International Journal of Internet of Things 2019, 8(1): 10-16
DOI: 10.5923/j.ijit.20190801.02

Threat Detection Model in Edge for the Internet of Things - A Comprehensive Multi-Layer Solution to Detect Threats in Edge and Cluster Large Networks of IoT

Karthika Venkatraman, Jihad Qaddour[*]

Department of Information Technology, Illinois State University, Normal, United States of America

Abstract The Internet of Things (IoT) is emerging rapidly, with a huge impact on consumer products, resulting in the emergence of AI-based products such as Google Home and Amazon Echo. As the scale of IoT products gets larger, the more they would have to process personal information of users on a daily basis and as a result, the enterprises will have to enhance the security features to protect the devices from attacks. The addition of security features might effectively mean an increase in processing capability, increase in cost and storage, while the device-makers look to keep the prices low. This paper attempts to analyze security risks surrounding the Internet of Things and offer intrusion detection solution at the edge by a detection mechanism in the router.

Keywords Internet of things, Security, Intrusion detection, Edge, Router

1. Introduction

In today's IoT landscape, with the technology still being nascent, it is greatly vulnerable to a myriad of attacks such as Distributed Denial of Service (DDoS), eavesdropping and data leaks as detailed in [3]. However, due to its limited processing capacity and limited storage, implementing traditional solutions such as periodic rotation of passwords, multifactor authentication, AES-128 encryption, host-based Intrusion Detection Systems/Intrusion Prevention systems, anti-virus software, malware detection, Distributed Denial of Service protection, bot protection remain a challenge.

As a result, solutions surrounding intrusion detection or intrusion prevention need to account for the limitations in IoT devices and work around it. The challenges that stem from limitations in IoT could be potentially circumvented through a distributed processing approach that does the processing across devices in the network. However, this will require manufacturers to scale up the processing, memory and storage capacity of devices in the future. Alternatively, the IoT devices could be clustered and a centralized processing manager could be deployed for each IoT cluster. While this approach could bring in the benefits of distributed control and inter-manager communication for alerts and coordination, this will also add to the hardware overhead in the IoT network causing a significant increase in the overall cost of implementation.

In this paper, a solution that effectively utilizes the untapped processing capacity of routers is presented in the form of a software application that could be installed in a router and serve as a threat detection mechanism in an on-going basis. This paper is organized into six sections. The second section provides a literature review of existing work discussing existing solutions. The third section discusses challenges in existing work which this paper attempts to solve. The fourth section illustrates the proposed new architecture of threat detection in edge, along with a cluster management method and automation technique to update benchmark values.

2. Literature Review

As businesses increasingly get electronic devices into the market that can interact with each other, regulatory measures are necessary to ensure security in them. However, in the case of IoT, the regulatory efforts are not consistent with the unprecedented growth of devices [2]. As per the report demonstrated in [4], DBS Asian Insights has predicted that the IoT installed base will grow from 6.3M units in 2016 to 1.25B in 2030. This could potentially offset the security mechanisms that protect technology preceding IoT leaving networks open for attacks and loss of data. Recognizing that the risks outweigh opportunities in IoT, the Internet of Things Cybersecurity Improvement Act was passed in 2017,

[*] Corresponding author:
jqaddour@ilstu.edu (Jihad Qaddour)

Published online at http://journal.sapub.org/ijit

which seeks to enforce security in devices procured by the government. However, a broader regulation as of today is not a reality [5].

One of the major deterrents of IoT security is the economic value it brings. The return on investment for the funds dispatched in enhanced security does not make economic sense for companies to actively pursue security. As a result, the possibility of improved security might not materialize in the foreseeable future. This calls for cost-effective, readily- available solutions that could be incorporated without a significant demand for infrastructure on the part of device-makers or end-users.

