the detection of environment-sensitive malware. It has capacity to extract the suspected program from the VM and evaluates them on multiple operating environments. It is also capable of multiple behavioral sequences similarity (MBSS) check algorithm, that relates the behaviors of a suspected program noticed in multiple operating environments, and ascertains the suspected program is an environment-sensitive malware or not. Kumara and Jaidhar [33] proposed a hypervisor oriented automated internal-external (A-IntExt) malware detection model. It employs a protected and lightweight in-VM-assisted component to gather internally state information. It possesses an intelligent cross view analyzer (ICVA) at hypervisor that regularly checks the supplied data by the in-VM component to detect hidden, dead and malicious processes.

3.3. Malware detection techniques in database

By utilizing the iOS devices as a case study, D'Orazio *et al.* [34] presented for the iOS devices, the capacity for pairing mode (which allows institution of a genuine relationship between iOS device and private computer) and the usage for exfiltration of hidden data. A data exfiltration model with the capacity to subject iOS gadgets to scanning for susceptibilities against data exfiltration was developed for the exploitation of the genuine relationship between a private computer and iOS gadget for collection and transmission of user data from victim gadget to an intruder.

In their investigation, Chen *et al.* [35] proposed unbalanced classification methods consisting of synthetic minority oversampling technique (SMOTE) + support vector machine (SVM), SVM cost-sensitive (SVMCS), and C4.5 cost sensitive (C4.5CS) methods for machine-learning based mobile malware detection using imbalanced network traffic. An imbalanced data gravitation-based classification (IDGC) algorithm was deployed to classify imbalanced data when they approach certain threshold where the behavior of the algorithm for classification is significantly degraded. A simplex imbalanced data gravitation classification (S-IDGC) model was developed to decrease the time cost of IDGC without affecting the behavior of the classification process and a machine-learning based correlative standard prototype system was proposed for users to detect the performance of various classification algorithms on similar dataset. The architecture of the prototype system is shown in Figure 5 and can be grouped into these units: (1) unified network traffic data (2) traffic processing and classifier settings, (3) comparative performance results, and (4) operating procedures of the system.

An efficient malware detection technique in cloud infrastructure employing CNN (convolutional neural network) was proposed by Abdelsalam *et al*. [36]. The accuracy of the CNN classifier was enhanced by employing a new 3d CNN (wherein the input is an assembly of samples over a period of time) that largely assists in reducing the mistakenly labelled samples in the course of data collection and training. Elsewhere, Silakari and Chourasia [37] proposed the accelerated chaotic map particle swarm optimization (ACMPSO K-means) technique which combines PSO and K-means to give satisfactory result for the detection of malware in cloud computing infrastructure.

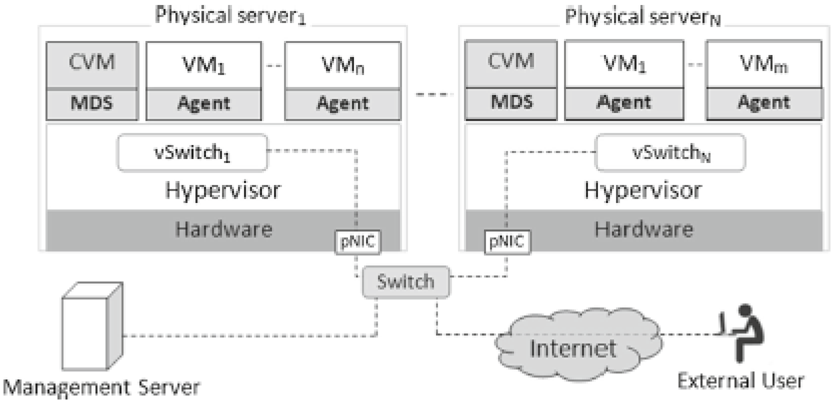

Figure 4. The design of the AMD architecture [30]

Sun *et al.* [38] developed CloudEyes; a cloud-based malware detection system that offers effective and trusted security services in resource constrained devices. CloudEyes offers questionable bucket cross-filtering, a new signature detection system reliant on the reversible sketch structure that offers hindsight and efficient positions for fragments of malicious signature. In a complementary study [39], another cloud-based detection system that gives protection to data privacy of both the cloud server and the client. PriMal employs a recently developed Private Malware Signature Set Intersection (PMSSI) protocol to activate the cloud server and client for the achievement of malware confirmation without exposing the data privacy in semi-honest model. Figure 6 shows the system architecture for PriMal. Q. K. Ali Mirza *et al.* [40] proposed CloudIntell; an intelligent machine learning method for the enhancement of malware detection rate and a cloud-based framework for supporting and hosting the implementation of the methodology. An automated feature tool was developed for extraction, that obtained features from over 200,000 files in an efficient manner.

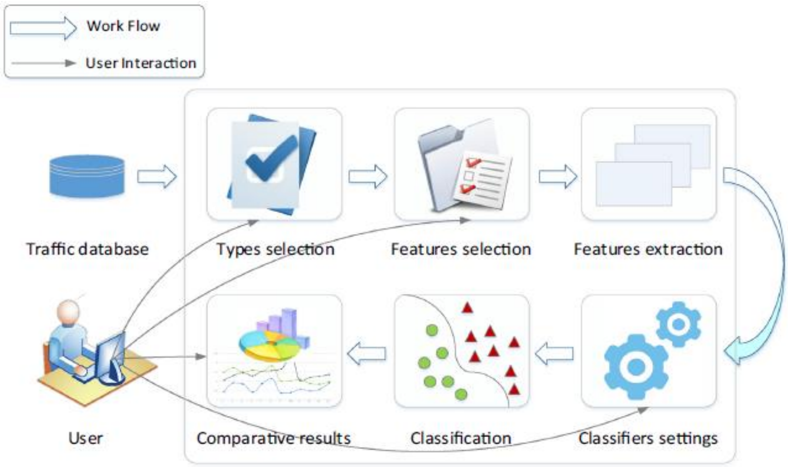

Figure 5. A relative standard prototype system architecture for network traffic-based malware detection [35]

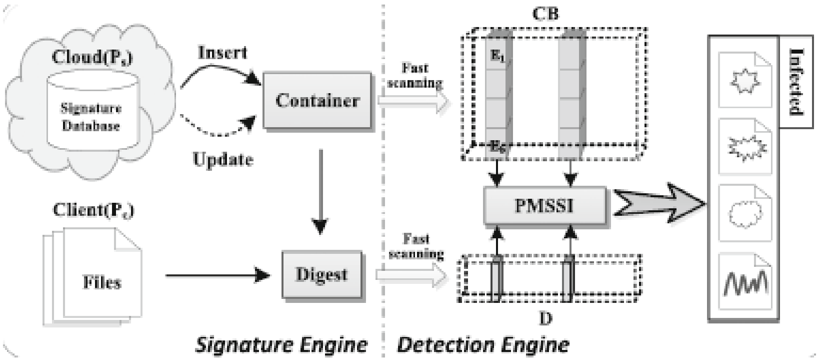

Figure 6. System architecture for PriMal [39]

3.4. Malware detection techniques in network services

Having identified the vulnerabilities of IoT devices and applications to sensor-based threats owing to the absence of decent security measures available for controlling the usage of sensors by apps, Sikder *et al*. [41] conducted a comprehensive survey on the existing countermeasures developed specially for sensors security in IoT devices and gave feasible recommendations for future research exploration. The existing security mechanisms for prevention of sensor-based threats were categorized into two entities. For the first category i.e. the enhancement of existing sensor management systems, the developed systems include: (i) Semadroid-an Android sensor management system which offers users a monitoring and logging feature that makes the utilization of sensors by app explicit, (ii) Aware-an authorization architecture for android that extends Android Middleware for controlling access to privacy-sensitive sensors and with the capacity of at most 7% of the users being tricked by examples of four kinds of attack, contrary to an average of 85% for previous approaches, and (iii) 6thSense-a comprehensive context-aware architecture with approx. 97% accuracy and F-score, that employs the entire sensor data in real time and decides if the current context of the sensors or not utilizing several machine learning-based techniques.

For the second category i.e. protection of sensed data, the developed mechanisms include: (i) location-privacy preserving mechanisms (LPPMs)-a mechanism to limit the probability of success of inference attack, on location data and offers a robust defense against white-box attacks when integrated with targeted maneuvers (reduction of probability success of white-box attacks to 3%), (ii) single inverter ring oscillator (SIRO)-a countermeasure to immune IoT devices and applications from power analysis and electromagnetic emanation attacks, and (iii) AuDroid-a model for securing communications through audio channels whenever applications utilize the device’s microphones and speakers.

Owing to the diverse existence of malicious software (malcode/malware) which poses great issues for network and end host security, Gupta *et al*. [42] developed a new graph pruning system for establishing the inheritance relationships between several instances of malcode that relies on temporary information and major general phrases detected in the malcode descriptions. Comprehensive investigation revealed the identification of 669 distinct malware families by algorithm which can be of great use to domain experts and can assist in the design and development of proactive strategies to prevent malware attacks.

Being one of the most adopted IoT application protocols, Firdous *et al*. [43] presented the message quelling telemetry protocol (MQTT) threat model as shown in Figure 7 and conducted an analysis on the denial of service (DoS) attack which targets MQTT brokers. The investigators setup a testbed using virtual machines for the testing of an MQTT broker server performance in the course of a DoS attack. A simulation utilizing 2000 PUBLISH messages (4MB payload each) were transferred to the broker which triggered the crash of the broker in 30 s. In the course of the attack, CPU load rise steadily and the memory was elevated to 100% prior the crash of the MQTT, while the network traffic reached 100MB/s in the course of the payload flooding attack.

Alhawi *et al*. [44] leveraged on different machine learning methods for Windows ransomware network traffic detection. NetConverse was introduced out of the methods where in the data collection phase of their experiment, samples of network traffic were collected for the ransomeware and benign Windows apps while the feature extraction phase retrieves the necessary features and attaches them for the creation of the utilized dataset. Lastly, in the machine learning classifier phase, training and testing of numerous algorithms located in Waikato Envitonment for knowledge Analysis 3.8.1 (WEKA) tool for machine learning to find the optimal detection model. Messabi *et al*. [45] presented a novel approach using Python for the detection of DNS malware through the use of various behavioral features for the recognition of malicious domains from the legitimate ones before they are opened by the user. The approach employs the collection of most common DNS-based feature utilized in past investigations so as to obtain the optimal results.