This warrants an immediate security solution that addresses the most crucial among problems facing IoT. As discussed in [6], there are three ways to go about implementing security in IoT viz at a centralized location, on-the-edge and on the gateway. The benefits of implementation include the ability to have an overall view, reduced cost, energy efficiency, and ease-of-sharing. However, it comes with the cost of single point failure and slow feedback. Implementing security in gateway equipment such as cellphones allows instant feedback from users, privacy, the ubiquity of equipment and the possibility of data aggregation however at the cost of complexity, software management, and reduced security. The third method is security in the edge which brings instant feedback, reduces data transfer and improves security. The disadvantages of security in edge according to [6] is the cost, energy consumption and lack of overview.

These disadvantages can be addressed by effective use of equipment that is already available in the market. [7] demonstrates a load-balancing solution by tapping on to underutilized line cards to share the router’s processing power with supporting algorithms. This brings us to the potential of routers which could be used for edge security and address its inherent disadvantages cost, energy consumption and lack of overview.

[8] is another instance of the router’s potential to support resource-intense activities. [8] is an adaptive edge-computing solution based on regressive admission control and fuzzy queuing to react to quality-of-service changes within heterogeneous networks. This supports the security solution described in the rest of the paper.

3. Challenges in Current Work

A. *Distributed Security Mechanism for resource-Constrained IoT devices*

[9] offers a distributed mechanism to secure IoT network in multiple layers. The approach here is to secure the entire network by processing lighter operations in the end-point, heavier operations in a gateway while using symmetric encryption for data objects combined with native wireless security. In this case, the gateway provides additional protection by securing data using TLS. The proposed solution in the paper has been demonstrated through real-time evaluations for targeted Class-0 IoT devices. While this offers layers of security, this does not address the need for intrusion detection in IoT.

B. *Analysis of Security Mechanisms Based on Clusters IoT Environments*

[10] begins with a systematic analysis of existing research in IoT at the privacy level and control access in this type of environment. Then presents the issues that need to be immediately addressed, from different clusters and identified areas of application. This paper does not offer a solution to the issues identified.

C. *Flow-sensitive and Context-sensitive mechanisms*

[11] attempts to capture the baseline behavior of IoT devices to use them for anomaly detection. While noting that the existing methods are either flow-sensitive or context-sensitive information to capture system call context which limits the scope and accuracy. The paper proposes using context-sensitive features based on control-flow and details the process to detect anomalies using the same.

4. Proposed Solution

This paper proposes a solution in two-layers. The first layer utilizes the processing capacity of routers to detect threats in IoT edge. The focus areas of this paper are Distributed Denial of Service (DDoS), password attacks and code injection which can further be extended to detect all other types of threats in the future. Threat detection in the edge is implemented as a software application that can be installed in the router to monitor incoming and outgoing traffic pertaining to IoT. The second layer organizes the IoT network into clusters with a fixed size, each with a manager to facilitate intra-cluster inter-cluster communication.

A. IoT security through a router application

A router application monitors all incoming and outgoing traffic to and from IoT devices to detect Distributed Denial of Service, password attacks and code injection in three software modules. Intelligence about Denial of Service can be obtained by observing the following factors measured against baseline values expected for both peak and regular hours of IoT traffic.

- The volume of ingress traffic in terms of packets per second and high level of deviation from baseline
- Percentage of CPU utilization and high level of deviation from baseline
- Throughput and high level of deviation from baseline
- Bandwidth and high level of deviation from baseline
- More than one parameter demonstrating moderate to high deviation from baseline
- One or more parameters demonstrating moderate to high deviation from baseline over a period of time
- All of the aforementioned deviations occur at a time frame that is not peak hour activity
- Possible attack packets from one more source

Denial of Service (DoS) and its variants can be measured with a high degree of accuracy using any of the following algorithms PART, BayesNet, IBK, Logistic, J48, Random Committee, and InputMapped. However, research in [13] along with [14] shows PART as the top choice for DoS with an accuracy of 98.5539% which is what will be used in the proposed solution. PART is a Partial Decision Tree algorithm and is an advanced version of C4.5 and RIPPER algorithms. The most notable aspect of this algorithm is it doesn't need global optimization to generate reliable rules. PART decision trees will be used on IoT dataset in edge to classify attacks from possible false alarms.