In order to develop a malware detection model for cloud environment, Yadav [46] proposed a new consolidated WFCM-AANN (weighted fuzzy c-means clustering algorithm with auto associative neural network). The proposed system consists of two modules. On the first hand, the clustering module is employed to gather the input dataset into clusters with the use of the WFCM clustering. On the other hand, the centroidal part from the clustered dataset is subjected to the periodic AANN which is utilized in characterizing intrusion state of the information.

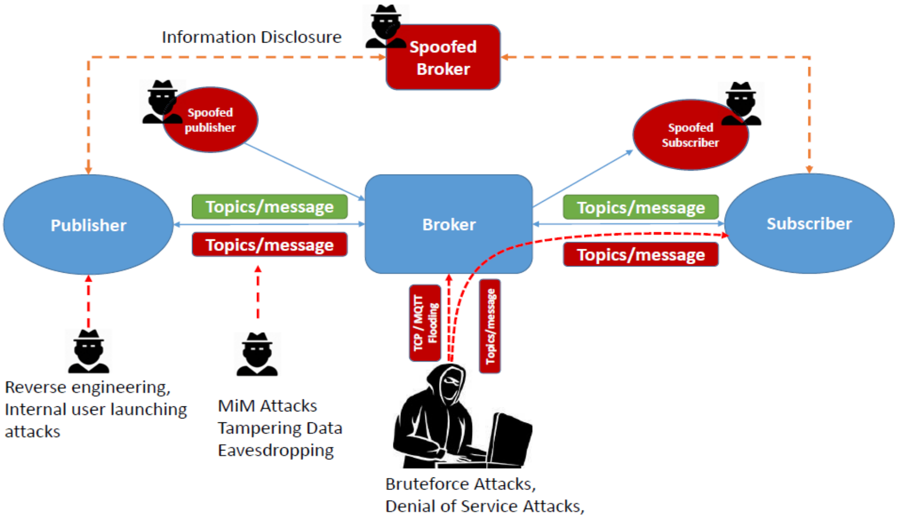

Figure 7. MQTT threat model [43]

By relying on text semantics of network flows, Wang *et al*. [47] proposed a framework for malware detection that can endure encrypted HTTPS and non-encrypted HTTP traffic in home networks, bring-your-own-device (BYOD) enterprise networks, and 3G/4G mobile networks. The proposed technique handles all HTTP flow as a document, and thereafter utilizes the word segmentation reliant on N-gram generation for generation candidate features for effective characterization of a certain HTTP flow. Finally, an SVM classifier is trained to automatically detect if the unknown traffic is malicious or benign.

For the purpose of ensuring information security, Kang *et al*. [48] classified malware into families by employing the word2vec model and the long short-term memory (LSTM). The word2vec was used to extract names of opcodes as well as API function from the assembly source and later vectorized into vectors having smaller value of dimensions for the reduction of learning time and improve the classification rate. The LSTM was then used to obtain the classification results by receiving the vectorized results. Prasse *et al*. [49] developed and investigated a model for malware detection that relies on LSTMs which utilizes just the observable areas of the HTTPS traffic. A VPN client was deployed to numerous client computers for observing the relationships between executable files and network flows on numerous client computers. Finally, anti-virus devices were used to obtain in retrospect, which of the network flows in the training and evaluation dataset emanate from malware. Malik and Kaushal [50] proposed CREDROID which detects malicious apps by relying on their DNS queries and the data it transfers to remote server by carrying out the comprehensive evaluation of network traffic logs in offline mode. The technique is semi-automated and works on several considerations including the remote server that connects the application, data being transfered as well as the protocol that is utilized in communicating for the identification of the credibility of the application.

4. FUTURE APPROACHES FOR SECURITY ENHANCEMENT IN IoT APPLICATIONS

Sikder *et al*. [41] proposed the open issues listed as follows as future directions in the context of sensor-based threats: (i) investigation of anticipated functionality for threats identification, (ii) standard security mechanisms adoption, (iii) Fine-grained control of the sensors, (iv) control data distribution between sensors, (v) protection of sensor data when at rest, (vi) leakage prevention of confidential data, (vii) integrity protection of sensor operations, and (viii) adoption of intrusion systems for detecting attacks. Upon conducting a state-of-the-art survey on security attacks in IoT, J. Deogirikar and A. Vidhate [51] proposed a need for refinement in the current network architecture as well as the creation of a novel network architecture that is lightweight for future work. With this in place at the security layers in each network layer, there is high possibility of solving the performance and security related issues in IoT applications.

Focusing on the smart devices and multimedia applications which provides remote monitoring of our daily activities. Shifa *et al.* [52] proposed for future investigation a lightweight encryption to preserve the privacy of multimedia data within IoT environment. The proposed system ensures the security of organizations and individuals by tackling the various security levels needed by multimedia applications at various stages in the operation. For the total implementation of the system for different device capabilities, the investigators emphasized on the performance and security evaluation by considering several potential attacks against the encryption system the validation for the efficiency of the implemented multi-level partial encryption techniques. In addition, future investigations must also focus on threat and investigation of retail IoT devices through the development of systems for statistics measurement of malware threats as well as the continuous monitoring and analysis of new malware threats in IoT devices. Table 1 in appendix presents the performance and security characteristics of analysis techniques for malware detection in different platforms.

5. CONCLUSION

In this study, we have documented comprehensive analysis techniques with cutting-edge solutions to address malware threat in IoT applications. In particular, static, dynamic as well as hybrid analyses have been adopted by researchers to confront security issues plaguing several IoT applications. The effectiveness of the documented analysis techniques was demonstrated using case studies including smart home systems, smart factories, smart gadgets and IoT application protocols. Systems such as 6thSense, a comprehensive context-aware architecture for sensors security in IoT devices offers approximately 97% accuracy and F-score. Similarly the location-privacy preserving mechanisms (LPPMs) with integrated targeted maneuvers offers a robust defense against white-box attacks by reducing the probability success of white-box attacks to 3%. It was discovered from all the techniques reviewed that the investigation of anticipated functionality for sensor-based threats and investigation of a lightweight encryption to preserve the privacy of multimedia data within IoT environment as key areas that must be worked on in future investigations.

APPENDIX

Table 1. Performance and security characteristics of analysis techniques for malware detection in different platforms

Existing approach to malware threat		Performance Char.			Security Char.				
Technique used	Cate-gory	Computer power	Memory Footprint	Energy Consume	Integrity	Data Protection	General Char.	Limitations	Ref.
An SVM active learning algorithm based Android malware detection architecture called AntiMalDroid with the capacity for the detection and restriction of several common and few novel malwares running on the Android platform	Sf	Y	Y	Y	Y	N	Efficient; Reliable; Flexible;	Extension to other Linux-based embedded systems is necessary for the validation of the technique	[15]
Web-based Android malicious software detection and classification system based on client-server architecture, static analysis and web-scraping methods	Sf	Y	N	N	Y	Y	Reliable; Efficient	Accuracy can be further increased	[16]
APK Auditor Android client APK; APK Auditor Signature Database Applications; APK Auditor central server; APK Auditor administration portal	Sf	N	Y	N	Y	Y	External resource dependency	External system dependencies make the whole system unavailable when these dependencies are out of service.	[17]
StaDynA andriod malware detection solution based on combination of static and dynamic analysis of applications	Sf	N	Y	N	Y	Y	Flexible	Android emulator is slow	[18]
A data exfiltration model with the capacity to scan iOS devices for vulnerabilities against data exfiltration	Sf	Y	N	Y	Y	N	Reliable	Not supported by other IoT devices and big data systems	[19]
A bio-inspired Android malware detection system capable of examining new variants of known malware and also to detect the existence of dangerous permissions observed in mobile device applications	Sf	Y	N	N	Y	N	Adaptable; Efficient	The limitation observed in the study lies in the ability of the system to detect Android malware in the cloud	[20]
A system comprising of leveraging ontologies and machine-learning techniques for malware analysis into Android permissions ecosystem.	Sf	Y	N	Y	Y	N	Efficient; Reliable	The inclusion into the graph and use of the already trained classifier could be trivial.	[21]
A two-stage detection method based on machine learning to achieve the multi-label detection of malware.	Sf	Y	N	Y	Y	N	Efficient; Reliable	Accuracy can be further increased	[22]
A near-future User-Centric IoT (UCIoT) network data flow built on STRIDE and DREAD models capable of identifying high-level threats in Smart Home System and concentration areas for investigators	Hd	Y	N	N	Y	Y	Reliable; Adaptable	The proposed model for threat assessment is yet to be tested on industrial IoT systems	[23]
A scheme for minimizing security susceptibilities and threats in IoT devices as well as improving the security of the IoT service environment	Hd	Y	N	N	Y	Y	Adaptable; Flexible	There is a need to reduce the size and optimize binaries used in the implementation of the technique	[24]
A technique that exploits virtual environments and agent-based simulation for evaluating cybersecurity solutions for the future of IoT applications in practical strategies	Hd	Y	N	N	Y	N	Efficient	Current findings revealed the necessity for a system upgrade in future investigations.	[25]
Comparison study of static, dynamic, and hybrid analyses by relying on data set, feature extraction and selection methods, detection techniques as well as the efficiency of these techniques based on client-server	Hd	N	N	Y	Y	N	Reliable	Accuracy can be further increased	[26]