Authentication in home appliances is usually carried out through mobile phones using an authentication application. The solution proposed in this paper relies on correctly identifying legitimate devices and blocking all other unknown devices out of the network. The attack detection, in this case, is distributed among router and mobile device.

Password attack detection can happen by tracking authentication requests coming from known sources and identifying legitimate devices through multi-factor behavior-based fingerprinting techniques involving factors beyond IP, mac address, OS and RFID.

In the event that a node in the IoT network is compromised, code injection is a likely next step on the part of attackers. Code or malware injection can be detected using proprietary software such as Avast, Bitdefender, BullGuard, Panda or McAfee. Alternatively, open-source equivalents can be used to achieve the same results. However, the biggest disadvantage here is the resources it consumes. Malware detection is to be performed in a distributed fashion between edge, gateway and remote processing by accessing a cloud-based application.

In order to optimize resource utilization in the router or other edge devices, malware detection and code injection should between on-device computation and cloud-based malware signature detection. The module in IDS software at the edge should be built to detect common signatures and if the code needs to be checked for advanced threats or analyzed further, it should be sent to a cloud-based solution that detects malware/malicious code and alerts administrators about it. However, the cloud-based solution could potentially increase the turnaround time.

While advanced routers such as CISCO 7600 come with DDOS protection and have an intrusion capability, they are not designed to handle IoT traffic nor are the performance metrics benchmarked and optimized for IoT. The benefit of the proposed solution is, it is designed specifically for a particular IoT network whose traffic is observed and analyzed over time in an automated manner. This level of granularity greatly helps minimize false positives and false negatives which are crucial success factors for IoT.

B. *Clustered approach with managers to facilitate efficient communication and security*

While the intrusion detection mechanism in the edge might suffice for small networks, additional controls would prove useful in case of large enterprise-grade IoT networks.

The second step in ensuring security in IoT is to organize the entire IoT network in the form of clusters for ease of management and communication. The devices in the IoT network could be grouped based on the content and be assigned a manager which serves as the single point of communication for the cluster and has maximum processing capacity in the cluster. This manager will handle authentication for the cluster, facilitate updates and perform security functions in a distributed security solution for IoT network. The biggest benefit of this approach is that it renders the capability to perform resource-intense functions for the entire clusters comprising or heterogeneous devices that may or may not have the capacity to support security functions themselves.

C. *Authenticating cluster managers and IoT intrusion detection application*

An important success factor is to enforce mutual authentication between IoT cluster managers and IoT intrusion detection application. This step will help eliminate the possibility of rouge edge devices and rogue endpoints in the network. Furthermore, authentication should also be enforced when IoT cluster managers interact with each other.

This can be accomplished through an authentication mechanism involving PUF algorithm. The Physical Unclonable Function (PUF) described in [20] is an entity embodied in a physical structure such as microchips and is often used in applications with high-security requirements. This will be used along with a fuzzy extractor or key extractor to uniquely identify IoT devices, device managers and router.

[21] shows a mechanism of DDoS protection implemented in routers through packet classification and traffic control. While the mechanism proposed in [12] differs from the model discussed in this paper, it demonstrates the feasibility of innovative applications in router devices.

5. Architecture

A. Intrusion Detection

The proposed solution for intrusion detection has a traffic handler that parses incoming and outgoing traffic for IoT devices in the network and then analyzes the parsed traffic to identify possible Distributed Denial of Service, password attacks and code injection attacks through three separate modules in this application.

In addition, this application maintains a cloud repository to persist logs that have saves only events of interest over time, allowing complex AI based analytics using cloud resources, while saving storage from logging all events.

Finally, the architecture encloses a module to alert system administrators about an ongoing attack that requires immediate action. In a home network, the owner plays the role of an administrator and receives email alerts.