architecture, static analysis and web-scraping methods									
A concept of dynamic permutation that handles both hardware Trojan and side-channel analysis attacks in emerging IoT applications	Hd	N	Y	Y	N	Y	Efficient; Reliable	Higher computational power is required	[27]
Cloud-based mobile malware detection system designed with Q-learning	Hd	Y	N	Y	Y	N	Efficient; Reliable	The Q-learning has a slow learning rate	[28]
Cloud-based mobile malware detection system designed withhotbooting-Q techniques	Hd	Y	Y	Y	Y	N	Efficient; Reliable	More dataset is required to ascertain the prospects of this method	[29]
Agent-based malware detection (AMD) architecture for securing high-risk virtual machine (VM) from malware at the initial stage of VM life cycle.	Hd	Y	N	Y	Y	N	Efficient; Reliable; Flexible	It needs to be extended for the detection of certain encrypted malware such as Ransomeware, that are challenging to detect without running them	[30]
A dynamic evaluation based introspection technique, named KVMInspector for malware detection in KVM-based cloud environment	Hd	Y	Y	Y	Y	N	Efficient; Reliable	There is need to integrate network monitoring functionalities with the KVMInspector in order for the program and network behaviour to be evaluated	[31]
FindEvasion, a cloud based technique for the detection of environment-sensitive malware	Hd	Y	N	N	Y	N	Efficient; Reliable	It consumes too much time and may be not be feasible for large-scale computing systems.	[32]
A hypervisor oriented Automated Internal–External (A-IntExt) malware detection model.	Hd	Y	N	Y	Y	N	Efficient; Adaptable	It is quite challenging to secure the ICVA is a challenge.	[33]
A data exfiltration model with the capacity to scan iOS devices for vulnerabilities against data exfiltration	Db	Y	N	Y	Y	Y	Reliable	Not supported by other IoT gadgets and systems having large data	[34]
Machine Learning based approach for malware detection for Android platform using Rain Forest (RF), Support Vector Machine (SVM), k-nearest neighbour (KNN), Naive Bayes (NB)	Db	N	Y	N	N	Y	Flexible; Adaptable;	Samples primarily emanate from the academic sample category of the institution or platform and have no metamorphic malware sample. Metamorphic malware has the capacity to avoid the suggested technique. It is obviously susceptible to obfuscation and packing	[35]
Malware detection technique in cloud infrastructure that employs CNN	Db	Y	N	N	Y	Y	Efficient; Reliable; Flexible	There is need to extend the scale of experiment in future study by employing more malware binaries	[36]
ACMPSO K-means for detection of malware in cloud computing infrastructure	Db	Y	N	N	Y	N	Reliable; Flexible	Investigation of more standard dataset is necessary before the full implementation of the approach	[37]
CloudEyes; a cloud-based detection system that offers effective and trusted security services in resource constrained devices	Db	Y	N	Y	Y	Y	Reliable	The detection needs further improvement by using better algorithms	[38]
PriMal; a cloud-based detection system that gives protection to data privacy of both the cloud server and the client	Db	Y	Y	N	Y	Y	Reliable	The detection needs further improvement by using better algorithms	[39]
CloudIntell; an intelligent malware detection system for the enhancement of malware detection rate	Db	Y	N	N	Y	N	Efficient; Reliable	The memory footprint needs to be improved	[40]
Countermeasures developed specially for sensors security in IoT devices	Ns	Y	N	N	Y	Y	Efficient; Adaptable	A total and extensive solution for autonomous policy enforcement, detailed coverage of the entire sensors, as well as an accurate trade-off between power and performance are yet to be developed	[41]
A new graph pruning system for	Ns	Y	N	N	Y	N	Adaptable;	Corpus of malware meta-	[42]

establishing the inheritance relationships between several instances of malcode that relies on temporary information and major general phrases detected in the malcode descriptions							Reliable	data must be expanded for detailing malware evolutionary attributes and collaboration with AV companies is essential for the development of systems with capacity to anticipate future trends in malware evolution	
Message Quelling Telemetry Protocol (MQTT) threat model with analysis on the Denial of Service (DoS) attack which targets MQTT brokers	Ns	N	Y	N	Y	N	Satisfactory	The method is restricted to quantify only the impact of DoS attacks	[43]

NetConverse; a machine learning technique for Windows ransomeware network traffic detection	Ns	Y	N	Y	Y	Y	Efficient;
							Reliable;
							Flexible
								The memory footprint requires improvement	[44]

A novel approach using Python for the detection of DNS malware	Ns	N	N	Y	Y	N	Reliable	The approach is not complex enough for the detection of all malware domains on the internet	[45]
A new consolidated WFCM-AANN for malware detection in cloud environment	Ns	Y	N	Y	Y	N	Efficient; Reliable	Only two normal and anomaly detection problems were investigated. Hence, detection of multiple types of attacks needs to be explored in future work.	[46]

A framework for malware detection that relies on text semantics of network flows	Ns	Y	N	Y	Y	N	Efficient;
							Relibale;
							Flexible
								The technique can not trigger all malicious behaviors	[47]

Utilization of word2vec model and Long Short-Term Memory (LSTM) for malware classification into families.	Ns	N	Y	Y	Y	Y	Efficient; Reliable	The technique requires high computing resources	[48]

LSTM based malware detection model that utilizes just the observable areas of HTTPS traffic	Ns	Y	N	Y	Y	N	Efficient;
							Reliable;
							Adaptable
								Further study is necessary with large dataset	[49]

CREDROID; a malware detection system that rely on DNS queries and the data it transfers to remote server through evaluation of network traffic logs in offline mode.	Ns	N	N	Y	Y	N	Reliable	It is almost impossible to map out the sent messages to the premium numbers by the malware	[50]

**Sf: software, Hd: hardware, Db: Database, Ns: Network services, Y: Yes, N: No*

REFERENCES

[1] L. Atzori, A. Iera, and G. Morabito, “The internet of things: A survey,” *Computer networks*, vol. 54, no. 15, pp. 2787-2805, 2010, doi: 10.1016/j.comnet.2010.05.010.

[2] J. Ahamed and A. V. Rajan, "Internet of Things (IoT): Application systems and security vulnerabilities," *2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA)*, Ras Al Khaimah, United Arab Emirates, 2016, pp. 1-5, doi: 10.1109/ICEDSA.2016.7818534.

[3] S. A. Akinboro, A. Omotosho, and M. Odusami, “An improved model for securing ambient home network against spoofing attack,” *International Journal Computer Network and Information Security*, vol. 10, no. 2, pp. 20-26, 2018, doi: 10.5815/ijcnis.2018.02.03.

[4] M. Odusami, *et al.*, “Android Malware Detection: A Survey,” in *International Conference on Applied Informatics, ICAI 2018*, Bogota, Colombia, 2018, pp. 255-266, doi: 10.1007/978-3-030-01535-0_19.

[5] J. Alfonso, N. Sanchez, J. M. Menendez, and E. Cacheiro, “Cooperative ITS communications architecture: the FOTsis project approach and beyond,” *IET Intelligent Transport Systems*, vol. 9, no. 6, pp. 591-598, 2015, doi: 10.1049/iet-its.2014.0205.

[6] E. Fleisch, “What is the internet of things? An economic perspective,” *Economics, Management, and Financial Markets*, vol. 5, no. 2, pp. 125-157, 2010.

[7] M. A. Razzaq, S. Habib, M. Ali, and S. Ullah, “Security issues in the Internet of Things (IoT): A comprehensive study,” *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 6, p. 383, 2017, doi: 10.14569/IJACSA.2017.080650.

[8] B. Min and V. Varadharajan, "Design and Evaluation of Feature Distributed Malware Attacks against the Internet of Things (IoT)," *2015 20th International Conference on Engineering of Complex Computer Systems (ICECCS)*, Gold Coast, QLD, Australia, 2015, pp. 80-89, doi: 10.1109/ICECCS.2015.19.

[9] S. Jha, S. Katzenbeisser, C. Schallhart, H. Veith and S. Chenney, "Enforcing Semantic Integrity on Untrusted Clients in Networked Virtual Environments," *2007 IEEE Symposium on Security and Privacy (SP '07)*, Berkeley, CA, USA, 2007, pp. 179-186, doi: 10.1109/SP.2007.16.

[10] Jon Howell and Stuart Schechter, “What You See is What they Get: Protecting users from unwanted use of microphones, camera, and other sensors,” *in Proceedings of Web 2.0 Security and Privacy Workshop*.

[11] A. P. Felt, M. Finifter, E. Chin, S. Hanna, and D. Wagner, “A survey of mobile malware in the wild,” in *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices*, pp. 3-14, 2011, doi: 10.1145/2046614.2046618.

[12] M. C. Mont, S. Pearson and P. Bramhall, "Towards accountable management of identity and privacy: sticky policies and enforceable tracing services," *14th International Workshop on Database and Expert Systems Applications, 2003. Proceedings.*, Prague, Czech Republic, 2003, pp. 377-382, doi: 10.1109/DEXA.2003.1232051.

[13] A. W. Atamli and A. Martin, "Threat-Based Security Analysis for the Internet of Things," *2014 International Workshop on Secure Internet of Things*, Wroclaw, Poland, 2014, pp. 35-43, doi: 10.1109/SIoT.2014.10.

[14] M. Nawir, A. Amir, N. Yaakob and O. B. Lynn, "Internet of Things (IoT): Taxonomy of security attacks," *2016 3rd International Conference on Electronic Design (ICED)*, Phuket, Thailand, 2016, pp. 321-326, doi: 10.1109/ICED.2016.7804660.

[15] M. Zhao, F. Ge, T. Zhang, and Z. Yuan, “AntiMalDroid: An efficient SVM-based malware detection framework for android,” in *International Conference on Information Computing and Applications*, Springer, Berlin, Heidelberg, 2011, vol. 243, pp. 158-166, doi: 10.1007/978-3-642-27503-6_22.

[16] I. A. Dogru and O. Kiraz, “Web-Based Android Malicious Software Detection and Classification System,” *Applied Sciences*, vol. 8, no. 9, p. 1622, 2018, doi: 10.3390/app8091622.

[17] K. A. Talha, D. I. Alper, and C. Aydin, “APK Auditor: Permission-based Android malware detection system,” *Digital Investigation*, vol. 13, pp. 1-14, 2015, doi: 10.1016/j.diin.2015.01.001.

[18] Y. Zhauniarovich, M. Ahmad, O. Gadyatskaya, B. Crispo, and F. Massacci, “Stadyna: Addressing the problem of dynamic code updates in the security analysis of android applications,” in *Proceedings of the 5th ACM Conference on Data and Application Security and Privacy*, pp. 37-48, 2015, doi: 10.1145/2699026.2699105.