The intrusion detection solution is to be implemented as an application in the router. In a home network, this eliminates the need for additional hardware and costs associated with it, while utilizing the untapped potential of router.

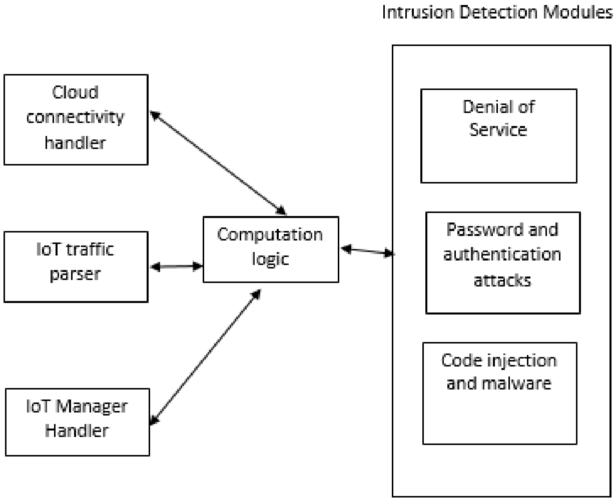
Intrusion Detection Modules
Cloud connectivity handler
IoT traffic parser
IoT Manager Handler
Computation logic
Denial of Service
Password and authentication attacks
Code injection and malware

Figure 1. Architecture for Intrusion Detection

B. Cluster Management

As described in previous sections, the purpose of establishing a clustered IoT system is to distribute layers of security in strategic points in the network instead of consolidating them all in one point, which might cause bottleneck situations.

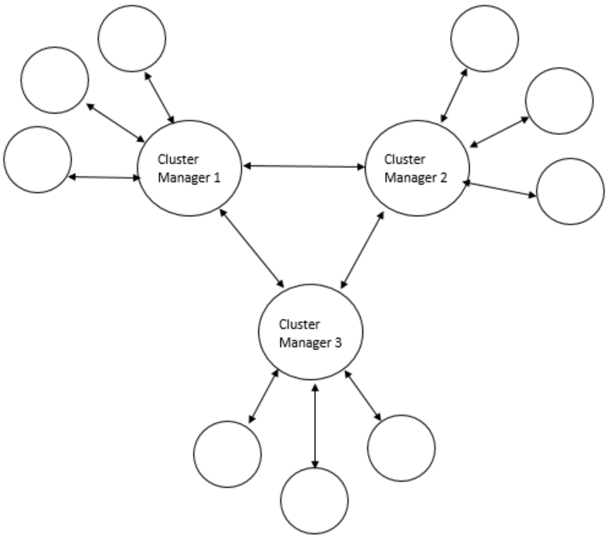

Figure 2. Cluster management in IoT through managers

[18] Some of the most popular algorithms in the industry for cluster management are smart local moving, Louvain, Infomap and label propagation. When these algorithms were analyzed based on the metrics adjusted Rand score, normalized mutual information, and a variant of normalized mutual information used in previous work, it was identified that smart local moving is a better algorithm for cluster management. The study conducted in [18] demonstrated the potential of this algorithm to perform community detection in networks with tens of millions of nodes and hundreds of millions of edges. The smart local move uses the modularity function and optimization of Louvain algorithm [22] and maximizes its benefits for large scale implementation of modularity. It uses a resolution parameter to determine the granularity level, which should be consistent with our cluster size.

The cluster manager could be a raspberry pi which has processing power up to 6.5 W [27]. The usage of raspberry pi to implement IoT is well documented in [23], [24], [25] and [26]. [28] shows how the ubiquitous raspberry pi is an inexpensive and effective way of getting compute power into a range of commercial-class solutions requirements.