[19] C. J. D’Orazio, K. R. Choo and L. T. Yang, "Data Exfiltration From Internet of Things Devices: iOS Devices as Case Studies," in *IEEE Internet of Things Journal*, vol. 4, no. 2, pp. 524-535, April 2017, doi: 10.1109/JIOT.2016.2569094.

[20] M. F. Ab Razak, N. B. Anuar, F. Othman, A. Firdaus, F. Afifi, and R. Salleh, ”Bio-inspired for features optimization and malware detection,” *Arabian Journal for Science and Engineering*, vol. 43, no. 12, pp. 6963-6979, 2018, doi: 10.1007/s13369-017-2951-y.

[21] L. C. Navarro, A. K. W. Navarro, A. Gregio, A. Rocha, and R. Dahab, “Leveraging ontologies and machine-learning techniques for malware analysis into Android permissions ecosystems,” *Computers & Security*, vol. 78, pp. 429-453, doi: 10.1016/j.cose.2018.07.013.

[22] F. Yang, Y. Zhuang, and J. Wang, “Android Malware Detection Using Hybrid Analysis and Machine Learning Technique,” in *International Conference on Cloud Computing and Security*, vol. 10603, pp. 565-575, 2017, doi: 10.1007/978-3-319-68542-7_48.

[23] N. Akatyev and J. I. James, “Evidence identification in IoT networks based on threat assessment,” *Future Generation Computer Systems*, vol. 93, pp. 814-821, 2019, doi: 10.1016/j.future.2017.10.012.

[24] S. K. Choi, C. H. Yang, and J. Kwak, “System Hardening and Security Monitoring for IoT Devices to Mitigate IoT Security Vulnerabilities and Threats,” *KSII Transactions on Internet & Information Systems*, vol. 12, no. 2, pp. 906-918, 2018, doi: 10.3837/tiis.2018.02.022.

[25] A. Funrfao, L. Argento, A. Parise, and A. Piccolo, “Using virtual environments for the assessment of cybersecurity issues in IoT scenarios,” *Simulation Modelling Practice and Theory*, vol. 73, pp. 43-54, 2017, doi: 10.1016/j.simpat.2016.09.007.

[26] S. Sharmeen, S. Huda, J. H. Abawajy, W. N. Ismail and M. M. Hassan, "Malware Threats and Detection for Industrial Mobile-IoT Networks," in *IEEE Access*, vol. 6, pp. 15941-15957, 2018, doi: 10.1109/ACCESS.2018.2815660.

[27] J. Dofe, J. Frey and Q. Yu, "Hardware security assurance in emerging IoT applications," *2016 IEEE International Symposium on Circuits and Systems (ISCAS)*, Montreal, QC, Canada, 2016, pp. 2050-2053, doi: 10.1109/ISCAS.2016.7538981.

[28] L. Xiao, Y. Li, X. Huang and X. Du, "Cloud-Based Malware Detection Game for Mobile Devices with Offloading," in *IEEE Transactions on Mobile Computing*, vol. 16, no. 10, pp. 2742-2750, 1 Oct. 2017, doi: 10.1109/TMC.2017.2687918..

[29] X. Wan, G. Sheng, Y. Li, L. Xiao and X. Du, "Reinforcement Learning Based Mobile Offloading for Cloud-Based Malware Detection," *GLOBECOM 2017-2017 IEEE Global Communications Conference*, Singapore, 2017, pp. 1-6, doi: 10.1109/GLOCOM.2017.8254503.

[30] R. Patil, H. Dudeja, and C. Modi, “Designing in-VM-assisted lightweight agent-based malware detection framework for securing virtual machines in cloud computing,” *International Journal of Information Security*, vol. 19, no. 2, pp. 147-162, 2020, doi: 10.1007/s10207-019-00447-w.

[31] P. Mishra, I. Verma, and S. Gupta, “KVMInspector: KVM Based introspection approach to detect malware in cloud environment,” *Journal of Information Security and Applications*, vol. 51, p. 102460, 2020, doi: 10.1016/j.jisa.2020.102460.

[32] X. Jia, G. Zhou, Q. Huang, W. Zhang, and D. Tian, “Findevasion: an effective environment-sensitive malware detection system for the cloud,” in *International Conference on Digital Forensics and Cyber Crime*, Springer, Cham, 2017, vol. 216,pp. 3-17, doi: 10.1007/978-3-319-73697-6_1.

[33] M. A. A. Kumara and C. D. Jaidhar, “Leveraging virtual machine introspection with memory forensics to detect and characterize unknown malware using machine learning techniques at hypervisor,” *Digital Investigation*, vol. 23, pp. 99-123, 2017, doi: 10.1016/j.diin.2017.10.004.

[34] F. H. A. Jabar, J. I. Mohammad, A. F. M. Zain, and A. B. Hasan, "Data Exfiltration of Ultrasonic Signal in Computer Security System: A Review,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol 10, no 2, pp. 490-497, May 2018, doi: 10.11591/ijeecs.v10.i2.pp490-497.