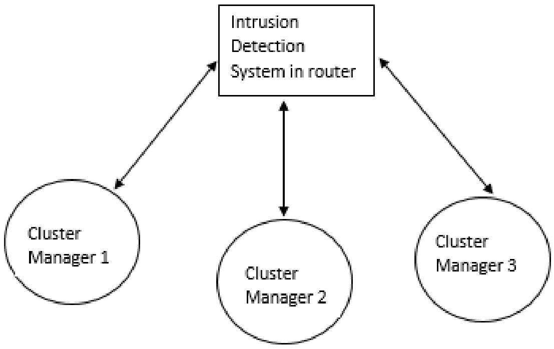

Figure 3. Cluster managers and their interaction with the intrusion detection system

C. Automation for performance benchmarking

Performance metrics are to be determined in a manner consistent with the size and scale of the specific IoT network for which security features are being implemented. Since the baseline values could vastly differ for non-IoT networks and baseline values in an IoT network could potentially change over time, it is essential to calculate and update baseline values periodically. This could be implemented in an automated fashion and be made to run periodically to check and update baseline values.

These baseline parameters are to be determined in various categories such as peak hour metrics, regular hour metrics, and low traffic metrics. The performance values captured in every one of these categories are to be used as baseline values to calculate the deviation of values at a given point in time.

Table 1. Sample Performance Baseline

Performance Metric	Baseline Values In Category		
	Peak hour	Regular hour	Low traffic hour
Utilization	Value 1	Value 2	Value 3
Latency	Value 4	Value 5	Value 6
Dynamic Trending	Value 7	Value 8	Value 9
Jitter	Value 10	Value 11	Value 12
Granular data retention	Value 13	Value 14	Value 15

The values in the above are baseline values to calculate deviation. While the above table primarily features five parameters to measure IoT network performance, this could be further enhanced further to achieve a higher level of accuracy.

The following three tables demonstrate ways to capture deviations that could be utilized to analyze traffic in a structured manner, which would, in turn, facilitate accurate detection of IoT based network attacks.

Table 2. Sample Peak Hour Performance Deviations

Performance Metric	Deviations in peak hour	
	Peak hour	Deviation
Utilization	Peak hour utilization	Deviation 1
Latency	Peak hour latency	Deviation 2
Dynamic Trending	Peak hour dynamic trending	Deviation 3
Jitter	Peak hour jitter	Deviation 4
Granular data retention	Peak hour granular data retention	Deviation 5

Table 3. Sample Regular Hour Performance Deviations

Performance Metric	Deviations in regular hour	
	Peak hour	Deviation
Utilization	Regular hour utilization	Deviation 1
Latency	Regular hour latency	Deviation 2
Dynamic Trending	Regular hour dynamic trending	Deviation 3
Jitter	Regular hour jitter	Deviation 4
Granular data retention	Regular hour granular data retention	Deviation 5

Table 4. Sample Off-Hour Performance Deviations

Performance Metric	Deviations in the low-traffic hour	
	Peak hour	Deviation
Utilization	Off-hour utilization	Deviation 1
Latency	Off-hour latency	Deviation 2
Dynamic Trending	Off-hour dynamic trending	Deviation 3
Jitter	Off-hour jitter	Deviation 4
Granular data retention	Off-hour granular data retention	Deviation 5

D. Overall Architecture

Below is an architecture that demonstrates how cluster managers, IoT Intrusion Detection application and malware detector interacts.

Below is a flow chart that shows the overall workflow of IoT Intrusion Detection Application in the router.

E. Future Enhancements

Machine learning techniques in conjunction with data mining techniques listed in [15] can be applied in enterprise-grade advanced routers that can learn traffic patterns over time and can render capabilities to detect advanced persistent threats and other stealth attacks designed to evade detection. Furthermore, detection techniques listed in [17] can be used to maximize the effectiveness of intrusion detection in the router. In addition, open source databases of malicious IP addresses and MAC addresses can be used to detect attack packets.

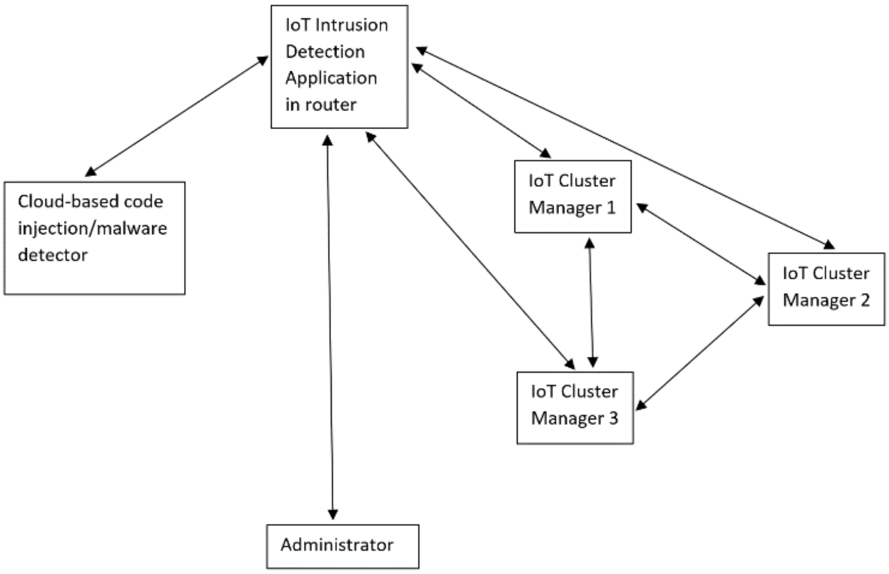
IoT Intrusion Detection Application in router
Cloud-based code injection/malware detector
IoT Cluster Manager 1
IoT Cluster Manager 2
IoT Cluster Manager 3
Administrator

Figure 4. Interaction between IoT Intrusion Detection application, Cluster managers, Cloud-based Malware detection system and the administrator