[35] Z. Chen, *et al., “*Machine learning based mobile malware detection using highly imbalanced network traffic,” *Information Sciences*, vol. 433-434, pp. 346-364, 2018, doi: 10.1016/j.ins.2017.04.044.

[36] M. Abdelsalam, R. Krishnan, Y. Huang and R. Sandhu, "Malware Detection in Cloud Infrastructures Using Convolutional Neural Networks," *2018 IEEE 11th International Conference on Cloud Computing (CLOUD)*, San Francisco, CA, USA, 2018, pp. 162-169, doi: 10.1109/CLOUD.2018.00028.

[37] Nancy, S. Silkari, and U. Chourasia, “Malware Detection Techniques in Cloud Computing Infrastructure using ACMPSO-k means,” *International Journal of Computer Science and Information Security*, vol. 14, no. 8, p. 29, 2016.

[38] H. Sun, X. Wang, R. Buyya, and J. Su, “CloudEyes: Cloud-based malware detection with reversible sketch for resource-constrained internet of things (IoT) devices,” *Software: Practice and Experience*, vol. 47, no. 3, pp. 421-441, 2016, doi: 10.1002/spe.2420.

[39] H. Sun, J. Su, X. Wang, R. Chen, Y. Liu, and Q. Hu, “Primal: Cloud-based privacy-preserving malware detection,” in *Australasian Conference on Information Security and Privacy*, Springer, Cham, 2017, pp. 153-172, doi: 10.1007/978-3-319-59870-3_9.

[40] Q. K. Ali Mirza, I. Awan, and M. Younas, “CloudIntell: An intelligent malware detection system,” *Future Generation Computer Systems*, vol. 86, pp. 1042-1053, 2018, doi: 10.1016/j.future.2017.07.016.

[41] A. K. Sikder, G. Petracca, H. Aksu, T. Jaeger, and A. S. Uluagac, “A survey on sensor-based threats to internet-of-things (iot) devices and applications,” *arXiv preprint arXiv:1802.02041*, 2018.

[42] A. Gupta, P. Kuppili, A. Akella and P. Barford, "An empirical study of malware evolution," *2009 First International Communication Systems and Networks and Workshops*, Bangalore, India, 2009, pp. 1-10, doi: 10.1109/COMSNETS.2009.4808876.

[43] S. N. Firdous, Z. Baig, C. Valli and A. Ibrahim, "Modelling and Evaluation of Malicious Attacks against the IoT MQTT Protocol," *2017 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData),* Exeter, UK, 2017, pp. 748-755, doi: 10.1109/iThings-GreenCom-CPSCom-SmartData.2017.115.

[44] O. M. K. Alhawi, J. Baldwin, and A. Dehghantanha, “Leveraging machine learning techniques for windows ransomware network traffic detection,” in *Cyber Threat Intelligence*, Springer, Cham, 2018, vol. 70, pp. 93-106, doi: 10.1007/978-3-319-73951-9_5.

[45] K. Al Messabi, M. Aldwairi, A. Al Yousif, A. Thoban, and F. Belqasmi, “Malware detection using dns records and domain name features,” in *Proceedings of the 2nd International Conference on Future Networks and Distributed Systems*, New York, United States of America, 2018, vol 29, pp. 1-7, doi: 10.1145/3231053.3231082.

[46] Ram Mahesh Yadav, “Effective analysis of malware detection in cloud computing,” *Computers & Security*, vol. 83, pp.14-21, 2019, doi: 10.1016/j.cose.2018.12.005.

[47] S. Wang, Q. Yan, Z. Chen, B. Yang, C. Zhao and M. Conti, "Detecting Android Malware Leveraging Text Semantics of Network Flows," in *IEEE Transactions on Information Forensics and Security*, vol. 13, no. 5, pp. 1096-1109, May 2018, doi: 10.1109/TIFS.2017.2771228.

[48] J. kang, S. Jang, S. Li, Y.-S. Jeong, and Y. Sung, “Long short-term memory-based malware classification method for information security,” *Computers & Electrical Engineering*, vol. 77, pp. 366-375, 2019, doi: 10.1016/j.compeleceng.2019.06.014.

[49] P. Prasse, L. Machilca, T. Pevny, J. Havelkam and T. Scheffer, “Malware detection by analysing encrypted network traffic with neural networks,” in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, Springer, Cham, 2017, vol. 10535, pp. 73-88, doi: 10.1007/978-3-319-71246-8_5.

[50] J. Malik and R. Kaushal, “CREDROID: Android malware detection by network traffic analysis,” in *Proceedings of the 1st acm workshop on privacy-aware mobile computing*, New York, NY, United States, 2016, pp. 28-36, doi: 10.1145/2940343.2940348.

[51] J. Deogirikar and A. Vidhate, "Security attacks in IoT: A survey," *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC),* Palladam, India, 2017, pp. 32-37, doi: 10.1109/I-SMAC.2017.8058363.

[52] A. Shifa, M. N. Asghar and M. Fleury, "Multimedia security perspectives in IoT," *2016 Sixth International Conference on Innovative Computing Technology (INTECH),* Dublin, Ireland, 2016, pp. 550-555, doi: 10.1109/INTECH.2016.7845081.