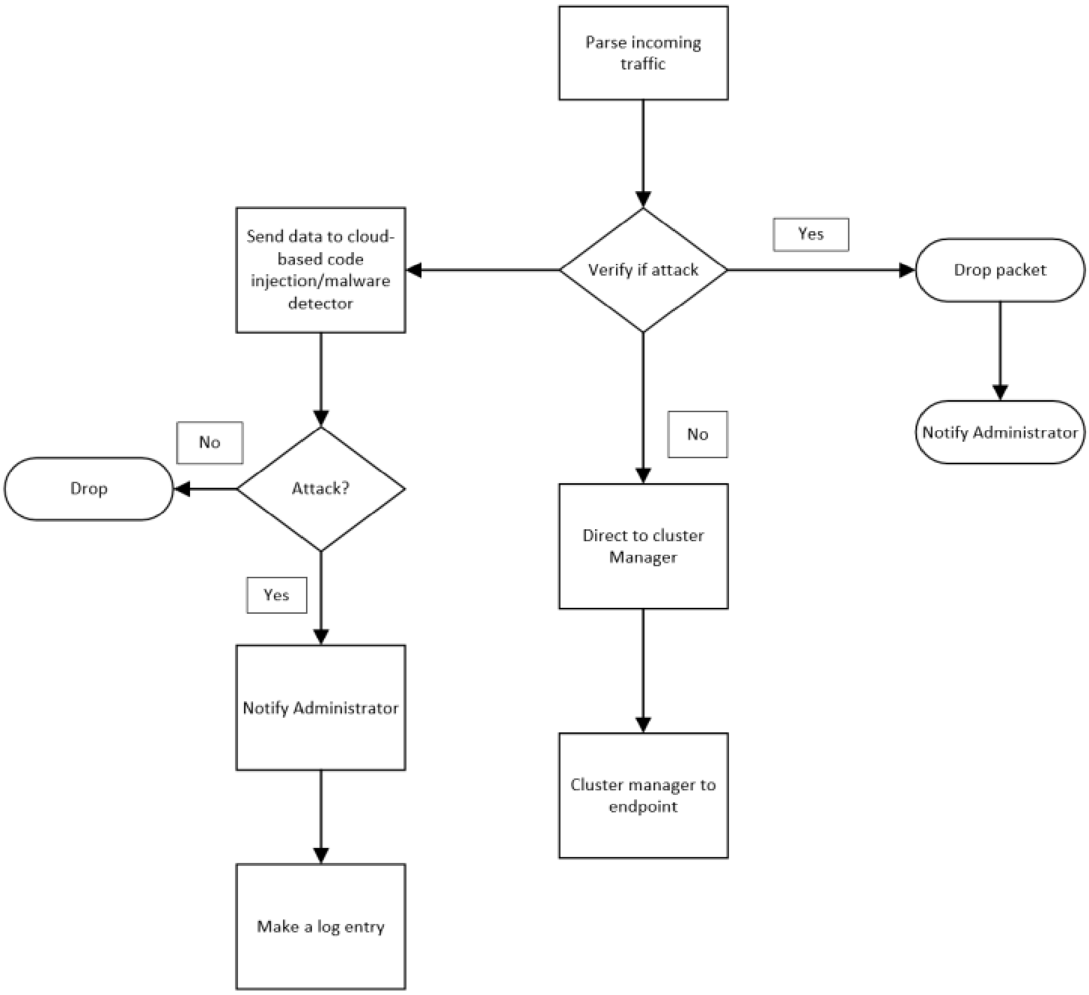

Figure 5. The workflow of IoT Intrusion Detection Application

6. Conclusions

Thus the solution discussed so far, utilizes the unused processing capacity of routers to host an IDS/IPS application, while eliminating additional hardware needs. Furthermore, it offers cluster management and performance benchmarking approaches that could be applied to existing IoT networks while keeping the costs low [28].

REFERENCES

[1] Marat Radan, Isaac Keslassy, "Tapping into the router's unutilized processing power", *Computer Communications (INFOCOM) 2015 IEEE Conference on*, pp. 2569-2577, 2015.

[2] Kozlov, D., Veijalainen, J., & Ali, Y. (2012, February). Security and privacy threats in IoT architectures. In *Proceedings of the 7th International Conference on Body Area Networks* (pp. 256-262). ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[3] Razzaq, M. A., Gill, S. H., Qureshi, M. A., & Ullah, S. (2017). Security issues in the Internet of Things (IoT): a comprehensive study. *International Journal of Advanced Computer Science and Applications (IJACSA)*, *8*(6), 383-388.

[4] https://www.forbes.com/sites/louiscolumbus/2018/12/13/2018-roundup-of-internet-of-things-forecasts-and-market-estimates/#7066fffb7d83.

[5] Laya, A., Markendahl, J., & Andersson, P. (2013). Business Challenges for Services Based on New Technology-Analysis of IoT Service and Mobile Payment Cases. *Effective, Agile and Trusted eServices Co-Creation*, 91.

[6] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.

[7] M. Radan and I. Keslassy, "Tapping into the router's unutilized processing power," *2015 IEEE Conference on Computer Communications (INFOCOM)*, Kowloon, 2015, pp. 2569-2577. doi: 10.1109/INFOCOM.2015.7218647.

[8] M. Jutila, "An Adaptive Edge Router Enabling Internet of Things," in *IEEE Internet of* Things *Journal*, vol. 3, no. 6, pp. 1061-1069, Dec. 2016. doi: 10.1109/JIOT.2016.2550561

[9] King, J., & Awad, A. I. (2016). A distributed security mechanism for resource-constrained IoT devices. *Informatica*,

40(1).

[10] Gaona-Garcı́a, P., Montenegro-Marin, C., Prieto, J. D., & Nieto, Y. V. (2017). Analysis of Security Mechanisms Based on Clusters IoT Environments. *International Journal of Interactive Multimedia & Artificial Intelligence*, *4*(3).

[11] https://patents.google.com/patent/US20030140088A1/en.

[12] Davie, B. S., & Medved, J. (2009, August). A programmable overlay router for service provider innovation. In *Proceedings of the 2nd ACM SIGCOMM workshop on Programmable routers for extensible services of tomorrow* (pp. 1-6). ACM.

[13] Noureldien, N. A., & Yousif, I. M. (2016). Accuracy of machine learning algorithms in detecting DoS attacks types. *Science and Technology*, *6*(4), 89-92.

[14] G. Carl, G. Kesidis, R. R. Brooks and Suresh Rai, "Denial-of-service attack-detection techniques," in *IEEE Internet Computing*, vol. 10, no. 1, pp. 82-89, Jan.-Feb. 2006. doi: 10.1109/MIC.2006.5.

[15] Agah, S. A. (2017). Investigating Identification Techniques of Attacks in Intrusion Detection Systems Using Data Mining Algorithms. IJCSNS, 17(5), 174.

[16] G. Carl, G. Kesidis, R. R. Brooks and Suresh Rai, "Denial-of-service attack-detection techniques," in *IEEE Internet Computing*, vol. 10, no. 1, pp. 82-89, Jan.-Feb. 2006. doi: 10.1109/MIC.2006.5.

[17] Babar, S., Stango, A., Prasad, N., Sen, J., & Prasad, R. (2011, February). Proposed embedded security framework for internet of things (iot). In Wireless Communication, Vehicular Technology, Information Theory and Aerospace & Electronic Systems Technology (Wireless VITAE), 2011 2nd International Conference on (pp. 1-5). IEEE.

[18] Emmons S, Kobourov S, Gallant M, Börner K (2016) Analysis of Network Clustering Algorithms and Cluster Quality Metrics at Scale. PLoS ONE 11 (7): e0159161. doi:10.1371/journal.pone.0159161.

[19] bin Basir, M. A., & binti Ahmad, F. New Feature Selection Model-Based Ensemble Rule Classifiers Method For Dataset Classification.

[20] Tehranipoor, F., Karimian, N., Xiao, K., & Chandy, J., "DRAM based intrinsic physical unclonable functions for system level security", *In Proceedings of the 25th edition on Great Lakes Symposium on VLSI, (pp. 15-20). ACM, 2015.*

[21] Chan, E. Y., Chan, H. W., Chan, K. M., Chan, V. P., Chanson, S. T., Cheung, M. M., ... & Lam, L. C. (2004, May). IDR: an intrusion detection router for defending against distributed denial-of-service (DDoS) attacks. In Parallel Architectures, Algorithms and Networks, 2004. Proceedings. 7th International Symposium on (pp. 581-586). IEEE.

[22] https://github.com/deepminder/SLM4J.

[23] Maksimović, M., Vujović, V., Davidović, N., Milošević, V., & Perišić, B. (2014). Raspberry Pi as Internet of things hardware: performances and constraints. *design issues*, *3*(8).

[24] Patchava, V., Kandala, H. B., & Babu, P. R. (2015, December). A smart home automation technique with raspberry pi using iot. In *2015 International Conference on Smart Sensors and Systems (IC-SSS)* (pp. 1-4). IEEE.

[25] Zhao, C. W., Jegatheesan, J., & Loon, S. C. (2015). Exploring iot application using raspberry pi. *International Journal of Computer Networks and Applications*, *2*(1), 27-34.

[26] Ansari, A. N., Sedky, M., Sharma, N., & Tyagi, A. (2015, January). An Internet of things approach for motion detection using Raspberry Pi. In *Proceedings of 2015 International Conference on Intelligent Computing and Internet of Things* (pp. 131-134). IEEE.

[27] https://www.raspberrypi.org/documentation/faqs/#powerReqs.

[28] Edwards, C. (2013). Not-so-humble raspberry pi gets big ideas. *Engineering & Technology*, *8*(3), 30-33.